

# Usporedba metoda za kompresiju oblaka točaka

---

**Belemečić, Leonardo**

**Master's thesis / Diplomski rad**

**2023**

*Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj:* **University of Dubrovnik / Sveučilište u Dubrovniku**

*Permanent link / Trajna poveznica:* <https://um.nsk.hr/um:nbn:hr:155:435496>

*Rights / Prava:* [In copyright](#)/[Zaštićeno autorskim pravom.](#)

*Download date / Datum preuzimanja:* **2024-07-28**



*Repository / Repozitorij:*

[Repository of the University of Dubrovnik](#)



SVEUČILIŠTE U DUBROVNIKU  
ODJEL ZA ELEKTROTEHNIKU I RAČUNARSTVO

LEONARDO BELEMEČIĆ

USPOREDBA METODA ZA  
KOMPRESIJU OBLAKA TOČAKA

DIPLOMSKI RAD

Dubrovnik, siječanj, 2023.

SVEUČILIŠTE U DUBROVNIKU  
ODJEL ZA ELEKTROTEHNIKU I RAČUNARSTVO

USPOREDBA METODA ZA  
KOMPRESIJU OBLAKA TOČAKA

DIPLOMSKI RAD

Studij: Elektrotehničke i komunikacijske tehnologije u pomorstvu

Kolegij: Kompresija podataka i zaštitno kodiranje

Mentor: doc.dr.sc. Anamaria Bjelopera

Student: Leonardo Belemečić

Dubrovnik, siječanj, 2023.

## SAŽETAK

Oblak točaka jedan je od najkorištenijih oblika prikazivanja različitih 3D objekata te njihovih atributa, a predstavlja skupove točaka u  $(x, y, z)$  koordinatama trodimenzionalnog virtualnog prostora dobivenih kao rezultat različitih vrsta skeniranja. Kroz rad će se objasniti procesi dobivanja oblaka točaka te će se opisati baze slika koje se koriste za analize ovakvog načina prikazivanja objekata. Kako bi se takvi realistični načini prikazivanja mogli koristiti u različitim primjenama pohranjivanja i prijenosa, jedan od prioriteta je i način kompresije oblaka točaka s težnjom na lossy kompresiju. U ovome radu su opisani i analizirani postojeći načini kompresije oblaka točaka uključujući MPEG G-PCC i V-PCC te kompresije temeljene na algoritmima dubokog učenja uključujući i najnovija dostignuća. Za usporedbu rezultata pojedinih vrsta kompresije koriste se subjektivne i objektivne mjere za ocjenu kvalitete oblaka točaka.

***Ključne riječi:** Oblak točaka; V-PCC kompresija; G-PCC kompresija; Lossy kompresija; Lossless kompresija*

## ABSTRACT

The point cloud is one of the most used forms of displaying various 3D objects and their attributes, and represents sets of points in  $(x, y, z)$  coordinates of the three-dimensional virtual space obtained as a result of different types of scanning. The paper will explain the processes of obtaining point clouds, and it will be described the database of images that are used for the analysis of this way of displaying objects. In order to be able to use such realistic rendering methods in various storage and transmission applications, one of the priorities has become a point cloud compression method with a tendency towards lossy compression. In this paper, existing methods of point cloud compression are described and analyzed, including MPEG G-PCC and V-PCC, and compression based on deep learning algorithms, including the latest developments. To compare the results of individual types of compression, subjective and objective measures are used to evaluate the quality of point clouds.

***Keywords:** Point cloud; V-PCC; G-PCC; Lossy compression; Lossless compression*

# SADRŽAJ

1	UVOD .....	1
2	Oblak točaka.....	2
2.1	3D lasersko skeniranje .....	3
2.2	Metode fotogrametrije .....	5
3	Problemi i razvoj kompresije oblaka točaka .....	8
4	Skupovi podataka najčešćih oblaka točaka i 3D uzoraka.....	10
5	MPEG 3D PCC STANDARDI .....	12
5.1	MPEG kompresija oblaka točaka temeljena na videu (V-PCC).....	13
5.1.1	V-PCC arhitektura.....	14
5.1.2	Generiranje zakrpa .....	14
5.1.3	Pakiranje zakrpa .....	17
5.1.4	Geometrija i mapa zauzetosti .....	18
5.1.5	Ispuna slike, širenje grupe, ponovno bojanje i kompresija videa.....	18
5.1.6	Izgladivanje geometrije i atributa.....	20
5.1.7	Kompresija atlasa .....	21
5.2	MPEG kompresija oblaka točaka temeljena na geometriji (G-PCC) .....	22
5.2.1	G-PCC arhitektura.....	22
5.2.2	Koordinacija transformacije i vokselizacije .....	23
5.2.3	Kodiranje geometrije.....	23
5.2.4	„Octree“ kodiranje.....	24
5.2.5	„Trisoup“ kodiranje.....	27
5.2.6	Prediktivno kodiranje .....	28
5.2.7	Aritmetičko kodiranje.....	28
5.2.8	Pretvorba atributa, ponovno bojanje i kodiranje .....	30
6	Kompresija oblaka točaka temeljena na algoritmima dubokog učenja.....	33
6.1	Duboko učenje temeljeno na točkama .....	33
6.2	Duboko učenje temeljeno na stablu .....	34
6.3	Kompresija geometrije oblaka točaka temeljena na vokselu.....	35
6.3.1	Kompresija slike s gubicima .....	36
6.3.2	Kompresija slike bez gubitaka .....	39
7	Procjena kvalitete oblaka točaka .....	42
7.1	Subjektivna procjena kvalitete oblaka točaka.....	42
7.2	Objektivna procjena kvalitete oblaka točaka .....	43

7.2.1	Mjere temeljene na projekcijama oblaka točaka .....	43
7.2.2	Mjere temeljene na geometriji i/ili atributima.....	44
8	Usporedba rezultata različitih metoda kompresije .....	47
9	Primjena oblaka točaka .....	50
10	ZAKLJUČAK .....	52
11	LITERATURA.....	53
12	POPIS SLIKA .....	54

# 1 UVOD

Predmet ovog diplomskog rada je usporedba metoda za kompresiju oblaka točaka. Iskustva proširene (engl. *Augmented Reality*), virtualne (engl. *Virtual Reality*) i mješovite stvarnosti (engl. *Mixed Reality*) trenutačno postaju sve popularnija jer zanimanje za snimanjem stvarnog svijeta u više dimenzija i predstavljanje korisniku nikad nije bilo veće. Ovakve tehnologije zahtijevaju ogromne računalne resurse za potrebe pohrane, prijenosa, obrade i vizualizacije pa je iz tog razloga potrebno poboljšati kvalitetu kompresije i obrade signala. U novije vrijeme s napretkom tehnologija 3D skeniranja pojavio se novi pristup temeljen na prikazima pomoću oblaka točaka. Oblak točaka je jednostavna struktura podataka definirana kao skup neorganiziranih točaka u 3D prostoru, obično zastupljenih u kartezijevom koordinatnom sustavu s x,y,z koordinatama. Može sadržavati razne attribute kao što su boja, dubina, udaljenost, a istovremeno se koristiti za predstavljanje statičkih i dinamičkih 3D objekata. Pomoću oblaka točaka mogu se ispuniti različiti zahtjevi u smislu preciznosti, gustoće točaka i srodnih atributa.

Svrha ovog diplomskog rada je istražiti, analizirati i usporediti metode za kompresiju oblaka točaka te ustanoviti koja je metoda najkvalitetnija i najkorištenija. Razvoj načina kompresije oblaka točaka rezultirao je velikim brojem efektivnih metoda kompresije koje omogućavaju jednostavno i lako spremanje i obradu. Cilj je odrediti koja metoda je najpouzdanija i gdje imamo najmanje grešaka u postupku kompresije.

Diplomski rad podijeljen je u 10 poglavlja. U uvodnom dijelu navodi se predmet istraživanja, upoznavanje s temom. U drugom poglavlju se opisuje oblak točaka, koje su njegove karakteristike, gdje se sve koristi i od čega se sastoji, dok se u trećem navode potencijalni problemi i razvoj oblaka točaka. U četvrtom poglavlju se opisuju skupine podataka koje se upotrebljavaju za oblake točaka. Nakon toga opisuje se MPEG (engl. *Moving Picture Experts Group*) kompresiju oblaka točaka i njene podjele. U šestom poglavlju se razrađuje kompresija oblaka točaka temeljena na algoritmima dubokog učenja, a sedmom poglavlju govorimo o procjeni kvalitete oblaka točaka gdje su objašnjene subjektivne i objektivne mjere za procjenu kvalitete oblaka točaka. U osmom poglavlju je opisana usporedba rezultata raznih metoda kompresije, te je u devetom poglavlju navedena primjena oblaka točaka. Zadnje poglavlje donosi zaključak i kratki osvrt na rad.

## 2. Oblak točaka

Oblak točaka (engl. *Point Cloud*) je skup pojedinačnih točaka ugrađen u 3D prostor koji sadrži geometrijske informacije opisane x,y,z koordinatama i fotometrijska svojstva kao što su boja, dubina, udaljenost itd. Svrha tog skupa točaka je kreiranje fotorealističnog 3D objekta. Oblaci točaka kreiraju se obično pomoću 3D skenera ili softvera za fotogrametriju koji mjeri više točaka na površinama objekta i mogu sadržavati razne informacije kao što je: boja, intezitet ili transparentnost.

Korištenje ove metode postaje sve popularnije i ako se usporede s modelima koji su ručno napravljeni oblaci točaka pružaju nam jednostavniji i realističniji prikaz. Oblaci točaka mogu se podijeliti na 2 komponente: geometriju (položaj svake pojedine točke) i attribute (dodatne informacije povezane sa svakom od tih točaka). Oblak točaka s vremenskom dimenzijom naziva se dinamičkim oblakom točaka, a bez vremenske dimenzije oblak statičkih točaka. Statički nam prikazuju samo jedan oblak točaka, a dinamički nam prikazuju sekvencu broja točaka.

Oblaci točaka često se usklađuju s 3D modelima ili drugim oblacima točaka što je proces poznat kao registracija skupa točaka. Broj točaka unutar oblaka točaka može se kretati od tisuću čineći jednostavne 3D objekte, do milijardi kako bi realno predstavljali složene 3D scene. Postoje dva glavna pristupa: čisto geometrijski koji se temelji na skupinama podataka „octree“ koje opisuju lokacije točaka u 3D prostoru i hibridni koji se temelji i na geometriji i na video kodiranju. Prvi pristup nam pruža točne informacije o 3D geometriji, ali sadrži vremensku redundanciju, dok drugi može učinkovito ukloniti vremensku redundanciju, ali se nakon projekcije može primijetiti manja geometrijska preciznost. Radi toga se mora optimizirati kompromis između učinkovitosti kompresije i točnog predviđanja.

S razvojem tehnologije prikupljanja 3D podataka postalo je puno lakše i brže dobiti oblake točaka iz stvarnih objekata. Međutim, različite tehnologije istovremeno unose različite neželjene informacije u prikaz podataka. Neka su izobličenja uzrokovana vanjskim razlozima kao što je neizbježno kretanje mobilnog laserskog skenera (MLS). Neke pogreške generiraju algoritmi rekonstrukcije. Radi toga je izuzetno važno razumjeti osnovni koncept tehnologija akvizicije kako bi se u najvećoj mjeri postigla učinkovita kompresija.

Općenito podaci oblaka točaka se snimaju pomoću: 3D laserskog skeniranja i tehnologije fotogrametrije.

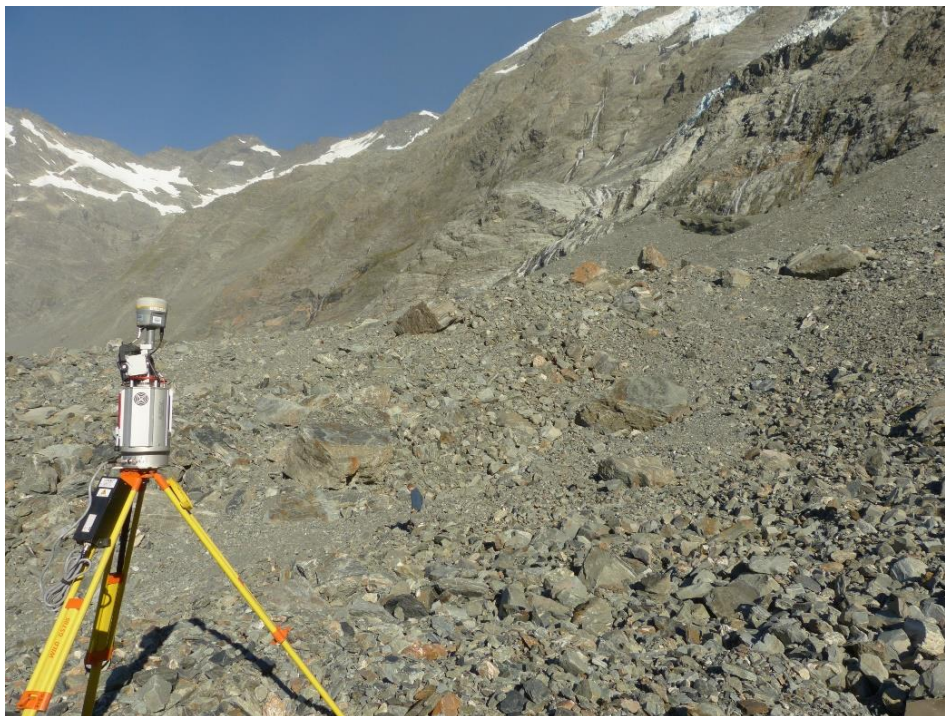


## 2.1 3D lasersko skeniranje

Tehnologija 3D laserskog skeniranja poznata je i kao svjetlosno zamjećivanje i klasifikacija - LiDAR (engl. *Light Detection and Ranging*) te snima 3D okruženje pomoću dvije različite tehnike mjerenja dometa: vremena leta i faznog pomaka. Skener na bazi vremena leta emitira laserski puls i mjeri vrijeme putovanja reflektiranog pulsa. Za to vrijeme skener temeljen na faznom pomaku emitira kontinuirani val moduliran amplitudom i mjeri fazni pomak između emitiranih i reflektiranih signala. Skeneri vremena leta imaju širi raspon mjerenja od metra do kilometra, dok skeneri faznog pomaka postižu veću točnost raspona i brzinu akvizicije što može uhvatiti tisuće točaka u sekundi.

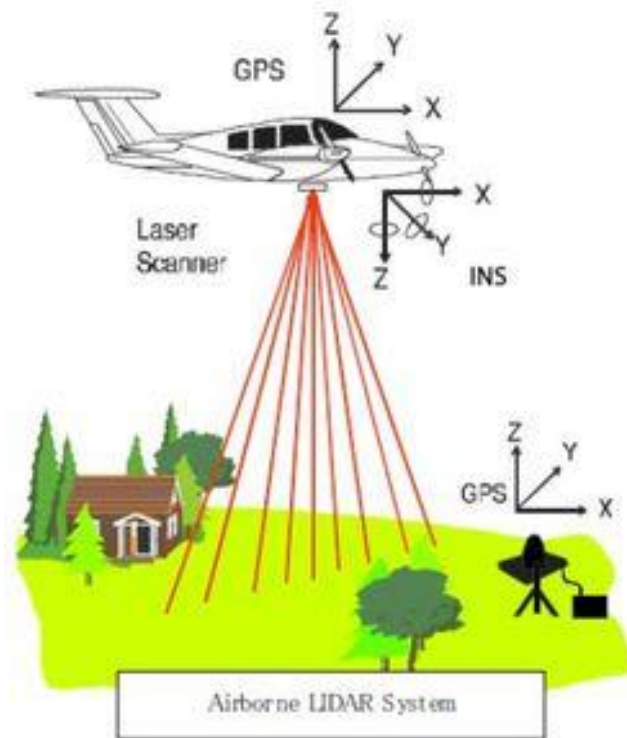
Skeneri se mogu podijeliti u tri kategorije ovisno o njihovim operativnim platformama kao što su zemaljski laserski skeneri – TLS (engl. *Terrestrial Laser Scanner*), laserski skener u zraku – ALS (engl. *Airborne Laser Scanner*) i mobilni laserski skener – MLS (engl. *Mobile Laser Scanner*).

TLS se obično montira na stativ smješten na fiksnom mjestu na tlu tijekom skeniranja okoline kako je prikazano na slici 1 zbog čega je poznat i kao zemaljski LiDAR skener. Pruža najveću stabilnost i točnost raspona jer ostaje miran tijekom operacije akvizicije. Ipak unatoč svemu tome potrebno je puno vremena za skeniranje i zahtijeva i da je okolina koju snima statična.



Slika 1. Primjer TLS skenera [7]

ALS predstavlja skener koji se montira na zrakoplov kako se vidi na primjeru na slici 2.. Namijenjen je skeniranju velikih objekata/scena, a poznat je i pod nazivom zračni LiDAR skener. Sposoban je uhvatiti podatke za oblak točaka, te terene u gradu s relativno niskom točnošću.



Slika 2. Primjer zračnog ALS skenera [8]

Na slici 3. prikazan je MLS poznat i kao mobilni LiDAR, montira se na mobilnu platformu kao što je vozilo ili dron za dinamičko skeniranje okoliša tijekom kretanja. Ipak vibracija vozila dovodi do nedosljednih uvjeta za skeniranje i može uzrokovati geometrijske pomake uhvaćenih 3D točaka. Međutim, upotreba LiDAR senzora ograničena je zbog njihove visoke cijene u usporedbi s onima koji koriste fotogrametrijske pristupe.



Slika 3. Primjer mobilnog MLS skenera [9]

## 2.2 Metode fotogrametrije

Nedavno su razvijeni različiti fotogrametrijski algoritmi i tehnologije koji su stekli veliku popularnost zbog lakog pristupa potrebnoj opremi. Prvo se snima zbirka preklapajućih 2D slika snimljenih s različitih lokacija koje predstavljaju različite dijelove ciljnog objekta. Nakon toga se koriste fotogrametrijski algoritmi za procjenu relativne lokacije 2D slika i njihovo rekonstruiranje u 3D oblak točaka. Najpopularniji algoritmi su: algoritam strukture iz pokreta – SfM (engl. *Structure from Motion*) i stereo algoritam – MVS (engl. *Multi-View Stereo*). Dodatno dubinske RGB-D (engl. *Red, Green, Blue - Depth*) kamere iskorištavaju obje metode. Riječ je o spoju RGB kamere u boji i senzora dubine. 3D oblak točaka u ovom se slučaju generira kombiniranjem 2D lokacija rasteriziranih piksela boja i odgovarajućih dubinskih vrijednosti.

Skeneri koji koriste fotogrametriju su bolji u pogledu prenosivosti od tehnika skeniranja temeljenih na LiDAR-u jer se ti skeneri montiraju na objekte pa im nedostaje prenosivost, pa se kao posljedica toga dinamički uhvaćeni oblaci točaka mogu deformirati što stvara poteškoće pri kompresiji. Zbog toga se češće koriste fotogrametrijske tehnike zbog svoje jednostavnosti pristupa i implementacija. Primjer fotogrametrije i 3D skeniranja prikazan je na slici 4. Ovisno o korištenom algoritmu oblaci točaka mogu biti nepotpuni, preklapajući ili kaotični. Predviđa se da će u skorijoj budućnosti skeneri biti dovoljno kompaktni za integraciju u mobilni telefon. Isto tako s napretkom tehnologija za duboko učenje moći će se postići brža rekonstrukcija i bolje otkrivanje oblaka točaka.



Slika 4. Fotogrametrija interijera i 3D skeniranje [6]

Kao što je već rečeno 3D oblak točaka može se dobiti skeniranjem prostora oko sebe pomoću LiDAR skenera ili nizom fotografija s određenim međusobnim preklapanjem. Iz njih se pomoću metode pokreta iz slike generira oblak točaka kojim se može dočarati 3D prikaz objekta. Svaka točka oblaka je prostorno određena i može joj se dodijeliti boja koja predstavlja određenu vrijednost ili može imati boju objekta koji predstavlja pa se u tom slučaju govori o boji teksture.



Slika 5. Oblak točaka dobiven isključivo iz fotografija iz zraka [10]

Na slici 5. prikazana je građevina trodimenzionalno kao i objekti oko nje. Nedostatak metode rekonstrukcije isključivo iz fotografija iz zraka je taj što se sve što se nalazi ispod prve plohe vidljive iz zraka nalazi u sjeni te plohe i ne može se kvalitetno prikazati pa se iz tog razloga ova metoda kombinira s 3D skeniranjem kako bi se dobio cjelokupni 3D prikaz koji je puno kvalitetniji i detaljniji kao što je prikazano na slici 6.



Slika 6. Cjelokupni 3D prikaz [10]

### 3. Problemi i razvoj kompresije oblaka točaka

Zbog sve većeg zanimanja za upotrebom oblaka točaka stvaraju se izazovi povezani s pohranom i prijenosom podataka iz oblaka točaka. Raznolike tehnologije kompresije prijeko su potrebne kako bi podržale razne zahtjeve. Na primjer oblak točaka može biti oskudan ili gust pa onda i algoritam kompresije moramo uskladiti s time. Također može biti statičan ili dinamičan što predstavlja njegovu varijabilnost tijekom vremena. Kompresija može biti i s gubitcima i bez gubitaka ovisno mogu li se informacije u potpunosti povratiti ili ne.

Napretkom tehnologije i razvojem računalnih tehnika možemo obraditi oblake točaka veće gustoće i raznolikosti. Unatoč tome još uvijek je veoma izazovno učinkovito prenositi, vizualizirati i pohranjivati objekte velikih razmjera ili sadržaj visoke kvalitete posebno kad su takvi oblaci točaka dinamični. Osim toga razne funkcionalnosti kao što su razina detalja LOD (engl. *Level of Details*), slučajni pristup, regija interesa – ROI (engl. *Region of Interest*) vrlo tražene kako bi se ostvarila realizacija vizualizacije 3D oblaka točaka u stvarnom vremenu

Posljednjih par godina se povećava interes za razvoj učinkovitih PCC (engl. *Point Cloud Compression*) metoda. 3D PCC standardi se razvijaju od Međunarodne organizacije za standardizaciju – ISO (engl. *International Organization for Standardization*) / Međunarodne elektrotehničke komisije - IEC (engl. *International Electrotechnical Commission*). Ovaj standard uključuje 2 pristupa: V-PCC i G-PCC.

V-PCC je alatni paket metoda kompresije temeljenih na videozapisima i ima za cilj pružiti mogućnosti dekodiranja niske složenosti za aplikacije koje zahtijevaju dekodiranje u stvarnom vremenu kao što su virtualna i proširena stvarnost. S druge strane smatra se da G-PCC pruža učinkovitu kompresiju bez gubitaka i kompresiju s gubitcima za uvođenje autonomne vožnje, 3D karata i drugih aplikacija koje istražuju LiDAR generirane oblake točaka.

Nakon više godina eksperimentiranja s različitim metodama MPEG je odabrao dva pristupa temeljena na projekciji i octree-u i optimizirao ih kroz niz eksperimenata. Prednost tradicionalnih metoda je u tome što mogu komprimirati 3D oblak točaka na općenitiji način, a istovremeno pružiti obećavajuće performanse. S napretkom tehnika akvizicije 3D scene objekti će se snimati brže i lakše. Prikaz 3D oblaka točaka može postati slika/videozapis sljedeće generacije s mnogo gušćom razlučivošću.

Budući da su tehnike dubokog učenja – DL (engl. *Deep Learning*) nedavno primijenjene na kompresiju 3D oblaka točaka nije predloženo mnogo radova za kompresiju atributa. Atributi kao što su boja, refleksija itd. čvrsto su povezani geometrijskim informacijama pa je potrebno dizajnirati zajednički proces učenja kako bi se izbjeglo pucanje veze između geometrije i informacije o

atributima. Slično tome, većina trenutnih metoda temeljenih na DL-u dizajnirana je za komprimiranje jednog okvira 3D oblaka točaka. Pružanje boljih performansi kompresije u geometrijskim informacijama također zadržava potencijal znatnog smanjenja vremenske redundancije između oblaka dinamičkih točaka.

## 4. Skupovi podataka najčešćih oblaka točaka i 3D uzoraka

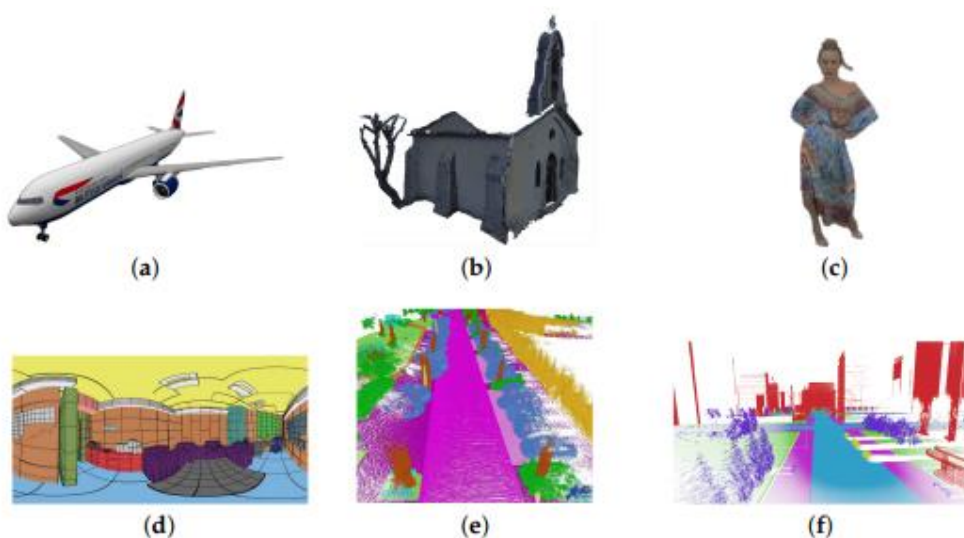
U ovome dijelu ukratko će se predstaviti skupove podataka najpopularnijih oblaka točaka i 3D uzoraka koji se predloženi za korištenje zajedno s raznim zadacima dubokog učenja. Ti skupovi predstavljaju najsuvremenija mjerila kojima se ocjenjuje i uspoređuje većina algoritama, a oni su sljedeći:

- ShapeNet je vrlo veliko spremište raznih oblika koje razumije mnogo različitih semantičkih klasa organiziranih u okviru WordNet-a, primjer kao na slici 7. a). Ti su oblici prikupljeni iz različitih repozitorija 3D CAD modela i označeni tako da rezultirajući modeli mogu riješiti različite zadatke. Veliki broj oznaka koje se nalaze u ovom skupu podataka čine ga savršenim kandidatom za mnoge različite zadatke kao što su klasifikacija, segmentacija dijelova, generiranje podataka, i rekonstrukcija.
- ModelNet je slično kao i ShapeNet spremište CAD 3D modela koji se mogu podijeliti u 40 klasa oblika objekata. Također u ovom skupu podataka, modeli su sintetički generirani i stoga, iako bez artefakata, nisu baš vjerni u odnosu na stvarne podatke. Većinu vremena ova dva skupa podataka koriste se zajedno koristeći Shapenet kao podatke o treningu i ModelNet kao modele testiranja.
- MPEG – uvjeti ispitivanja MPEG-a uključuju skup oblaka referentnih točaka koji su stečeni različitim strategijama koje se protežu od senzora strukture pokreta do LiDAR senzora što se vidi na slici 7. b). Točnije, dinamičke sekvence nabavljene su s ToF i LiDAR sensorima, dok statični podaci uključuju i modele laserskih skenera i uzorkovanu rekonstrukciju iz više slika kamere. Modeli su organizirani u tri kategorije, uključujući statičke modele, dinamičke objekte i dinamičke akvizicije.
- 8i Voxelized Full Bodies - 8i Voxelized Full Bodies (8iVFB) sastoji se od sekvenci oblaka s četiri točke gdje je na primjeru cijelog tijela ljudskog subjekta snimljeno s 42RGB kamere konfigurirane u 14 klastera pri 30 fps što je vidljivo na primjeru sa slike 7. c). Prostorne koordinate kvantificirane su s rezolucijama od 10 bita koje predstavljaju svaki model s kockom voksela  $1024 \times 1024 \times 1024$ ; za svaki slijed kocka se skalira tako da je to najmanja kocka koja sadrži cijeli slijed.
- Stanford 3D skup podataka o unutarnjoj sceni (S3DIS) sadrži 6 velikih unutarnjih prostora s 27 soba što se vidi na primjeru na slici 7. d). Svaka točka u scenama označena je jednom od 13 semantičkih kategorija. S3DIS spada u kategoriju statičkih skupova podataka koje se



obično koriste za zadatke klasifikacije oblaka točaka, ali je dizajniran da bude prikladan čak i za semantiku segmentacije, segmentaciju instance i otkrivanje objekata. Njegovi glavni primjeri primjene uključuju robotiku, proširenu stvarnost i urbanističko planiranje.

- KITTI je jedan od najpopularnijih javno dostupnih skupova podataka za autonomnu vožnju. To je sekvencijalni skup podataka stečen sustavom autonomne vožnje za snimanje sekvenci LiDAR okvira s pokretnim točkom vidljivosti. Sustav se sastoji od LiDAR senzora, stereo kamere, GPS-a i jedinice za inercijalno mjerenje kako bi mu se omogućili različiti zadaci kao što su: stereo snimanje, vizualna odometrija, otkrivanje 3D objekata i 3D praćenje. Kao sekvencijalni skup podataka, KITTI sadrži nekoliko okvira, ali rijetkih točaka. Također budući da sa stajališta senzora slijede smjer ceste oprema za nabavu ugrađena je na vozilo, povezane su uzorkovane LiDAR točke s oznakom ceste koje se distribuiraju pod određenim kutovima, što se lako može predvidjeti iz znanja o postavkama sustava.
- SemanticKITTI je skup podataka izgrađen na velodyne sekvenci KITTI skupa podataka i pruža uz 3D podatke oznake semantičke segmentacije i panoptičke segmentacije kao što se vidi na primjeru na slici 7. e). Sadrži detaljne napomene s 28 klasa, 22 različite scene i jedan je od najvećih skupova javnih podataka za autonomnu vožnju.
- SynthCity je skup podataka oblaka od 367,9 milijuna točaka nabavljen mobilnim laserskim skeniranjem u simuliranom okruženju što se vidi na primjeru na slici 7. f). Svakoj točki dodjeljuje se oznaka iz jedne od 9 kategorija. Problem korištenja ovih vrsta skupova podataka je veliki jaz između sintetičkih i stvarnih scena. Prvi općenito mogu biti vrlo realni, ali im nedostaje točnosti u detaljima, čak i ako ih je izuzetno lako označiti i steći.[13]



Slika 7. Primjeri nekih skupova podataka: (a) ShapeNet, (b) MPEG, (c) 8iVFB, (d) S3DIS, (e) SemanticKITTI, (f) SynthCity [13]

## 5. MPEG 3D PCC standardi

MPEG je jedna od glavnih standardizacijskih grupa koje se bave multimedijom. Još od početka 1990-ih MPEG standardizira standardne kompresije za 3D grafičke objekte kao što su 3D mreže, animirani objekti i oblaci točaka. Njegov cilj je izgraditi otvoreni standard za kompaktno predstavljanje 3D oblaka točaka. MPEG je 2017. započeo svoju standardizaciju kompresije oblaka točaka (PCC). Identificirane su tri kategorije oblaka točaka: kategorija 1 za statičke oblake točaka, kategorija 2 za dinamičke oblake točaka i kategorija 3 za LiDAR sekvence (dinamički dobiveni oblaci točaka). Nakon nekoliko godina intenzivnog razvoja događaja konačno su identificirane dvije tehnologije: G-PCC (PCC zasnovan na geometriji) za kategoriju 1 i kategoriju 3 i V-PCC (PCC zasnovan na videu) za kategoriju 2. V-PCC prihvaća pristup temeljen na projekciji koji omogućuje iskorištavanje visoko optimiziranih postojećih video kodera. Uglavnom se bavi vrlo gustim oblacima 3D točaka s ujednačenim uzorkovanjem i niskom razinom smetnji. G-PCC posvećen je upravljanju rijetkim 3D oblacima točaka te se temelji na čisto geometrijskom pristupu razgradnje temeljen na „octree“ strukturi podataka i koristan je za raznolike slučajeve upotrebe dizajnirane za rukovanje različitim oblacima točaka, od ljudskih modela do svjetskih, gradskih karata.

Kompresija trodimenzionalnih oblaka točaka fokusirana je na pronalazak i iskorištavanje višaka dobivenih tijekom snimanja: prostorni i geometrijski višak kod statičkih, a prostorni i vremenski višak kod dinamičkih oblaka točaka. Pohranjivanje podataka dobivenih pomoću oblaka točaka je težak i zahtjevan postupak pa se zbog toga koristi struktura podataka pod imenom „octree“ što je naprednija verzija „quadtree“ tehnologija korištenih u 2D. Kompresija oblaka točaka može se podijeliti na „intra“ i „inter frame“ kodere. „Intra frame“ koder provjerava samo jedan izolirani oblak točaka i komprimira ga iskorištavanjem prostornog i geometrijskog viška, za razliku od njega „inter frame“ predviđa trenutni oblak točaka baziran na prošlom oblaku iskorištavanjem vremenskih višaka.

Oblaci točaka se danas dosta koriste i uz to pružaju izuzetno velik broj informacija o snimljenom objektu pa su se u svrhu toga razvile specijalizirane „knjižnice“ pod nazivom PCL (engl. *Point Cloud Libraries*), a one sadrže razne algoritme za filtriranje, za predviđanje potrebnih veličina oblaka točaka, za rekonstrukciju površine, registraciju i segmentiranje. PCL je besplatni „open source“ program uspješno implementiran na razne platforme. Kako bi se olakšao njegov rad on se dijeli na skup manjih „knjižnica“ koje mogu istovremeno svaka za sebe raditi na različitim projektima. [10]

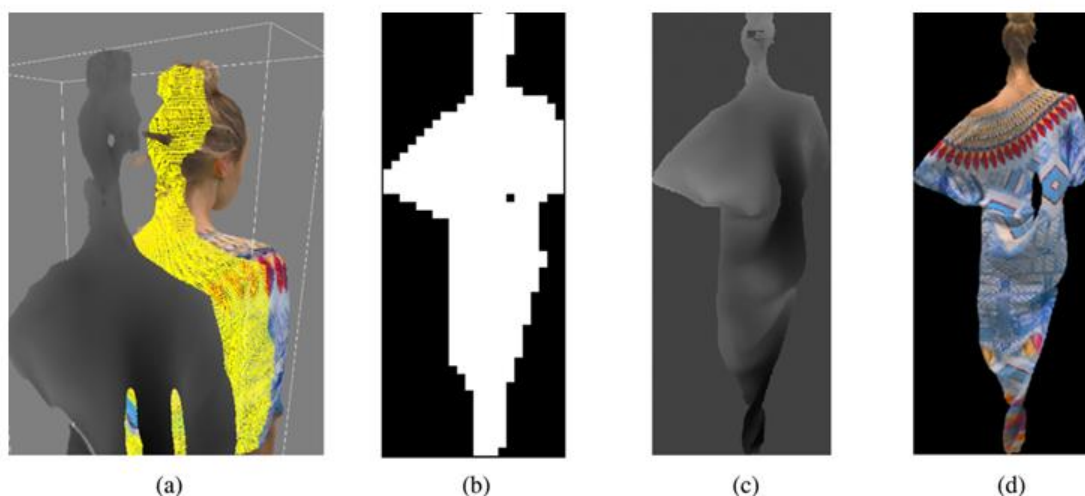
## 5.1 MPEG kompresija oblaka točaka temeljena na videu (V-PCC)

2D video kompresija uspješna je tehnologija lako dostupna zbog širokog prihvatanja standarda video kodiranja. Kako bi se iskoristila prednost ove tehnologije, V-PCC metode mogu pretvoriti 3D geometriju i podatke o atributima oblaka točaka u skup 2D zakrpa. Zakrpe se nakon toga mapiraju na unaprijed definirani skup 2D ravnina kroz ortogonalne projekcije kako bi se izbjeglo probleme s ponovnim uzorkovanjem i kako bi se omogućila kompresija bez gubitaka.

Projekcija 3D zakrpe na 2D zakrpu djeluje poput virtualne ortografske kamere hvatajući određeni dio oblaka točaka. U procesu projekcije oblaka točaka nekoliko virtualnih kamera registrira dijelove oblaka točaka i kombinira te slike kamere u sliku koja sadrži kolekciju projiciranih 2D zakrpa. Ovaj proces rezultira prikupljanjem metapodataka povezanih s projekcijom svake zakrpe.

Za svaki atlas postoje do tri pridružene slike: binarna slika nazvana mapa zauzetosti koja signalizira odgovara li piksel valjanoj 3D projiciranoj točki, geometrijska slika koja sadrži informaciju o dubini, tj. udaljenost između mjesta svake točke i ravnine projekcije, broj slika (atributa) kao što je tekstura (boja) svake točke, odnosno slika u boji koja sadrži R, G i B komponente informacija o boji. Za V-PCC slučaj, geometrijska slika je posebno slična karti dubine, budući da bilježi udaljenost između 3D točke i ravnine projekcije.

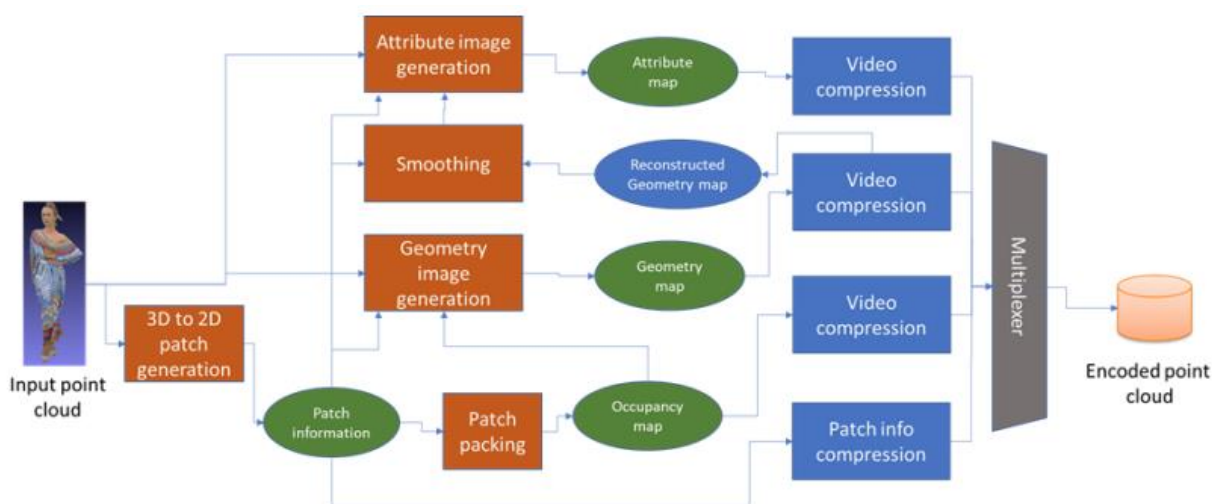
Na slici 8. je prikazan primjer 3D zakrpe i njene odgovarajuće projekcijske slike. Na slici je ilustrirana samo tekstura svake projicirane točke. Pristup temeljen na projekciji prikladan je za guste sekvence koje kada se projiciraju stvaraju neprekidne i glatke površine nalik na sliku. U rijetkim slučajevima za neke oblake točaka kodiranje temeljeno na projekciji možda neće biti učinkovito, te su za takve slučajeve prikladnije druge metode poput G-PCC-a.



Slika 8. (a) 3D zakrpa, (b) 3D karta zakrpe zauzetosti, (c) 3D slika geometrije zakrpe, (d) 3D slika teksture zakrpe [10]

### 5.1.1 V-PCC arhitektura

Slično standardima za video kodiranje, MPEG V-PCC specificira samo proces dekodiranja. Proces kodiranja je izvan opsega i prepušteno je proizvođaču koder da stvori najprikladniji koder koji generira bitstreamove u skladu s V-PCC specifikacijom. Arhitektura V-PCC koder ilustrirana je na slici 9.



Slika 9. Dijagram V-PCC koder [11]

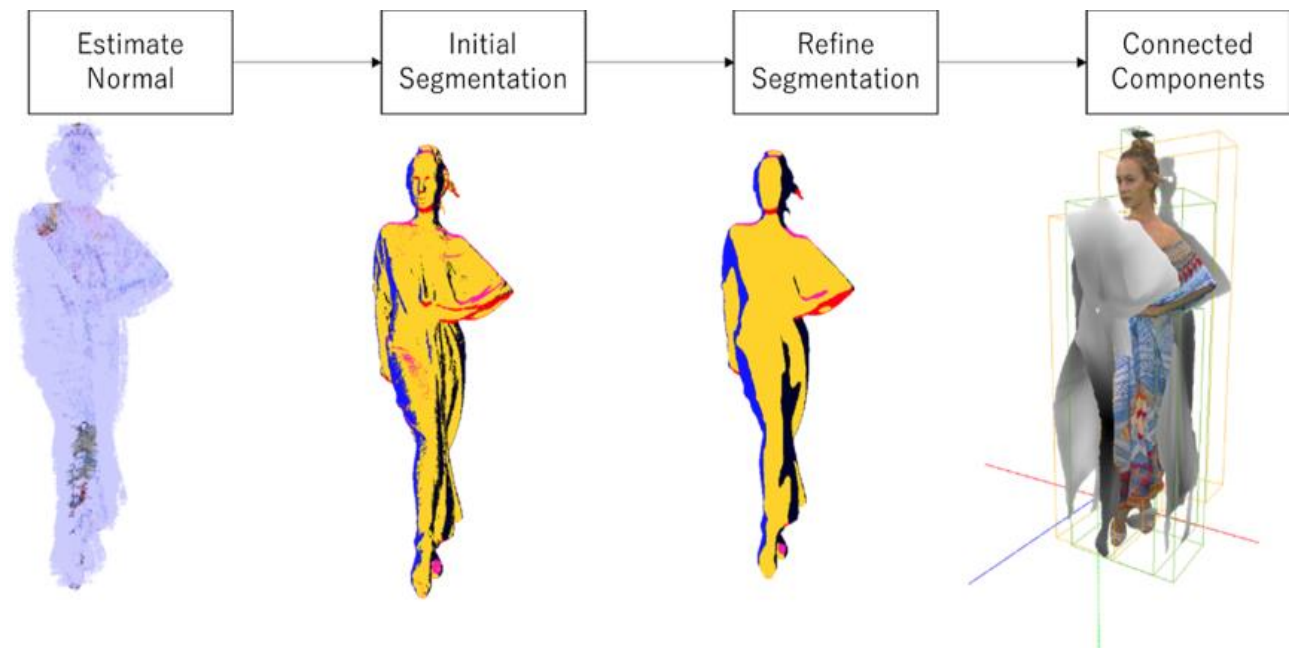
Tijekom standardizacije V-PCC-a razvijen je referentni softver TMC2 (engl. *Test Compression Model*) kako bi se izmjerila izvedba kodiranja i usporedile tehničke vrijednosti i prednosti predloženih metoda kodiranja oblaka točaka koje V-PCC razmatra. TMC2 koder primjenjuje nekoliko tehnika za poboljšanje performansi kodiranja, uključujući kako izvršiti pakiranje zakrpa, stvaranje mapa zauzetosti, geometrije, slika teksture i kompresiju mape zauzetosti, geometrije, teksture i informacija o zakrpama. Treba spomenuti da TMC2 koristi HEVC (engl. *High Efficiency Video Coding*) za kodiranje generiranih 2D videozapisa, ali korištenje HEVC-a nije obavezno i može se upotrebljavati bilo koji odgovarajući slikovni video koder.

### 5.1.2 Generiranje zakrpa

V-PCC razgrađuje ulazni oblak točaka u skup zakrpa koje se dalje mogu samostalno mapirati kroz jednostavnu ortogonalnu projekciju na 2D rešetku bez međusobnih sudaranja i bez potrebe za ponovnim uzorkovanjem geometrije oblaka točaka. Nadalje, proces stvaranja zakrpa ima za cilj generirati zakrpe s glatkim granicama minimizirajući njihov broj i izobličenja tijekom mapiranja.

U nastavku će se opisati kako generirati 3D zakrpe u TMC2 koderu. Prvo se procjenjuje normala za svaku točku. S obzirom na šest ortografskih pravaca projekcije svaka se točka pridružuje

smjeru projekcije koji daje najveći umnožak između normale točke i odabire se odgovarajući smjer projekcije. Klasifikacija točaka dalje se obrađuje prema klasifikaciji smjera projekcije susjednih točaka. Nakon što je klasifikacija točaka dovršena točke s istim smjerovima projekcije grupiraju se zajedno pomoću algoritma za povezivanje komponenti. Kao rezultat tog procesa svaka povezana komponenta predstavlja 3D zakrpu. 3D točke zakrpa se zatim projiciraju pravokutno na jedan od šest pravaca graničnog okvira poravnatog s osi na temelju pridruženog smjera projekcije 3D zakrpe. Na slici 10. se može vidjeti kako površina projekcije leži na graničnom okviru koji je poravnat s osi.



Slika 10. Generiranje zakrpa [11]

Budući da 3D zakrpa može imati više točaka projiciranih na istu lokaciju piksela TMC2 koristi nekoliko „karti“ za pohranjivanje tih preklapajućih točaka. Ako pretpostavimo skup točaka  $H$  koje se projiciraju na isto mjesto prema tome TMC2 može npr. odlučiti koristiti dvije karte: kartu za blizinu i daleku kartu. Karta za blizinu pohranjuje točku iz  $H$  s najnižom vrijednošću dubine  $D_0$ , dok daleka karta pohranjuje točku s najvećom vrijednošću dubine unutar definiranog intervala  $(D_0, D_0+D)$ . Korisnički definirana veličina  $D$  predstavlja debljinu površine i može se odrediti koderom i koristiti za poboljšanje kodiranja i rekonstrukciju geometrije.

TMC2 također ima opciju da kodira daleku kartu na 2 načina: kao diferencijalnu kartu u odnosu na kartu blazine u dva odvojena video toka ili da privremeno ispreplete daleku kartu i kartu blazine u jedan tok i kodira apsolutne vrijednosti. U oba slučaja veličina intervala  $D$  može se koristiti za poboljšanje procesa rekonstrukcije dubine. Druga karta se može u potpunosti odbaciti, a koristi

se samo shema interpolacije za generiranje udaljene karte. Nadalje, oba okvira mogu biti poduzorkovani i prostorno isprepletena kako bi se poboljšala kvaliteta rekonstrukcije u usporedbi s rekonstrukcijom s jednom jedinom kartom i učinkovitost kodiranja u usporedbi sa slanjem obje karte.

Daleka karta potencijalno može nositi informacije o više elemenata mapiranih na istu lokaciju. U načinu rada bez gubitaka TMC2 modificira udaljenu sliku karte tako da umjesto kodiranja jedne točke po uzorku više točaka se može kodirati koristeći poboljšani delta-dubinski EDD (engl. *Enhanced Delta-Depth*) kod. Unatoč tome još uvijek je moguće da neke točke možda nedostaju, a korištenje samo šest smjerova projekcije može ograničiti kvalitetu rekonstrukcije oblaka točaka. Kako bi se poboljšala rekonstrukcija proizvoljno orijentiranih površina standardu je dodano 12 novih načina koji odgovaraju kamerama postavljenim pod kutom od 45 stupnjeva. Rotacija oblaka točaka do 45 stupnjeva može se modelirati samo korištenjem cjelobrojnih operacija, čime se izbjegavaju problemi ponovnog uzorkovanja uzrokovani rotacijama.

Za generiranje slika prikladnih za video kodiranje TMC2 dodao je opciju operacija filtriranja. Npr. TMC2 definira veličinu bloka  $T \times T$  unutar koje vrijednosti dubine ne mogu imati veliku varijancu, a točke s vrijednostima dubine iznad određenog praga uklanjaju se iz 3D zakrpe. Također TMC2 definira raspon koji predstavlja dubinu, a ako je vrijednost dubine veća od dopuštenog raspona, uklanja se iz 3D zakrpe. Točke uklonjene iz 3D zakrpe kasnije se analiziraju algoritmom povezanih komponenti pa se mogu koristiti za generiranje nove 3D zakrpe. Daljnja poboljšanja u generiranju 3D zakrpa uključuju korištenja informaciju o boji za segmentaciju ravnine i biranje najbolje ravnine za projekciju dubine.

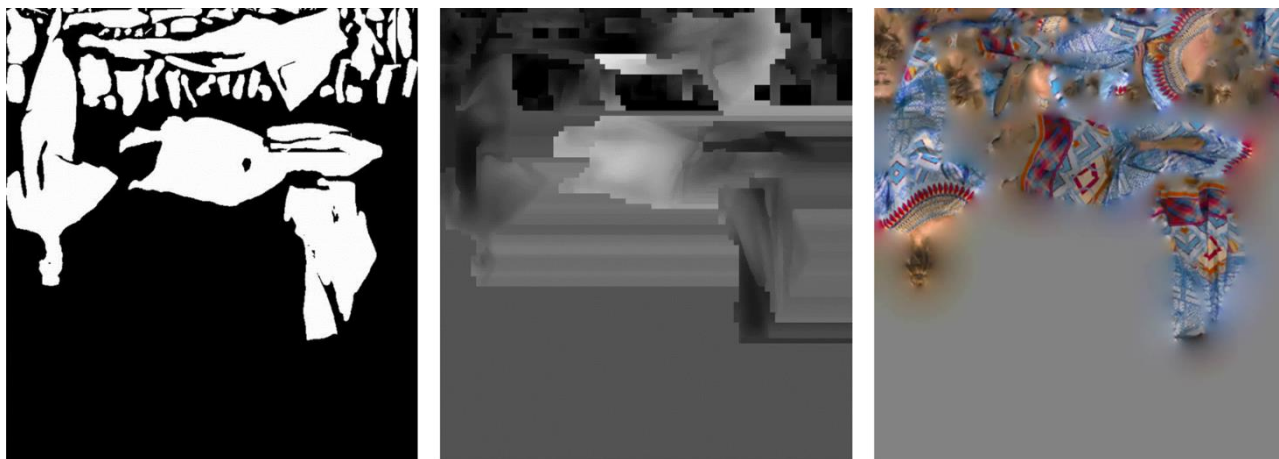
TMC2 također ima opciju ograničavanja minimalnog broja točaka u 3D zakrpi kao korisnički definiranog parametra. Broj generiranih zakrpa uvelike ovisi o složenosti oblaka točaka. Obično je za hvatanje više od 95% točaka potrebno manje od sto zakrpa. Zbog toga neće dopustiti stvaranje 3D zakrpe koja može sadržavati manji broj točaka od navedenog minimalnog broja točaka. Preostale točke obično su izolirane točke ili izolirane skupine točaka koje bi zahtijevale veliki broj zakrpa za hvatanje. Ovisno o ograničenjima primjene, koder može odlučiti odbaciti preostale točke uz rizik uvođenja rupa i pukotina u točkastom oblaku. Za kodiranja s gubitcima TMC2 može odlučiti jednostavno ignorirati takve točke, a za kodiranja bez gubitaka točke koje nisu uključene ni u jednu zakrpu mogu se kodirati dodatnim zakrpama za kodiranje bez gubitaka.

V-PCC omogućuje kontrolu minimalnog broja točaka po zakrpi omogućujući korisniku da filtrira izolirane točke ili male skupine izoliranih točaka koje mogu imati ograničen utjecaj na vizualnu kvalitetu. Koder također može odlučiti pohraniti sve ili podskup preostalih točaka u posebne zakrpe, nazvane neiskorištene ili neovisne zakrpe točaka. Ove dodatne zakrpe još su poznate i kao

pomoćne zakrpe te pohranjuju podatke izravno u geometrijsku sliku stvorenu za obične zakrpe, za 2D video kodiranje. Pomoćne se zakrpe mogu pohraniti u zasebnu sliku i kodirati kao poboljšana slika. Mogu se pohraniti u istim video sekvencama kao i uobičajene zakrpe.

### 5.1.3 Pakiranje zakrpa

Način pakiranja različitih zakrpa dobivenih projekcijom u 2D okviru također nije dio standarda, ali je tijekom razvoja standarda eksperimentirano s nekoliko metoda. Pakiranje zakrpa odnosi se na postavljanje projiciranih 2D zakrpa u 2D sliku veličine  $\check{S} * V$  (širina \* visina). Ovo je iterativni proces gdje se prvo zakrpe poredaju po veličini. Zatim se lokacija svake zakrpe određuje iscrpnim pretraživanjem u redosljedu rasterskog skeniranja, a odabire se prva lokacija koja garantira umetanje bez preklapanja. Kako bi se mogućnosti ugradnje dodatno poboljšale dopušteno je osam različitih orijentacija zakrpa. Blokovi koji sadrže piksele s važećim vrijednostima dubine koji pripadaju području i pokriveni su veličinom zakrpe smatraju se zauzetim blokovima i ne mogu ih koristiti druge zakrpe garantirajući na taj način da je svaki blok povezan samo jednom jedinstvenom zakrpom. U slučaju da nema slobodnog prostora za sljedeću zakrpu visina  $V$  slike se udvostručuje i ponovno se procjenjuje umetanje ove zakrpe. Nakon umetanja svih zakrpa konačna visina se skraćuje na minimalnu potrebnu vrijednost. Slika 11. prikazuje primjer mape popunjenosti pakiranih zakrpa, geometriju i slike tekstone.



Slika 11. Primjer pakiranja zakrpa [11]

Kako bi se poboljšala učinkovitost kompresije zakrpe sa sličnim sadržajem trebale bi biti postavljene na slične položaje tijekom vremena. Za generiranje vremenski dosljednog pakiranja TMC2 pronalazi podudaranja između zakrpa različitih okvira i pokušava umetnuti zakrpe koje se podudaraju na sličnu lokaciju. TMC2 koristi istu orijentaciju zakrpe i normalni smjer za zakrpe koje

se podudaraju, a samo različito kodira informacije o poziciji zakrpe i graničnom okviru. Informacije o podudaranju zakrpa mogu se koristiti za postavljanje zakrpa na dosljedan način kroz vrijeme. Prostor u atlasu je unaprijed dodijeljen kako bi se izbjeglo preklapanje zakrpa i zajamčila vremenska dosljednost. Daljnje optimizacije lokacija zakrpa mogu uzeti u obzir razne aspekte a ne samo izvedbu kompresije. Npr. postupak postavljanja zakrpa je modificiran kako bi se omogućilo dekodiranje s malim kašnjenjem tj. dekoder ne treba čekati kako bi dekodirao informacije svih zakrpa kako bi izvukao blok do informacija sa zakrpama iz projicirane 2D slike i može trenutačno odrediti koji pikseli pripadaju kojima zakrpama dok se dekodiraju.

#### **5.1.4 Geometrija i mapa zauzetosti**

Za razliku od geometrijskih mapa koje pohranjuju vrijednost u tri različita kanala geometrijske mape u V-PCC-u pohranjuju udaljenost između nedostajućih koordinata 3D položaja točaka i površine projekcije u 3D graničnom okviru koristeći samo kanal svjetline (engl. *Luminance Channel*) video sekvence. Budući da zakrpe mogu imati proizvoljne oblike neki pikseli mogu ostati prazni nakon pakiranja zakrpa. Kako bi razlikovao piksele u geometrijskom videu koji se koriste za 3D rekonstrukciju i neiskorištene piksele TMC2 šalje mapu popunjenosti povezanu sa svakim okvirom oblaka točaka. Karta popunjenosti ima korisnički definiranu preciznost blokova.

Mapa popunjenosti je binarna slika koja je kodirana pomoću video koda bez gubitaka i definira važnost informacija u 2D slikama sa teksturom i dubinom. Vrijednost 1 označava da postoji najmanje 1 važeći piksel u odgovarajućem bloku dok vrijednost 0 označava prazno područje koje je ispunjeno pikselima iz postupka popunjavanja slike. Karta popunjenosti ima korisnički definiranu preciznost  $B \times B$  blokova, gdje je za kodiranje bez gubitaka  $B = 1$ , a za kodiranje s gubicima obično se koristi  $B = 4$ , uz vizualno prihvatljivu kvalitetu, uz znatno smanjenje broja bitova potrebnih za kodiranje mape popunjenosti. Za video kodiranje binarna slika dimenzija (Š/B, V/B) pakira se u kanal svjetline i kodira se bez gubitaka, ali također je moguće kodiranje s gubicima mape zauzetosti. Za slučaj s gubicima kad je rezolucija slike smanjena mapa zauzetosti slike treba biti namještena na određenu nominalnu rezoluciju. Proces povećanja rezolucije može dovesti do dodatnih nepravilnosti u mapi zauzetosti što ima za efekt dodavanje točaka u rekonstruirani oblak točaka.

#### **5.1.5 Ispuna slike, širenje grupe, ponovno bojanje i kompresija videa**

Za geometrijske slike TMC2 popunjava prazni prostor između zakrpa koristeći funkciju podmetanja koja ima za cilj generirati glatku sliku prikladnu za veću učinkovitost video kompresije. Svaki blok piksela obrađuje se zasebno. Ako je blok prazan tj. nema valjanih vrijednosti dubine pikseli bloka se popunjavaju kopiranjem zadnjeg reda ili stupca prethodnog bloka u rasterskom

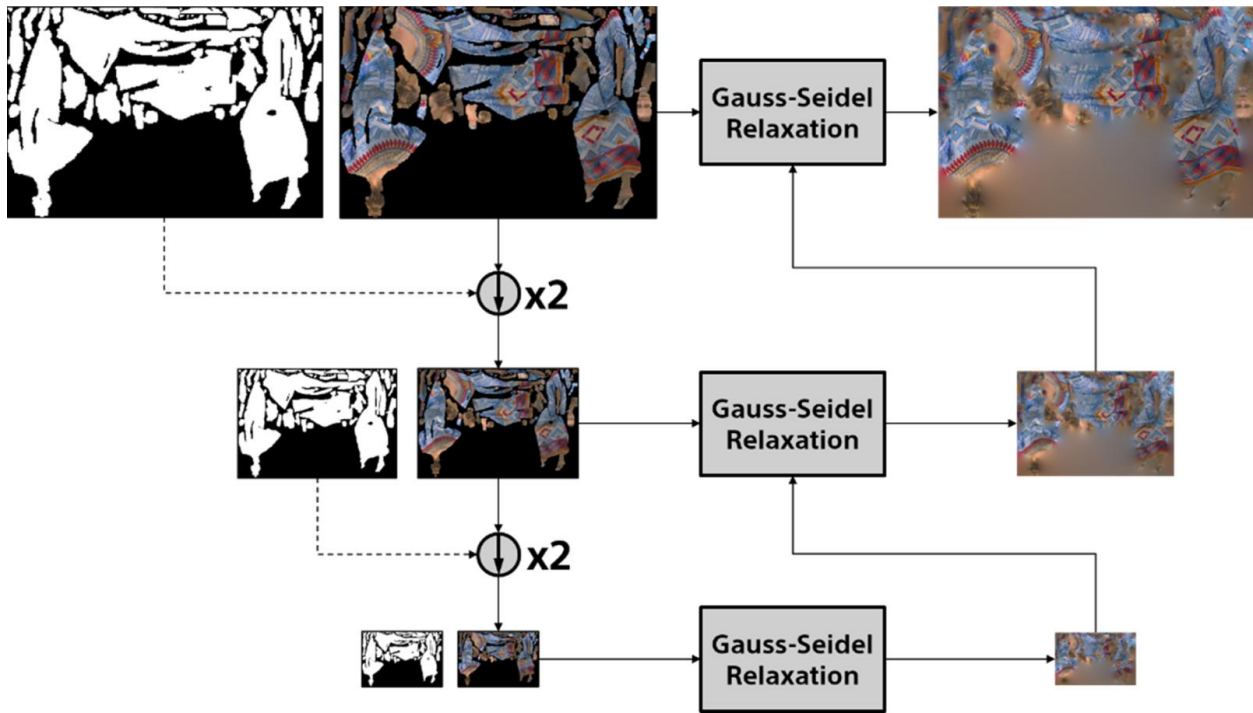


redosljednu skeniranja. Ako je blok pun tj. svi pikseli imaju važeće vrijednosti dubine ispuna nije potrebna. Ako blok ima i važeće i nevažeće piksele tada se prazne pozicije iterativno popunjavaju prosječnom vrijednošću svojih popunjenih susjeda. Postupak popunjavanja također je poznat kao dilatacija geometrije, izvodi se neovisno za svaki okvir. Međutim, prazni položaji za karte blizu i daleko su isti i korištenje sličnih vrijednosti može poboljšati učinkovitost kompresije. Stoga se provodi grupna dilatacija, gdje se izračunava srednja vrijednost praznih područja, a ista se vrijednost koristi za obje mape.

Uobičajeni neobrađeni format koji se koristi kao ulaz za video kodiranje je YUV420 s 8-bitnom svjetlinom i poduzorkovanim kanalima boje. TMC2 pakira geometrijsku sliku samo u kanal svjetline budući da geometrija predstavljena udaljenošću ima samo jednu komponentu. Također, pažljivo razmatranje GOP (engl. *Group of Pictures*) strukture kodiranja može dovesti do značajne učinkovitosti kompresije. U slučaju kodiranja bez gubitaka koordinate točaka koje se ne mogu prikazati u regularnim zakrpama isprepliću se blokovima i izravno pohranjuju u svjetlosni kanal. 10 bitni profili kao oni u HEVC enkoderima mogu poboljšati preciznost i performanse kodiranja. Dodatno savjeti za smještaj zacrpa u atlas mogu se dati enkoderu da poboljša proces procjene kretanja.

Budući da se rekonstruirana geometrija može razlikovati od izvorne TMC2 prenosi boju iz izvornog oblaka točaka u dekodirani oblak točaka i koristi vrijednosti ove nove boje za prijenos. Postupak ponovnog bojenja razmatra vrijednost boje najbliže točke iz izvornog oblaka točaka kao i susjedstvo točaka koje su bliže rekonstruiranoj točki kako bi se odredila što bolja moguća vrijednost boje.

Nakon što su poznate vrijednosti boja TMC2 preslikava boju iz 3D u 2D koristeći isto preslikavanje koje je primijenjeno na geometriju. Kako bi ispunio sliku u boji TMC2 može koristiti proceduru koja se temelji na interpolaciji mip-mapa i kako bi još dodatno poboljšao ispunu korištenjem prorijeđenog linearnog optimizacijskog modela kao što je prikazano na slici 12.



Slika 12. Ispuna tekture mip mapa slike s rijetkom linearnom optimizacijom [11]

Mip mape su unaprijed izračunate optimizirane sekvence slika, od kojih je svaka prikaz progresivno niže rezolucije od prethodne. Interpolacija mip-mapa stvara više-rezolucijski prikaz tekture slike vođene mapom zauzetosti, čuvajući aktivne piksele čak i kad su oni popunjeni s praznim pikselima. Rijetka linearna optimizacija temeljena na Gauss-Seidelovoj metodi može se koristiti za popunjavanje praznih piksela na svakoj skali rezolucije. Manje rezolucije onda se koriste kao početne vrijednosti za optimizaciju slika s povećanim uzorkovanjem na višoj razini. Na taj način pozadina je glatko ispunjena vrijednostima koje su slične rubovima zakrpa. Isti princip usrednjavanja pozicija koje su prije bile prazna područja mape mogu se koristiti za atribute također. Niz podstavljenih slika zatim se pretvara u boji iz RGB444 u YUV420 i kodira tradicionalnim video koderima.

### 5.1.6 Izgladivanje geometrije i atributa

Proces rekonstrukcije koristi dekodirane bitstreamove za mapu popunjenosti, geometriju i slike atributa za rekonstrukciju 3D oblaka točaka. Kada TMC2 koristi dvije mape, blisku i daleku mapu, a vrijednosti iz dviju slika dubine su iste, TMC2 može generirati nekoliko duplih točaka. To može utjecati na kvalitetu, kao i na stvaranje neželjenih točaka za kodiranje bez gubitaka. Kako bi se prevladao ovaj problem proces rekonstrukcije je modificiran kako bi se stvorila samo jedna točka po koordinati zakrpe kada su koordinate pohranjene u bliskoj i dalekoj mapi jednake učinkovito skraćujući duplikate točaka.

Kompresija geometrije i atributnih slika te dodatne točke uvedene zbog poduzorkovanja karte zauzetosti mogu uvesti artefakte koji bi mogli utjecati na rekonstruirani oblak točaka. TMC2 može koristiti tehnike za poboljšanje lokalne kvalitete rekonstrukcije. Slične metode naknadne obrade mogu signalizirati i izvršiti na strani dekodera. Na primjer, kako bi se smanjili mogući geometrijski artefakti uzrokovani segmentacijom TMC2 može izgladiti točke na granicama zakrpa koristeći proces poznat kao 3D geometrijsko izgladivanje. Jedna potencijalna metoda kandidata za izgladivanje oblaka točaka identificira točke na rubovima zakrpe i izračunava težište dekodiranih točaka u maloj 3D mreži. Nakon što se izvede težište i broj točaka u mreži  $2 \times 2 \times 2$ , primjenjuje se uobičajeno korišteni trilinearni filter.

Zbog procesa segmentacije oblaka točaka i dodjele zakrpa koji mogu postaviti zakrpe s različitim atributima jedne blizu drugih na slikama vrijednosti boja na granicama zakrpa također mogu biti sklone vizualnim artefaktima. Nadalje artefakti blokovskog kodiranja mogu stvoriti uzorke rekonstrukcije koji nisu glatki preko granica zakrpa stvarajući vidljive artefakte spojeva u rekonstruiranom oblaku točaka. TMC2 ima opciju izvođenja izgladivanja kako bi se smanjio učinak spojeva na rekonstruirani oblak točaka. Osim toga izgladivanje atributa izvodi se u 3D prostoru kako bi se iskoristilo ispravno susjedstvo pri procjeni nove izgladene vrijednosti.

### 5.1.7 Kompresija atlasa

Mnoge od prije objašnjenih tehnika imaju nenormativne aspekte što znači da se izvode samo kod koda, a dekodir ih ne mora biti svjestan. Međutim, nekoliko aspekata, posebno povezanih s metapodacima koji se koriste za rekonstrukciju 3D podataka iz 2D videopodataka normativni su i trebaju se poslati dekoderu za pravilnu obradu. Npr. položaj i orijentacija zakrpa, veličina bloka korištena u pakiranju i drugi su metapodaci atlasa koji se trebaju prenijeti u dekodir. V-PCC standard definira tok metapodataka atlasa koji se temelji na NAL (engl. *Network Abstraction Layer*) jedinicama gdje se prenose informacije o zakrpi podataka. Struktura jedinice NAL slična je strukturi bitstreama koja se koristi u HEVC što omogućuje veću fleksibilnost kodiranja. Opis normativnih elemenata sintakse kao i objašnjenje tehnika kodiranja za tok metapodataka kao što je međupredviđanje podataka zakrpa može se pronaći u trenutnoj standardnoj specifikaciji.

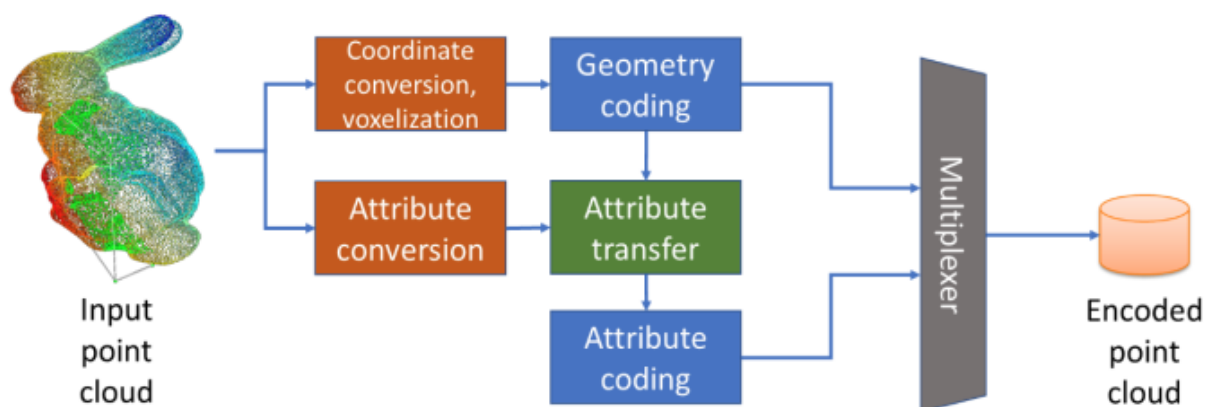
## 5.2 MPEG kompresija točaka oblaka temeljena na geometriji (G-PCC)

Dok se V-PCC pristup kodiranju temelji na 3D u 2D projekcijama, G-PCC za razliku od njega kodira sadržaj izravno u 3D prostoru. Kako bi se to postiglo G-PCC koristi strukture podataka kao što je „octree“ koje opisuje lokacije točaka u 3D prostoru. Nadalje, G-PCC ne pretpostavlja koordinatni prikaz ulaznog oblaka točaka. Točke imaju unutarnju vrijednost temeljenu na cijelom broju pretvorenu iz prikaza vrijednosti s pomičnim zarezom. Ova je konverzija konceptualno slična vokselizaciji ulaznog oblaka točaka, a može se postići skaliranjem, prevođenjem i zaokruživanjem.

Drugi ključni koncept za G-PCC je definicija pločica i rezova kako bi se omogućila funkcionalnost paralelnog kodiranja. U G-PCC-u isječak je definiran kao skup točaka (geometrija i atributi) koje se mogu neovisno kodirati i dekodirati. Sa svoje strane pločica je skupina odsječaka s informacijama o graničnom okviru. Pločica se može preklapati s drugom pločicom, a dekodirer može dekodirati djelomično područje oblaka točaka pristupom određenim odsječcima. Jedno ograničenje trenutnog G-PCC standarda je da je definiran samo za *intra* predviđanje, tj. trenutno ne koristi nikakav alat za vremensko predviđanje. Unatoč tome tehnike temeljene na procjeni kretanja oblaka točaka i međupredviđanju razmatraju se za sljedeću verziju standarda.

### 5.2.1 G-PCC arhitektura

Na slici 13. prikazan je blok dijagram koji prikazuje G-PCC referentni koder također poznat kao TMC13 kojeg će se detaljnije opisati u nastavku. Nije zamišljen za predstavljanje kompletnog skupa funkcionalnosti TMC13, već samo nekih od njegovih osnovnih modula.



Slika 13. Dijagram G-PCC referentnog koda [2]

G-PCC kodiranje je sekvencijalni proces koji najprije obrađuje geometriju, a zatim attribute. Ovo ponašanje je korisno za kodiranje gubitaka, usmjereno je potrebom da se najprije mapiraju izvorni

atributi na rekonstruiranoj (kodiranoj i dekodiranoj) geometriji i kodiranju tih mapiranih atributa umjesto izvornih.

Točke izvorne geometrije mogu se predstaviti brojevima s pomičnim zarezom u svjetskom koordinatnom sustavu. Dakle, prvi korak kodiranja geometrije je izvođenje transformacije koordinata nakon čega slijedi vokselizacija. Drugi korak sastoji se od geometrijske analize korištenjem sheme “octree” ili “trisoup”.

Konačno rezultirajuća struktura je aritmetički kodirana. Što se tiče kodiranja atributa TMC13 podržava izbornu konverziju iz RGB u YCbCr. Nakon toga koristi se jedan od tri dostupna alata za transformaciju, RAHT (engl. *Region Adaptive Hierarchical Transform*), predikcijska transformacija (engl. *Prediction Transform*) i Lifting transformacija. Nakon transformacije koeficijenti se kvantiziraju i aritmetički kodiraju.

### 5.2.2 Koordinacija transformacije i vokselizacije

Koordinate ulazne točke predstavljene su u njihovom koordinatnom sustavu koji pruža uređaj za nabavu i općenito se izražavaju kao brojevi s pomičnim točkama. U takvim slučajevima prije kodiranja G-PCC-a provodi se pretvorba u cjelobrojne vrijednosti. Kako bi se normalizirao izvorni oblak točaka koordinatni sustav ulaznog svijeta pretvara se u takozvani FCS (engl. *Frame Coordinate System*) predstavljen sa  $N$  bitovima za svaku od 3 kartezijske dimenzije. Stoga je granična kutija izvornog oblaka točaka geometrijski poravnana s FCS-om uz očuvanje omjera slike granične kutije.

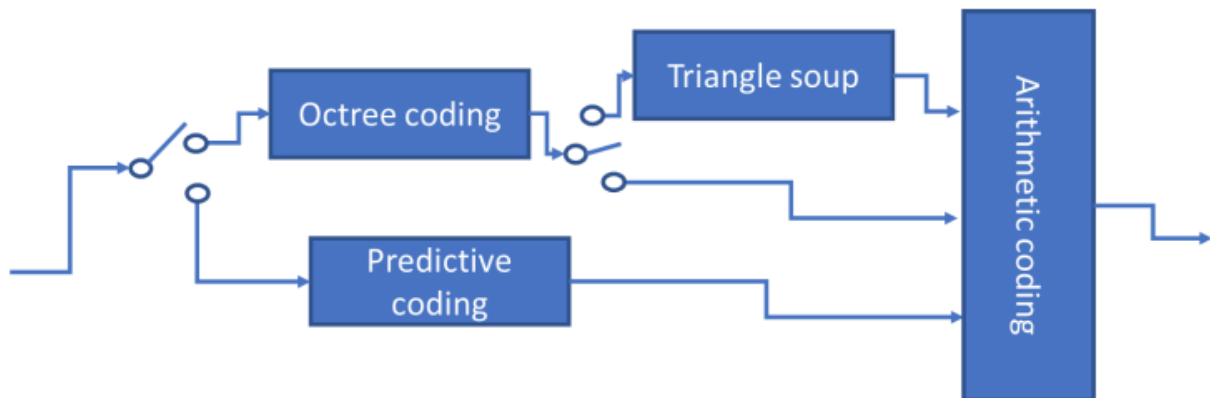
G-PCC koder obrađuje cjelobrojne vrijednosti, stoga je potrebna vokselizacija. Postupak kvantizacije primjenjuje se na točke predstavljene u FCS-u pretvarajući ulaznu dubinu bita iz  $N$ -bita (po dimenziji) u prikaz  $M$ -bita (gdje  $M < N$ ). Vrijednosti  $n$  ovise o preciznosti akvizicijskog sustava, a vrijednosti  $M$  o željenoj preciznosti rekonstrukcije.

Nakon primjene procesa kvantizacije vrlo je vjerojatno da se neke nove točke dupliciraju (imaju iste kvantizirane koordinatne vrijednosti). Takve se točke nazivaju dupliciranim točkama i mogu imati različite vrijednosti atributa jer izvorne nekvantizirane točke imaju različite vrijednosti atributa. Mogu se koristiti različite strategije za nadoknadu gubitka točnosti atributa zbog kvantizacije geometrije, a proces ponovnog bojanja uključen je u shemu kodiranja.

### 5.2.3 Kodiranje geometrije

G-PCC uključuje dva načina kodiranja geometrije. Prvi je korištenje „octree“ razgradnje 3D prostora i potencijalno se može nadopuniti alatom koji se naziva *trisoup* koji se koristi za prerano

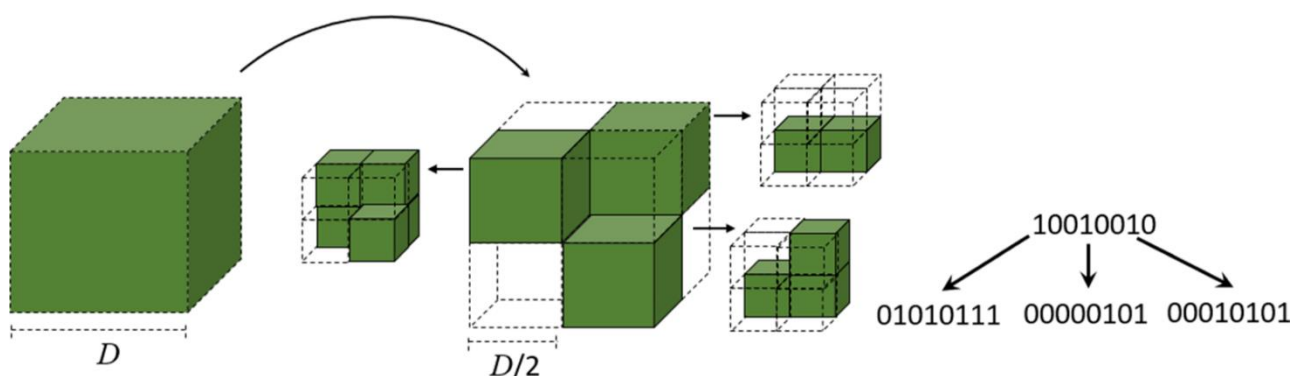
ukidanje razgradnje octree-a. Drugi je jednostavniji prediktivni način rada, gdje se točke uzastopno prelaze, a vrijednost trenutne točke predviđa se iz prethodnih (do 3 točke). Ovaj drugi način rada uglavnom je prikladan za objekte koje nabavljaju laseri kada se nalog za nabavu može koristiti za predviđanje vrijednosti koordinata točaka. Cjelokupna shema ovih načina kodiranja za geometriju prikazana je na slici 14.



Slika 14. Dijagram koda geometrije G-PCC-a [2]

#### 5.2.4 „Octree“ kodiranje

Vokselizirani oblak točaka predstavljen je pomoću „octree“ strukture na način bez gubitaka. Pretpostavimo da je oblak točaka sadržan u kvantiziranom volumenu  $D \times D \times D$  vokselu. U početku je volumen segmentiran okomito i vodoravno u osam podkocki s dimenzijama  $D/2 \times D/2 \times D/2$  vokselu, kao što je prikazano na slici 15.



Slika 15. Prva dva koraka u konstrukciji „octree“ procesa [11]

Ovaj se proces rekurzivno ponavlja za svaku zauzetu podkocku sve dok  $D$  ne bude jednak 1. Važno je istaknuti da je općenito samo 1% položaja vokselu zauzeto, što čini „octree“ strukturu vrlo prikladnom za predstavljanje geometrije oblaka točaka. U svakom koraku dekompozicije provjerava

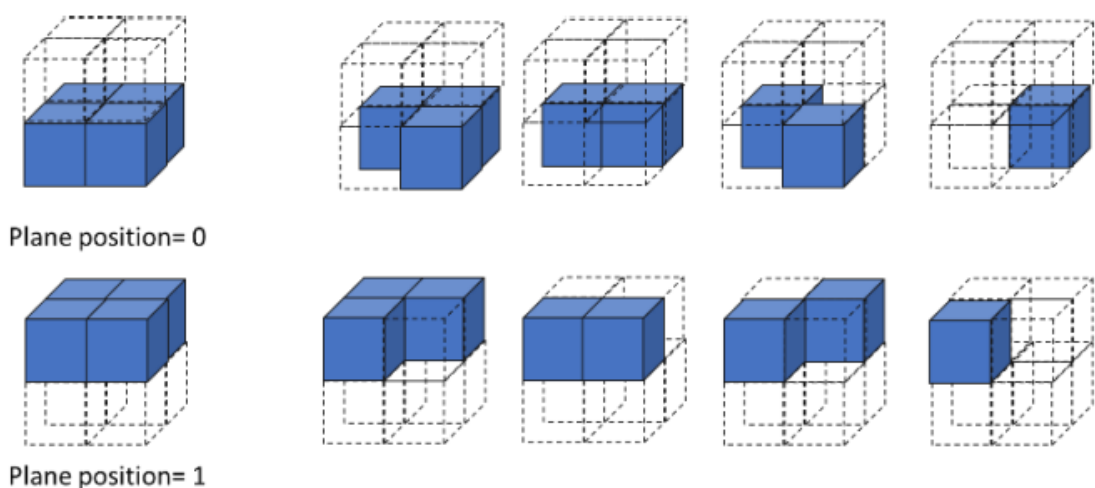
se koji su blokovi zauzeti, a koji nisu. Zauzeti blokovi su označeni kao 1, a nezauzeti blokovi su označeni kao 0. Okteti generirani tijekom ovog procesa predstavljaju stanje zauzetosti čvora okteta u 1-bajtnoj riječi i komprimirani su pomoću entropijskog kodera uzimajući u obzir korelaciju sa susjednim oktetima. Za kodiranje izoliranih točaka budući da unutar volumena nema drugih točaka s kojima bi se koreliralo uvodi se alternativna metoda za entropijsko kodiranje okteta, naime način izravnog kodiranja tj. DCM (engl. *Direct Coding Mode*). U DCM-u su koordinate točke izravno kodirane bez izvođenja ikakve kompresije. DCM način se zaključuje iz susjednih čvorova kako bi se izbjeglo signaliziranje upotrebe DCM-a za sve čvorove stabla.

Osim tradicionalne „octree“ razgradnje, G-PCC uključuje specifične alate za kodiranje za optimizaciju analize „octree-a“ : način izravnog kodiranja, planarni način rada, QTBT (engl. *Quad Tree/Binary Tree*) način rada i kutni / azimutalni način rada.

**Način izravnog kodiranja** - Razgradnja „octree-a“ djeluje učinkovito kada u određenoj kocki ima mnogo točaka. Kada objekt predstavlja izolirane točke koje se javljaju u slučaju rijetkih predmeta 8-bitni uzorak koji se koristi za opisivanje razine djeteta skuplji je od jednostavnog kodiranja izravnog vokselu koji sadrži izoliranu točku, mehanizam koji se naziva način izravnog kodiranja DCM. Stoga G-PCC iskorištava korelaciju informacija o čvoru nadređene razine kako bi utvrdio može li se DCM primijeniti. Točnije, postoje dva uvjeta za utvrđivanje prihvatljivosti DCM-a:

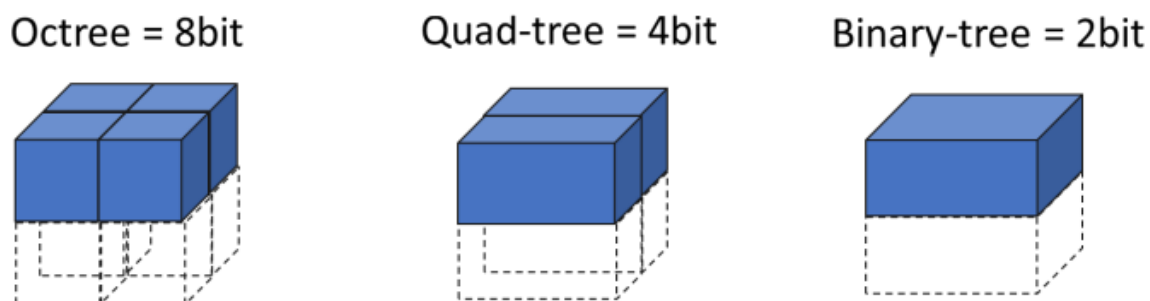
1. roditeljski čvor je jedino zauzeto dijete, a baka i djed je ili jedino okupirano dijete ili jedno od dvoje okupirane djece,
2. roditeljski čvor je jedino zauzeto dijete, a lica susjednih 6 kockica roditeljskog čvora nisu zauzeta.

**Planarni način rada i četverostruko/binarno stablo (QTBT)** -U slučaju rijetkih točkastih oblaka neoptimalno je predvidjeti koordinate od susjeda. Planarni način rada posvećen je točkama čak i kada su udaljene jedna od druge. Kada se takav slučaj utvrdi, smanjuje se broj mogućih slučajeva koji se koriste za predviđanje kao što je prikazano na slici. 16.



Slika 16. Primjer planarnog načina rada gdje su zauzeti čvorovi u istoj ravnini [2]

U planarnom načinu rada uzorak popunjenosti octree-a popunjen je korelacijom iz uzoraka susjedstva. Međutim, distribuciju je moguće izraziti ne pomoću octree-a, već koristeći „quad trees“ ili „binary trees“, što omogućuje smanjenje koda popunjenosti na 4 ili 2 bita. Ovaj način signalizacije može se koristiti za djelomična područja točkastog oblaka ili iz određene razine dubine octree-a. Koristeći QT ili BT samo za jednostavne čvorove lista uzorka, kod popunjenosti može biti kompaktniji bez gubitka točnosti od 8 do 4, odnosno 2 bita, kao što je prikazano na slici 17.



Slika 17. Smanjenje koda popunjenosti pomoću kodiranja s QT/BT [2]

Kutni način rada uveden je kako bi se poboljšao način planarnog kodiranja. Koristi kutnu udaljenost podređenog čvora od laserskih kutova za poboljšanje kompresije binarnog kodiranja popunjenosti kroz predviđanje položaja ravnine planarnog načina i predviđanje z-koordinatnih bitova u IDCM čvorovima.

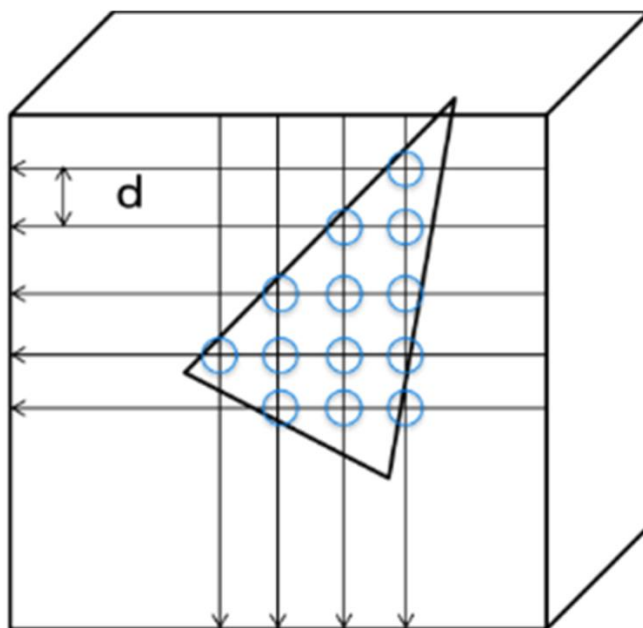


### 5.2.5 „Trisoup“ kodiranje

Kompresija geometrije temeljena na „trisoupu“ TGC (engl. *Trisoup-based Geometry Compression*) dodatni je alat za kompresiju geometrije koji se može koristiti komplementarno razgradnji octree-a. U octree-u zauzeti čvor lista predstavlja konačnu rezoluciju točkastih koordinata. U TGC-u zauzeti čvor lista predstavlja 3D kocku koja može sadržavati jednu ili više točaka unutra. Svaka zauzeta 3D kocka na razini lista je predstavljena kao površina trokutastih traka na kojima se svaki trokut može oblikovati vrhovima na rubovima 3D kocke.

To je izborni alat za kodiranje koji poboljšava subjektivnu kvalitetu pri nižoj brzini prijenosa jer kvantizacija daje grubu prilagodbu brzine. Ako je trisoup omogućen bitstream geometrije postaje kombinacija octree-a, indikatora segmenta i informacija o položaju vrha.

U procesu dekodiranja dekođer izračunava točku presjeka između ravnine „trisoup“ mreže i vokselizirane mreže. Broj izvedenih točaka u dekoderu određen je udaljenošću mreže  $d$  voksel, koja se može kontrolirati kao što se vidi na slici 18.

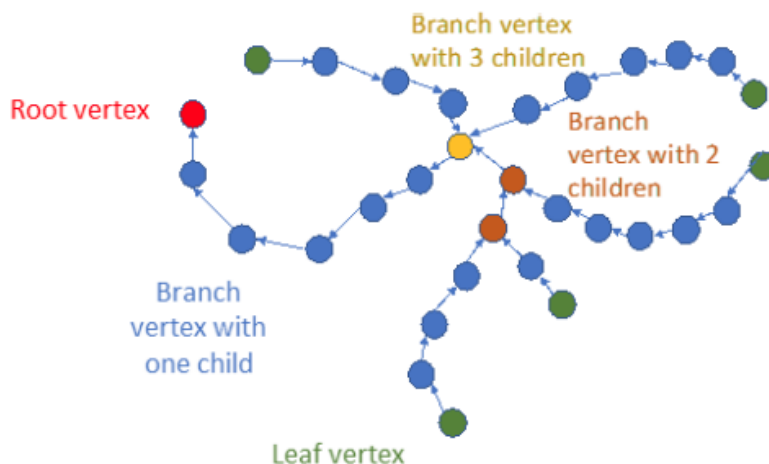


Slika 18. Izvođenje „Trisoup“ točke na dekoderu [11]

Ove informacije o trokutu kodirane su umjesto koordinata točaka koje pripadaju 3D kocki. Na strani dekodera generira se odgovarajući broj točkastih koordinata iz površina trokuta. Ovisno o formiranju koordinata ulazne točke može biti identificirano 3 do 12 vrhova. S ovim vrhovima može biti jedan do deset trokuta u 3D kocki.

### 5.2.6 Prediktivno kodiranje

Za razliku od razgradnje octree-a koja rekurzivno dijeli graničnu kutiju prediktivno kodiranje geometrije je metoda kodiranja od točke do točke. Ova je metoda namijenjena pokrivanju slučajeva upotrebe s malim kašnjenjem, npr. za rukovanje unosom 3D senzora u stvarnom vremenu. U ovom načinu rada koder raspoređuje skup točaka kao stablo predviđanja, kao što je ilustrirano na slici 19. Stablo predviđanja može se granati u do tri podstabla u jednoj točki.



Slika 19. Primjer načina predviđanja geometrije [2]

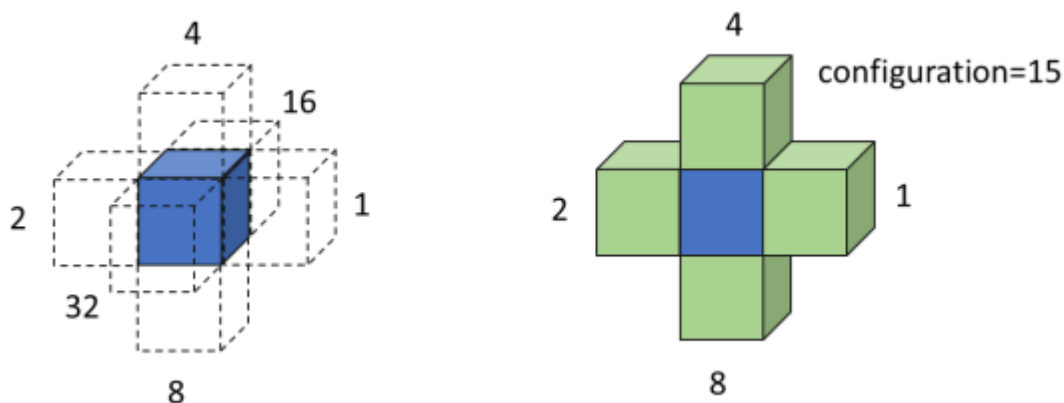
U ovom primjeru stablo počinje od crvene točke, a zatim se grana na žutoj točki u tri. Svaki vrh može se predvidjeti samo od svojih predaka na stablu. Različite boje koriste se za označavanje broja djece za svaki vrh grane. Položaji vrhova kodirani su pohranjivanjem odabranog načina predviđanja i dobivenih ostataka predviđanja. Aritmetičko kodiranje koristi se za daljnje sažimanje generiranih vrijednosti. Izvedbe koderu ovise o konstrukciji stabala i informacijama kao što je redoslijed skeniranja koje se mogu koristiti.

### 5.2.7 Aritmetičko kodiranje

Posljednji korak geometrijskog koderu je aritmetički koder sistematiziran s obzirom na način kodiranja geometrije („octree“, „trisoup“ ili prediktivni). Za „octree“ način rada osigurano je nekoliko poboljšanja u G-PCC-u. Kako bi se maksimizirao dobitak aritmetičkog kodiranja koristi se modeliranje vjerojatnosti temeljeno na kontekstu.

Ovo aritmetičko kodiranje temeljeno na kontekstu zaobilazi se za neke specifične slučajeve kao što je DCM gdje podaci statistički nisu prikladni za kodiranje entropije. Postoji dosta kontekstnih modela za učinkovito kodiranje 8-bitnih uzoraka u G-PCC-u. Kao što je prikazano na slici 20. može

se zauzeti do šest susjeda roditeljskog čvora.

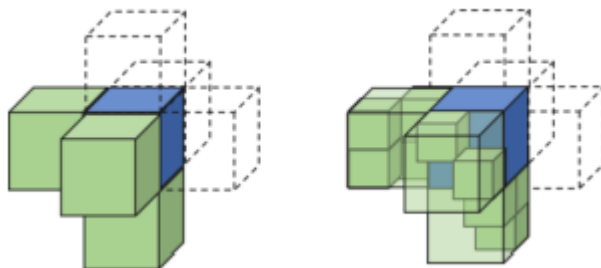


Slika 20. Konfiguracija susjeda NC (lijevo) i primjer za NC=15 (desno) [2]

Stoga su moguće 64 različite konfiguracije susjeda NC (engl. *Neighbor Configurations*). Kada je NC nula, to znači da nadređeni čvor nema zauzeti čvor susjeda. Ako nadređeni čvor ima samo jedno zauzeto dijete podliježe kodiranju s DCM-om. Inače 8-bitni uzorak je kodiran kontekstom  $NC = 0$ .

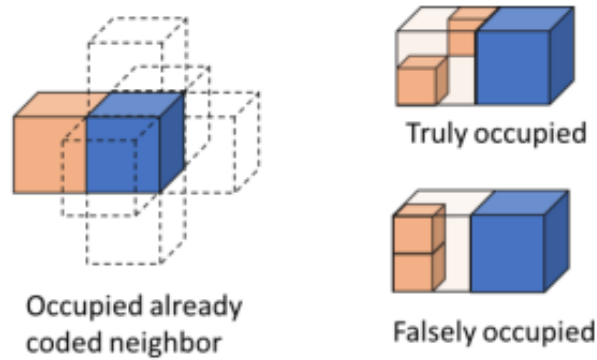
Osim NC-a daljnje usavršavanje konteksta provodi se dodavanjem prethodno kodiranih bitova u 8-bitni uzorak istog čvora što povećava broj konfiguracija na 128 pomnoženo sa 64. G-PCC smanjuje 128 mogućih kombinacija u 10 nepromjenjivih slučajeva pomoću simetrija i rotacija.

Dodatno poboljšanje učinkovitosti kompresije postiže se iskorištavanjem kodiranih podređenih čvorova koji su susjedni trenutnom čvoru kao što je prikazano na slici 21.



Slika 21. Iskorištavanje već kodiranih dječjih čvorova za daljnju učinkovitost kompresije [2]

Pregledom već kodiranih podređenih čvorova moguće je točno utvrditi je li zauzeti susjedski blok doista zauzet i time povezan s trenutnim čvorom što se vidi na slici 22.



Slika 22. Određivanje stvarno ili lažno okupiranog susjeda [2]

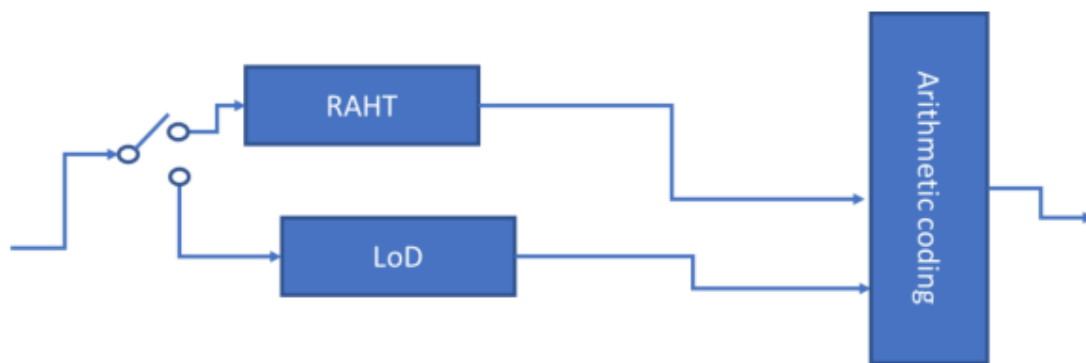
Proces rekonstrukcije geometrije dekodira i rekonstruira koordinate točaka. Rekonstrukcija ne znači potpuni proces dekodiranja jer ne sadrži aritmetičko dekodiranje. Glavni razlog rekonstrukcije geometrije je korištenje za ponovni izračun podataka atributa prema dekodiranim informacijama geometrije. Ako je kompresija informacija o položaju bez gubitaka procesi rekonstrukcije geometrije i prijenosa atributa nisu potrebni i može se provesti paralelno kodiranje (geometrija i atribut).

### 5.2.8 Pretvorba atributa, ponovno bojanje i kodiranje

U točkastom oblaku može postojati mnogo različitih vrsta atributa (npr. boja, normala, refleksija, laserski ID itd.). Međutim, trenutni G-PCC podržava samo podatke o boji i refleksiji. Atributi boje u G-PCC-u predstavljeni su u prostoru boja YCbCr. Pretvorba prostora boja potrebna je ako se izvorni prostor boja za akviziciju razlikuje. Za kodiranje podataka o reflektiranju koristi se isti put kodiranja podataka u boji, pri čemu je jedina razlika u tome što je dimenzija refleksije jedna umjesto tri.

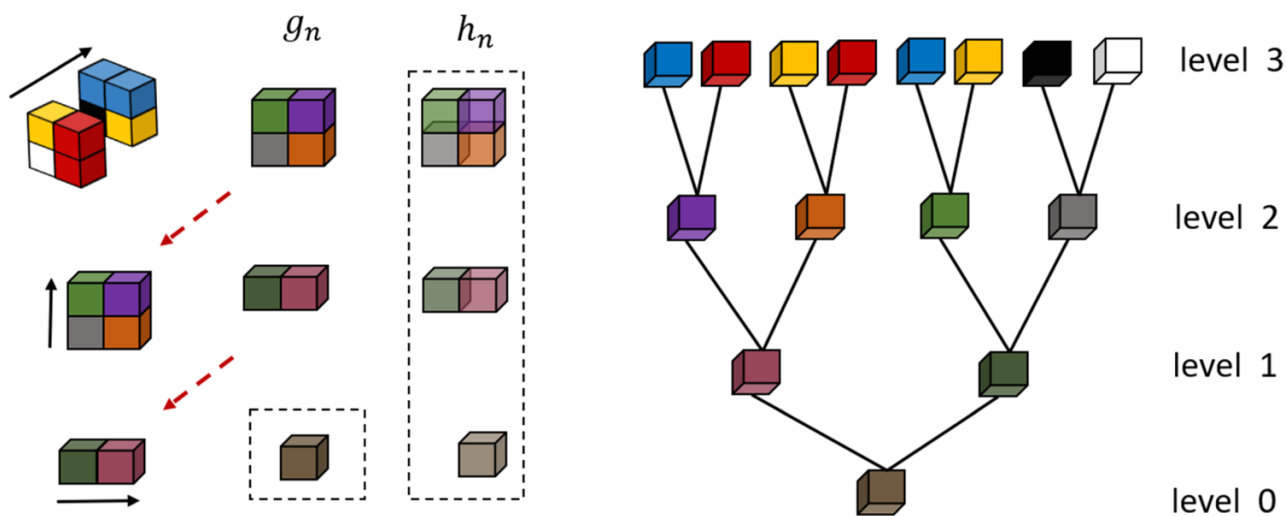
Proces pretvorbe atributa ponovno komponira attribute na temelju geometrije rekonstrukcije kada je kompresija geometrije s gubitcima. U ovom slučaju većina točkastih mjesta nakon kodiranja razlikuje se od onih izvornog oblaka ulazne točke. Stoga je potrebno ponovno napuniti vrijednosti atributa kao što su boja i reflektiranje. Postoji mnogo mogućih načina za ponovno bojanje točaka, jednostavan je korištenje procijenjenog prosjeka pomoću najbližih točaka iz ulazne točke oblaka. Budući da se taj proces obavlja samo na koderu i izvan je opsega standardizacije postoji prostor za daljnje poboljšanje. Ipak dobro ponovno bojanje izravno utječe na izvedbu koderu.

Postoje dvije metode za kodiranje atributa u G-PCC-u: RAHT i LoD, svaka s nekim prednostima i ograničenjima s obzirom na gustoću točkastih oblaka. Cjelokupna shema ovih načina kodiranja za attribute prikazana je na slici 23.



Slika 23. Dijagram kodera atributa G-PCC [2]

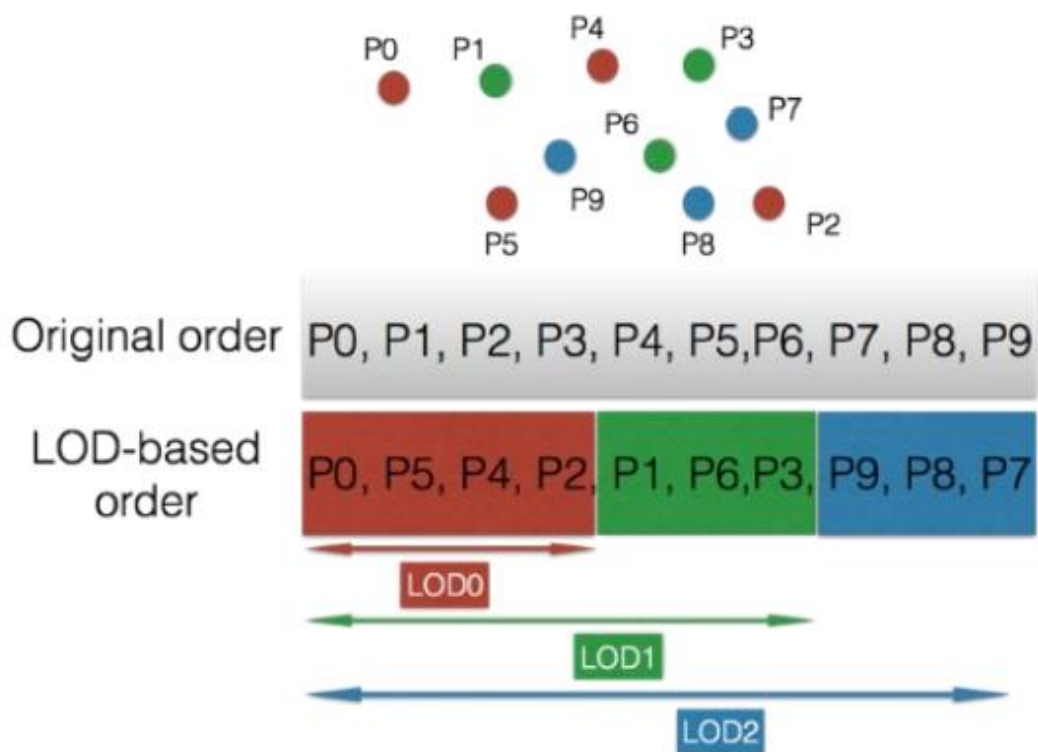
RAHT transformacija - Glavna ideja koja stoji iza RAHT-a je korištenje vrijednosti atributa na nižoj razini octree-a za predviđanje vrijednosti na sljedećoj razini. Temelji se na 3D Haar transformaciji. RAHT se izvodi uzimajući u obzir reprezentaciju oblaka točaka u octree-u. Kreće se od listova octree-a, s najviše pozicije i nastavlja unatrag dok ne dođe do korijena tj. najniže razine. Transformacija se primjenjuje na svaki čvor i izvodi se u tri koraka, po jedan u svakom smjeru x, y i z kao što je prikazano na slici 24.



Slika 24. Proces transformacije bloka  $2 \times 2 \times 2$  [11]

LoD generacija - Proces proizvodnje LOD-a ima za cilj pružiti prostornu skalabilnost G-PCC-u. Točke su odvojene u slojeve. Atributi povezani s točkama predviđaju se iz točaka koje su već kodirane u istom sloju ili iz točaka na donjem sloju. Primjer generiranja LoD-a prikazan je na slici 24. Proces generiranja LoD-a mora generirati drugačiji redoslijed kodiranja atributa od geometrijskog. Npr. grupiranje na temelju udaljenosti vrši se na slici 25.

Proces generiranja LoD-a problem je kodera, što znači da mogu postojati različite još učinkovitije strategije grupiranja točaka u slojeve. Nakon određivanja redoslijeda i grupiranja točaka u slojevima predviđaju se vrijednosti atributa u svakoj točki i njihove razlike kodiraju se pomoću aritmetičkog kodiranja. Proces predviđanja određuje vrijednost predviđanja trenutne točke od najbližih već kodiranih točaka. Pretraživanje najbližih točaka povećava kompjutorsku složenost kodera, kao i učinkovitost kompresije.



Slika 25. Proces generiranja LOD-a [2]

## 6 Kompresija oblaka točkaka temeljena na algoritmima dubokog učenja

Trenutno se učenje značajki naširoko koristi u oblacima točkaka. Metode se mogu svrstati u dvije kategorije:

- 1) sirove metode temeljene na točkama koje izravno troše nestrukturirane i neuređene oblake točkaka za modele dubokog učenja,
- 2) k-dimenzionalne metode temeljene na stablu (Kd – stablo) koje redovito predstavljaju oblak točkaka prije nego što u modele unesu informacije.

Glavni problem s dubokim učenjem u oblaku točkaka je taj što tipična konvolucijska arhitektura zahtijeva vrlo pravilan format ulaznih podataka kao što su slike ili vremenske značajke. Budući da oblak točkaka nije u redovnom formatu uobičajeni pristupi su transformacija podataka u 3D voksel rešetki ili projekcije.

Prvo će se objasniti metode dubokog učenja temeljene na točkama, a zatim metodu dubokog učenja temeljenu na stablu.

### 6.1 Duboko učenje temeljeno na točkama

Trenutno postoji nekoliko modela koji izravno konzumiraju oblak točkaka bez gubitka informacija. Na temelju osnovnog modula ovih modela podijeljeni su u 5 kategorija: temeljeni na „PointNetu“, temeljeni na dubokim konvolucijskim neuronskim mrežama CNN (engl. *Convolutional Neural Network*) – „ConvNets“, ponavljajuće neuronske mreže RNN (engl. *Recurrent Neural Network*) temeljene na autoenkoderu AE i ostale.

„PointNet“ je bio početni pristup za novu vrstu neuronske mreže koja izravno koristi neuređene oblake točkaka, koji također vodi računa o permutacijskoj invarijantnosti točkaka u oblaku točkaka. „Pointnet“ može izvršiti klasifikaciju objekata, segmentaciju dijelova, semantičku analizu scene. Glavna značajka Pointneta je da je mreža robusna s obzirom na ulazne poremećaje i korupciju. Također mreža može naučiti sažeti oblik prema rijetkom skupu ključnih točkaka.

Konvolucijska neuronska mreža CNN ili „ConvNet“ mrežna je arhitektura za duboko učenje koje uči izravno iz podataka. CNN-ovi su posebno korisni za pronalaženje uzoraka u slikama za prepoznavanje objekata, klasa i kategorija. Oni također mogu biti vrlo učinkoviti za klasificiranje audio podataka, vremenskih serija i signala. Konvolucijske neuronske mreže obično se koriste za vizualne slike, pomažući računalu da identificira i uči iz slika. CNN se smatra moćnijim od RNN-a. RNN uključuje manje značajki kompatibilnosti u usporedbi s CNN-om. CNN uzima ulaze fiksnih veličina i generira izlaze fiksnih veličina. RNN može rukovati proizvoljnim ulazno/izlaznim duljinama.

Rekurentna neuronska mreža (RNN) vrsta je umjetne neuronske mreže koja koristi sekvencijalne podatke ili podatke vremenskih serija. RNN pretvara neovisne aktivacije u ovisne aktivacije pružajući iste težine i pristranosti svim slojevima, smanjujući tako složenost povećanja parametara i pamćenja svakog prethodnog izlaza dajući svaki izlaz kao ulaz sljedećem skrivenom sloju. Glavna razlika između CNN-a i RNN-a je sposobnost obrade vremenskih informacija — podataka koji dolaze u sekvencama, kao što je rečenica. Rekurentne neuronske mreže dizajnirane su upravo za tu svrhu, dok konvolucijske neuronske mreže nisu u stanju učinkovito interpretirati vremenske informacije. Ponavljajuća neuronska mreža (RNN) je klasa umjetnih neuronskih mreža gdje veze između čvorova mogu stvoriti ciklus, dopuštajući da izlaz iz nekih čvorova utječe na kasniji unos u iste čvorove. To mu omogućuje da pokazuje vremensko dinamičko ponašanje.

Autokoder je tehnika učenja bez nadzora za neuronske mreže koja uči učinkovito predstavljanje podataka (kodiranje) trenirajući mrežu da zanemari "šum" signala. Autokodere se mogu koristiti za uklanjanje šuma sa slike, kompresiju slike i u nekim slučajevima čak i za generiranje slikovnih podataka. CNN se također može koristiti kao autokoder za smanjenje šuma slike ili bojanje. Kada se CNN koristi za smanjenje šuma slike ili bojanje primjenjuje se u okviru Autoenkodera, tj. CNN se koristi u dijelovima autoenkodera za kodiranje i dekodiranje. Autoenkoder ima funkciju učenja grupiranja podataka.

## 6.2 Duboko učenje temeljeno na stablu

Kd stablo izgrađeno je na oblaku od osam točaka. Čvorovi su numerirani od korijena do lista u Kd stablu. Zbog nepravilnosti oblaka točaka predloženi su postupci temeljeni na Kd stablu za istraživanje lokalnog i globalnog konteksta. Modeli temeljeni na Kd stablu uzimaju oblake točaka kao redovite prezentacije prije nego što unesu informacije u modele dubokog učenja. Ove metode postupno uče vektor prikaza oblaka točaka duž stabla. Eksperimentalni rezultati na zahtjevnim skupovima podataka pokazali su da Kd stablo pruža različite značajke oblaka točaka. Imamo 3 metode: Kd mrežu, 3D kontekstualnu mrežu („3DContextNet“) i „Multiresolution Tree Networks (MRTNet)“.

Kd mreža radi s nestrukturiranim oblakom točaka i dizajnirana je za zadatke prepoznavanja 3D modela. Arhitektura provodi transformaciju množenja i dijeli parametre tih transformacija prema podjeli oblaka točaka na koji se Kd stablo primjenjuje. Za razliku od glavne konvolucijske arhitekture koja obično zahtijeva rasterizaciju jednoličnu dvodimenzionalnu ili trodimenzionalnu mrežu, Kd mreža se ne oslanja na to na bilo koji način čime se izbjegava loše skaliranje.



Baš kao i Kd-mreža, „3DcontextNet“ je predložen za hvatanje lokalnih i globalnih značajki oblaka točaka pomoću strukture Kd-stabla. Za razliku od operacije definiranja Kd-mreže na Kd stablo, „3DcontextNet“ koristi Kd-stablo za predstavljanje oblaka 3D točaka bez mijenjanja prostornih odnosa i može se koristiti za klasifikaciju 3D objekata i semantičku segmentaciju. Postoje dvije glavne komponente u ovoj arhitekturi, tj. učenje značajki na više razina i agregacija značajki za izdvajanje globalnih kontekstualnih informacija.

Za razliku od Kd-mreža i „3DcontextNet-a“, oblaci točaka prvo se sortiraju pomoću „Kd-stabla u MRTNet-u“. Korišteno Kd-stablo može predstavljati oblake točaka u hijerarhijskom poretku koji čuva lokalitet. Konkretno, operacija skupljanja definirana može se koristiti za konstrukciju hijerarhijskog sortiranja, a skaliranje oblaka točaka u više razlučivosti korisno je za očuvanje lokaliteta. Budući da Kd-stablo dijeli oblake točaka ovisnost među njima se više ne održava. Nakon što su oblaci točaka sortirani 1D konvolucija i udruživanje koriste se za izgradnju „MRTNeta“. Eksperimentalni rezultati o klasifikaciji oblika otkrivaju da „MRTNet“ ima prednosti male memorije i velike brzine konvergencije tijekom obuke. „MRTNet“ se također može koristiti kao koder i dekoder za generiranje oblika.

### **6.3 Kompresija geometrije oblaka točaka temeljena na vokselu**

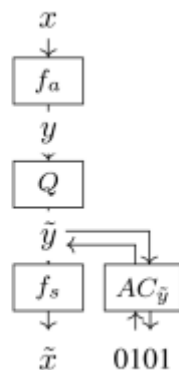
Predstavljamo dvije metode temeljene na učenju za kodiranje geometrije oblaka točaka. Dvije metode koje ciljaju kompresiju s gubitcima i kompresiju bez gubitaka, odnosno zajednička im je činjenica da koriste prikaz geometrije temeljen na vokselu. Ovaj prikaz omogućuje nam da proširimo poznate arhitekture koje se koriste za generiranje 2D slike i kompresiju na 3D. Pokazuje se da kada je gustoća oblaka točke dovoljno visoka pristup temeljen na vokselu postiže najsvremeniju izvedbu u usporedbi s tradicionalnim metodama temeljenim na „octree-u“ kao što je MPEG G-PCC. Nedavno su predložene metode kompresije oblaka dubokih točaka DPCC i pokazalo se da pružaju značajne prednosti kodiranja u usporedbi s tradicionalnim metodama. U nastavku će se obraditi dvije nedavno predložene metode za kodiranje: metoda za kodiranje s gubitcima - (*lossy* metoda) i metoda za kodiranje bez gubitaka - (*lossless* metoda).

Vokselizacija je proces koji kvantificira koordinate oblaka točaka do cjelobrojne preciznosti prije procesa kodiranja i tipka se u većini koda za diskretizaciju geometrije. Također iznosimo implicitnu hipotezu da je oblak točaka dovoljno gust da pokaže lokalne korelacije među susjednim točkama na voksel mreži - u drugim terminima pretpostavljamo da između točaka nema previše "praznog prostora". To nam omogućuje korištenje dubokih neuronskih mreža s 3D konvolucijama temeljenim na vokselu za koje se pokazalo da su posebno učinkovite u kompresiji oblaka točaka.

Kada se oblak točke vokselizira, njegova geometrija može se izraziti kao binarni signal preko voksel mreže. U principu voksel se smatra zauzetim ako sadrži barem jednu točku, a inače nije zauzet. Na temelju tog opažanja, metode temeljene na učenju za kodiranje geometrije obično definiraju dekodiranje kao problem binarne klasifikacije. Umjesto toga, u kompresiji bez gubitaka potrebna je eksplicitna, točna procjena vjerojatnosti popunjenosti voksel: u ovom slučaju dekodiranje se može protumačiti kao proces generiranja voksel.

### 6.3.1 Kompresija slike s gubicima

Uzimajući u obzir oblake točaka kao binarni signal na voksel mreži mogu se koristiti konvolucijske neuronske mreže CNN (engl. *Convolutional Neural Networks*) za kompresiju geometrije oblaka točaka. Naime, arhitekture autoenkodera temeljene na CNN-u imaju za cilj pretvoriti ulaz u niži dimenzionalni nevidljivi prostor i rekonstruirati izlaz identičan ulazu. To se može tumačiti sa stajališta kodiranja transformacije s analizom  $f_a$  i sinteznom transformacijom  $f_s$  kao što je prikazano na slici 26.

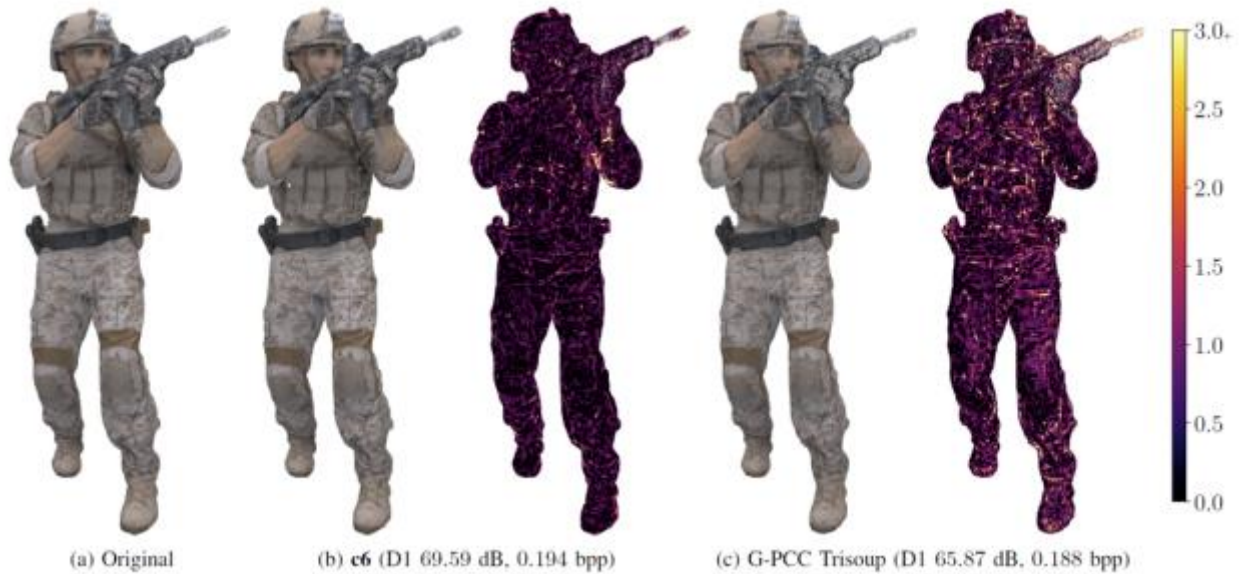


Slika 26. Jednostavna arhitektura autoenkodera za kompresiju [3]

gdje je:  $f_a$  - transformacija analize,  $f_s$  - transformacija sinteze,  $Q$  se odnosi na kvantizaciju, a AC na aritmetičko kodiranje s pripadajućim modelom entropije.

Dio kvantizacije također je bitan u tome što mora biti različit kako bi se trenirala neuronska mreža. Osim toga raspodjela nevidljivog prostora modelirana je kao dio mreže, a rezultirajuća entropija omogućuje optimizaciju izobličenja brzine mreže za određene kompromise s poremećajima brzine. Vjerojatnosti predviđene modelom entropije tada se mogu koristiti aritmetičkim koderom kako bi kodirao i dekodirao nevidljivi prostor.

2019. godine predstavljena je upotreba CNN-ova za kompresiju geometrije oblaka točaka, a predloženi pristup znatno nadmašuje MPEG G-PCC kako je prikazano na slici 27.



Slika 27. Usporedba koda za kompresiju geometrije [3]

Vidi se da pristupi koji se temelje na učenju mogu postići znatno niže distorzije pri ekvivalentnom bitrateu.

Kompresija slike s gubitkom je proces koji uklanja neke podatke iz slikovne datoteke, smanjujući ukupnu veličinu datoteke. Ovaj je proces nepovratan, što znači da će podaci o datoteci biti trajno uklonjeni. Neki od algoritama koji se koriste za kompresiju s gubitkom uključuju: diskretnu valičnu transformaciju, metodu fraktalne kompresije i transformaciju enkripcije.

Ovi algoritmi kompresije s gubitkom mogu znatno smanjiti veličinu datoteke, ali to obično dolazi s kompromisom - kvaliteta slike će također degradirati. Zato je bolje imati sigurnosnu kopiju datoteke prije bilo kakvih promjena.

Format **JPEG** izvrstan je primjer kompresije s gubitkom. Datoteke u **JPEG** formatu su prikladne za slike i fotografije bez prozirnosti. Izvrsni su za web mjesto kojem je potrebno brže vrijeme učitavanja jer možete podesiti omjere kompresije uz održavanje odgovarajuće ravnoteže. Na slici 28. se vidi primjer kompresije s gubitkom koristeći ShortPixel.



Slika 28. Primjer prije i poslije kompresije slike s gubitkom [12]

Vidljivo je da se originalna i komprimirana slika golim okom ne razlikuju. Gubitak informacija može se primijetiti samo ako zumiramo komprimiranu sliku. Veličina datoteke je na ovaj način smanjena za 85%.

Kad se govori o kompresiji s gubitkom geometrije oblaka statičkih točaka u PCC-u je precizna rekonstrukcija geometrijskih informacija od najveće važnosti za omogućavanje visokokvalitetnog iscrtavanja i interaktivnih aplikacija. Tako je kodiranje geometrije bez gubitaka nedavno istraženo u MPEG-u, ali čak i najsuvremenije tehnike trude se komprimirati više od oko 2 bita po zauzetom vokselu. To rezultira velikim troškovima pohrane i prijenosa za bogate oblake točaka. Kompresija s gubitkom temelji se na „octree-u“ koji postižu geometrijsku kompresiju promjenjive brzine promjenom dubine „octree-a“. Nažalost smanjenje dubine eksponencijalno smanjuje broj točaka. Kao rezultat toga kompresija s gubicima temeljena na „octree“ stablu ima tendenciju proizvesti "blokove" rezultate u fazi renderiranja sa srednjim do niskim brzinama prijenosa. Kako bi se djelomično ublažio ovaj problem predlaže se korištenje valičnih transformacija i volumetrijskih funkcija za zbijanje energije signala oblaka točaka. Međutim budući da još uvijek koriste prikaz stabla njihova metoda pokazuje brzu degradaciju geometrije pri nižim brzinama prijenosa. Dok prethodni pristupi koriste ručno izrađene transformacije predlaže se pristup vođen podacima koji se temelji na naučenim konvolucijskim transformacijama koje izravno rade na vokselima. Konkretno predstavlja se metoda za učenje analiza i transformacija sinteze prikladnih za kompresiju geometrije oblaka točaka. Nadalje tumačenjem geometrije oblaka točaka kao binarnog signala definiranog preko mreže vokseli dekodiranje se pretvara u problem klasificiranja je li dati voksel zauzet ili ne. Dodatno za razliku od metoda koje se temelje na „octree-u“ ova metoda ne

pokazuju eksponencijalno smanjenje broja točaka pri snižavanju brzine prijenosa. Također pokazuje se kako ovaj model dobro generalizira korištenjem potpuno različitih skupova podataka za obuku i testiranje.

### 6.3.2 Kompresija slike bez gubitaka

Za razliku od kompresije s gubitkom kompresija slike bez gubitaka neće smanjiti kvalitetu slike. To je zato što kompresija bez gubitaka uklanja samo dodatne nebitne podatke koje je automatski dodao uređaj korišten za snimanje fotografije. Kompromis je u tome što se neće vidjeti značajno smanjenje veličine slikovne datoteke. Kao rezultat toga vjerojatno neće uštedjeti puno prostora za pohranu. Standardni algoritmi kompresije bez gubitaka uključuju: Huffmanovo kodiranje, aritmetičko kodiranje i kodiranje dužine izvođenja. Tehnika kompresije bez gubitaka najbolja je za slike s puno teksta i slike s prozirnrom pozadinom. Vrste formata koji mogu imati koristi od kompresije slike bez gubitaka su **.RAW** , **.BMP** , **.GIF** i **.PNG**. Na slici 29. je primjer ishoda kompresije bez gubitaka.



Slika 29. Primjer prije i poslije kompresije slike bez gubitka [12]

Kao što se vidi na slici 29. nema vidljivog gubitka kvalitete, međutim veličina slikovne datoteke smanjena je za samo 5%. stoga je kompresija bez gubitaka najbolja za slike koje moraju ostati visoke kvalitete.

Algoritmi kompresije bez gubitaka mogu se proširiti korištenjem entropijskih modela temeljenih na dubokom učenju. Ovi entropijski modeli predviđaju vjerojatnosti popunjenosti koje se zatim unose u entropijski koder. Za takva predviđanja mogu se koristiti različite vrste konteksta.

U kontekstu kodiranja octree-a informacije o roditeljskom čvoru mogu se koristiti kao kontekst. Kodiranje octree-a poboljšava se predlažući model entropije koji se temelji na neuronskoj mreži za kodove octree-a. Konkretno ova neuronska mreža uzima lokaciju i informacije od nadređenog čvora kako bi predviđela vjerojatnosti popunjenosti. Ovo se može proširiti na dinamičke oblake točaka i predlaže se upotreba kontinuiranih konvolucija za poboljšanje predviđanja vjerojatnosti korištenjem već dekodiranih okvira. Međutim, takav kontekst pruža samo ograničene informacije.

Uvođenjem sekvencijalne ovisnosti u rešetku vokselu mogu se koristiti vokseli na trenutnom LoD-u kao kontekst. Predlaže se duboki CNN s maskiranim zavojima nazvan VoxelDNN za kompresiju geometrije oblaka točaka bez gubitaka. Neuronska mreža predviđa vjerojatnost popunjenosti svakog vokselu, a vjerojatnosti se zatim šalju aritmetičkom koderu. Particioniranje octree-a optimizirano za brzinu poboljšava performanse na rjeđim regijama.

Izvedba kompresije dodatno je poboljšana tehnikama povećanja podataka i proširenja konteksta. Nedostatak takvih pristupa je da sekvencijalna ovisnost značajno povećava složenost. Značajno se smanjuje vremenska složenost pristupom s više razmjera. Smanjuje složenost paralelnom procjenom vjerojatnosti uz pristup s više razmjera koji smanjuje sekvencijalnu ovisnost pri procjeni vjerojatnosti popunjenosti.

Također, može se u potpunosti ukloniti sekvencijalne ovisnosti predviđanjem popunjenosti vokselu koristeći samo roditeljski LoD. Predlaže se neuronska mreža koja uzima kontekst vokselne mreže od matičnog LoD-a za predviđanje vjerojatnosti popunjenosti. Te se vjerojatnosti zatim koriste za entropijsko kodiranje. Osim toga predviđaju pomak za svaku točku kako bi dodatno precizirale koordinate. Iako su takvi pristupi razumno složeni oni ne uzimaju u obzir već dekodirane voksele na trenutnom LoD-u što ograničava performanse kompresije.

Oba dvije metode i metoda za kompresiju s gubicima i metoda za kompresiju bez gubitaka smanjuju veličinu slikovne datoteke, ali razlikuju se u metodama i rezultatima. Glavna razlika je da se kompresijom slike kod metode s gubicima trajno uklanjaju neki slikovni podatci, dok se kod kompresije bez gubitaka ne smanjuje kvaliteta slike i jedino se uklanjaju neki nebitni podatci. Kod oba dvije metode prikaz je temeljen na vokselu geometrije. Ovaj nam prikaz omogućuje proširenje dobro poznate arhitekture koje se koriste za generiranje 2D slike i kompresiju u 3D. Vidljivo je da kada je gustoća oblaka točaka dovoljno visoka pristup temeljen na vokselima postiže najsuvremeniju izvedbu u usporedbi s konvencionalnim metodama koje se temelje na „octree-u“ kao što je MPEG G-PCC. Ova prednost je dana sposobnošću predstavljanja temeljne geometrijske strukture (lokalne površine, objekti, itd.) koja nije obuhvaćena jednostavnim pristupima koji se temelje na octree-u. S

druge strane, kada je oblak točaka rjeđi ili oskudan (kao za LiDAR podatke), tehnike temeljene na vokselima ne funkcioniraju, a druge vrste pristupa su prikladnije, kao što su konvolucije temeljene na točkama i grafikonima.

## 7 Procjena kvalitete oblaka točaka

Mjere za procjenu kvalitete oblaka točaka se mogu podijeliti na subjektivne i objektivne mjere procjene kvalitete.

### 7.1 Subjektivna procjena kvalitete oblaka točaka

Kvaliteta iskustva QoE (engl. *Quality of Experience*) definirana je, prema COST Action Qualinetu, kao stupanj užitka ili dosađivanja korisnika aplikacije ili usluge. Na QoE utječe nekoliko čimbenika koji se općenito mogu podijeliti u tri glavne kategorije: čimbenici povezani s ljudima, čimbenici povezani sa sustavom i kontekstom. Za mjerenje QoE različitih multimedijских signala može se provesti subjektivna procjena kvalitete, koja predstavlja kvalitetu svake testirane stavke sadržaja jednim brojem što u nekim slučajevima možda nije dovoljno da u potpunosti opiše QoE. Na primjer, u tipičnoj subjektivnoj kampanji za procjenu kvalitete slike ili videozapisa promatrači gledaju niz originalnih i degradiranih slika ili video sekvenci i ocjenjuju njihovu kvalitetu numerički.

Subjektivna kvaliteta određene slike ili videozapisa mjeri se prosjekom svih ocjena korisnika za tu sliku ili videozapis, tj. srednjom ocjenom mišljenja MOS (engl. *Mean Opinion Score*) koja se smatra ocjenom kvalitete koju bi prosječni gledatelj dodijelio toj određenoj slici ili videozapisu. MOS rezultati prikupljaju se u skladu s dobro definiranim metodama i postupcima predloženim posljednjih desetljeća i usmjereni su na jamčenje uporabe istih eksperimentalnih postavki i uvjeta tijekom raznih procjena.

Najčešće metode procjene kvalitete subjektivne slike i videozapisa predložene su u preporuci ITU-R BT.500-14. Ova preporuka i druge povezane s njom definiraju metode pojedinačnog ili dvostrukog podražaja za provođenje subjektivne procjene kvalitete. Ovisno o tome kako je sadržaj prikazan korisniku. Neke od definiranih metoda “Single-Stimulus” (SS), “Double Stimulus Continuous Quality Scale” (DSCQS), “Stimulus-Comparison” (SC), and “Single Stimulus Continuous Quality Evaluation” (SSCQE). Najčešća metoda procjene subjektivne kvalitete je postupak DSCQS u kojemu promatrač ocjenjuje par slika ili video sekvenci koje dolaze iz istog izvora, od kojih se izvorni ili referentni signal promatra izravno bez daljnje obrade, a drugi prolazi kroz sustav ispitivanja koji je stvaran ili simulira pravi sustav, što rezultira obrađenim ili testnim signalom. Promatrač ocjenjuje izvorne i obrađene signale obično na ljestvici razlika što rezultira skupinom rezultata koji predstavljaju perceptivne razlike između referentnih i testnih videozapisa ili slika. Imaju i alternativne metode za procjenu slika ili video sekvenci u kojem korisnici procjenjuju slike ili video sekvence koje sadrže oštećenja koja se mijenjaju tijekom vremena. Trenutačno



subjektivna procjena oblaka točaka još nije standardizirana, ali slični postupci mogu se prilagoditi da budu kao uobičajene metode procjene kvalitete slike ili videozapisa.

## 7.2 Objektivna procjena kvalitete oblaka točaka

Objektivne mjere kvalitete vizualnih podataka kao što su slike i videozapisi, a time i oblaci točaka, općenito se upotrebljavaju kada je subjektivnu procjenu možda teško provesti. Alternativno, također se koriste u različitim scenarijima kao što su praćenje ili optimizacija sustava slikovne i videokomunikacije.

Objektivne mjere ili procjene kvalitete izračunavaju se prema određenom algoritmu i mogu se podijeliti u tri skupine prema vrsti ulaznih podataka koje zahtijeva algoritam:

- FR (engl. *Full Reference*) mjere: koriste se potpuni izvorni i degradirani vizualni podaci.
- RR (engl. *Reduced Reference*) mjere: koriste se samo neke značajke iz izvornih informacija o vizualnim podacima i degradiranih informacija o vizualnim podacima.
- NR (engl. *No Reference*) mjere: koriste se samo degradirane informacije o vizualnim podacima.

Objektivne mjere kvalitete za točkaste oblake trenutno se razvijaju koristeći postojeće mjere kvalitete razvijene za primjenu na sliku i videozapise, nakon nekih izmjena kako bi se nosili s tim s različitim oblicima prikaza. Općenito te se mjere mogu podijeliti u dvije glavne kategorije:

- mjere koje se temelje na projekcijama oblaka točaka izračunatim na 2D prostorima za koje se predviđaju točke
- mjere temeljene na geometriji i/ili atributima izračunane na izvornom 3D prostoru u kojem su prikazane informacije o oblaku točaka [13].

- 

### 7.2.1 Mjere temeljene na projekcijama oblaka točaka

Bilo koji oblak točaka može se projicirati na jednu ili više projekcijskih ravnina, a nakon toga se može procijeniti svaka projekcijska ravnina koristeći bilo koju od postojećih mjera kvalitete slike kao što je vršni omjer signal/šum ili indeks strukturne sličnosti. Ako se koristi nekoliko projekcijskih ravnina koje dovode do nekoliko projiciranih slika konačni rezultat može se izračunati kao srednja vrijednost rezultata povezanih sa svakom od projiciranih slika. Tu se koristi softver za renderiranje koji stvara vokseliziranu verziju oblaka točaka u stvarnom vremenu i projicira oblak točaka s 3D na

2D ravninu. Nakon toga se projicirane slike uspoređuju pomoću postojećih mjera za kvalitetu slike postižući visoku korelaciju s rezultatima subjektivne procjene [13].

### 7.2.2 Mjere temeljene na geometriji i/ili atributima

Općenito su predložene dvije različite metode za mjerenje geometrijskog izobličenja: udaljenost od točke do točke i udaljenost od točke do ravnine. Prvo vektor pogreške može se definirati kao vektor razlike između proizvoljne točke u prvoj točki oblaka pa do odgovarajuće točke identificirane algoritmom najbližeg susjeda u oblaku druge točke. Mjere od točke do točke djeluju izračunavanjem udaljenosti (duljina vektora pogreške) između svake točke u jednom od oblaka točaka (izvornih ili degradiranih) i najbliže točke u oblaku druge točke (degradiran ili originalan).

Nakon toga izračunata prosječna kvadratna udaljenost između parova točaka koristi se kao mjera izobličenja geometrije. Udaljenost se može definirati drugačije, pri čemu se u većini slučajeva koriste dva pristupa: Hausdorffova udaljenost (Jednadžba 1) i L2 norma. Kada se koristi L2 norma MSE (engl. *Mean Squared Error*)(Jednadžba 2) ili RMSE (engl. *Root Mean Squared Error*)(Jednadžba 3.) se može izračunati između svih parova najbližih točaka. Budući da se ova mjera može izračunati na dva različita načina, ovisno o redoslijedu oblaka točaka oblak prve točke može biti izvorni točkasti oblak, a drugi degradirani točkasti oblak i obrnuto), konačna mjera obično se definira kao mjera lošija/viša ocjena koja se naziva simetrična ocjena [13].

$$Haus_{p2p} = \max (\|E_{i,j}\|_2^2) \quad (1)$$

$$MSE_{p2p} = \frac{1}{n} \sum_{i=1}^n \|E_{i,j}\|_2^2 \quad (2)$$

$$RMSE_{p2p} = \sqrt{MSE_{p2p}} = \sqrt{\frac{1}{n} \sum_{i=1}^n \|E_{i,j}\|_2^2} \quad (3)$$

U jednadžbama (1) – (3)  $E_{i,j}$  se definira kao vektor razlike između proizvoljne točke u prvom oblaku točaka do odgovarajuće najbliže točke u drugom oblaku točaka. Međutim, mjerenja od točke do točke ne uzimaju u obzir implicitne površine pa se zato proučava nova mjera nazvana „cloud to mesh“ ili „point to surface“ koja bi uspješno predstavljala površinu. Udaljenosti od oblaka do mreže približne su udaljenosti od površine do površine tako da se prvo uzorkuje jedna mreža oblaka točaka, a zatim se izračuna udaljenost točke do površine između svake uzorkovane točke temeljene na mreži

i druge površine oblaka točaka. Predlaže se i alternativna mjera zasnovana na geometriji za oblake točaka koja se zove „point to plane“ (p2pl). Ova mjera trebala bi postići veću korelaciju sa subjektivnom procjenom u usporedbi s p2pl mjerom.

Mjera p2pl može se izračunati na sljedeći način:

- Za svaku točku  $a_j$  u prvom oblaku točaka identificira se odgovarajuća točka  $b_j$  u drugom oblaku točaka npr. algoritmom najbližeg susjeda.
- Vektor pogreške  $E_{i,j}$  definiran je kao vektor razlike između proizvoljne točke u prvom oblaku točaka  $a_j$  i odgovarajuće najbliže točke u drugom oblaku točaka  $b_j$ .
- Jedinični vektor normale  $N_j$  izračunava se za svaku točku  $a$  u prvom oblaku točaka.
- Vektor pogreške projicira se na jedinični vektor normale izračunavanjem umnoška između vektora pogreške i vektora normale dobivajući projicirani vektor pogreške.
- Mjera p2pl izračunava se kao srednja vrijednost kvadrata svih projiciranih vektora pogreške.

Slično kao i kod *point to point* mjera, MSE, RMSE i Hausdorffova udaljenost mogu se koristiti u p2pl mjerama. Definicija MSE p2pl prikazana je u jednadžbi 4, RMSE p2pl u jednadžbi 5 i Hausdorff p2pl u jednadžbi 6.

$$MSE_{p2pl} = \frac{1}{n} \sum_{i=1}^n (\langle E_{i,j}, N_j \rangle)^2 \quad (4)$$

$$RMSE_{p2pl} = \sqrt{MSE_{p2pl}} = \sqrt{\frac{1}{n} \sum_{i=1}^n (\langle E_{i,j}, N_j \rangle)^2} \quad (5)$$

$$Haus_{p2pl} = \max (\langle E_{i,j}, E_{i,j}, N_j \rangle)^2 \quad (6)$$

Predstavlja se nova mjera koja koristi omjer vršnog signala i šuma PSNR (engl. *Peak Signal to Noise Ratio*) koja normalizira pogreške povezane s vršnom vrijednošću svakog oblaka točaka. Vršna vrijednost može se definirati drugačije, to je najveća dijagonalna udaljenost graničnog okvira oblaka točaka. U MPEG standardu to se naziva D1/D2 PSNR mjera koja je definirana jednadžbom 7.

$$PSNR_{geometry} = 10 \log_{10} \left( \frac{3p^2}{symmetricMSE_{geometry}} \right), p = 2^{pr} - 1 \quad (7)$$

gdje je  $p$  vršna vrijednost signala koja normalizira pogrešku, a  $pr$  je preciznost koordinata oblaka

točaka. U nazivniku se mogu koristiti i RMSE ili udaljenost temeljena na Hausdorffu.

MPEG standard predlaže MSE i PSNR mjere temeljene na atributima. Budući da je YUV prostor bolje povezan s ljudskom percepcijom provodi se konverzija iz RGB prostora u YUV prostor. Nakon toga se MSE vrijednost posebno izračunava za svaku komponentu boje. Obično se maksimalna vrijednost između dobivenih MSE vrijednosti koristi za izračunavanje simetričnog rezultata. Komponenta PSNR izračunava se prema jednadžbi 8. Ako komponente boje atributa za sve oblake točaka imaju dubinu 8 bita, onda je vršna vrijednost  $p$  koja se koristi u jednadžbi 8 255. Konačni globalni MSE ili PSNR izračunava se kao srednja vrijednost pojedinačnih MSE ili PSNR koristeći težinu od 6/8 za svjetlinu (Y) i 1/8 za obje boje (U i V).

$$PSNR_{attribute} = 10 \log_{10} \left( \frac{p^2}{symmetricMSE_{attribute}} \right) \quad (8)$$

Predlaže se i nova objektivna mjera kvalitete za oblake točaka nazvana PCQM (engl. *Point Centred Quarter Method*). Mjera koristi informacije iz značajki oblaka točaka koje se temelje na geometriji i na atributima i izračunava konačni rezultat kao kombinaciju nekoliko predloženih značajki. PCQM je ispitan na skupu podataka MPEG s tri kodera ( „Octree pruning“, G-PCC i V-PCC) svaki s tri razine kvalitete i dobivena je najveća korelacija sa subjektivnim rezultatima među svim testiranim objektivnim mjerama.

## 8 Usporedba rezultata različitih metoda kompresije

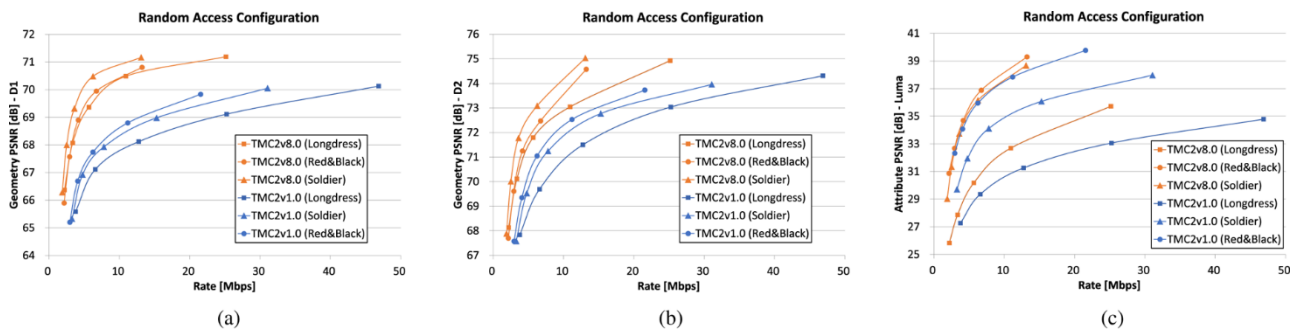
U ovom dijelu rada pokazat će se primjeri usporedbe izvedbe V-PCC-a za slučajeve primjene kompresije u AR/VR-u, te G-PCC-a za primjene kompresije prikaza arhive kulturne baštine i u autonomnoj vožnji.

Geometrija i atributi kodirani su korištenjem najnovijeg referentnog softvera TMC2v8.0 i TMC13v7.0 prema uobičajenim uvjetima ispitivanja koje je postavila MPEG grupa. Nadalje, izobličenja zbog artefakata kompresije mjere se alatom „pc\_error“, gdje se uz geometrijsko izobličenje od točke do točke ( $D 1$ ) i od točke do ravnine ( $D 2$ ) također navodi izobličenje atributa u smislu bitrate-a. Dio PCC skupa podataka koji se koristi najčešće u simulacijama prikazan je na slici 30. [11]



Slika 30. Set za testiranje oblaka točaka: a)Duga haljina b)Crveno i crno c)Vojnik d)Glava e) Ford [11]

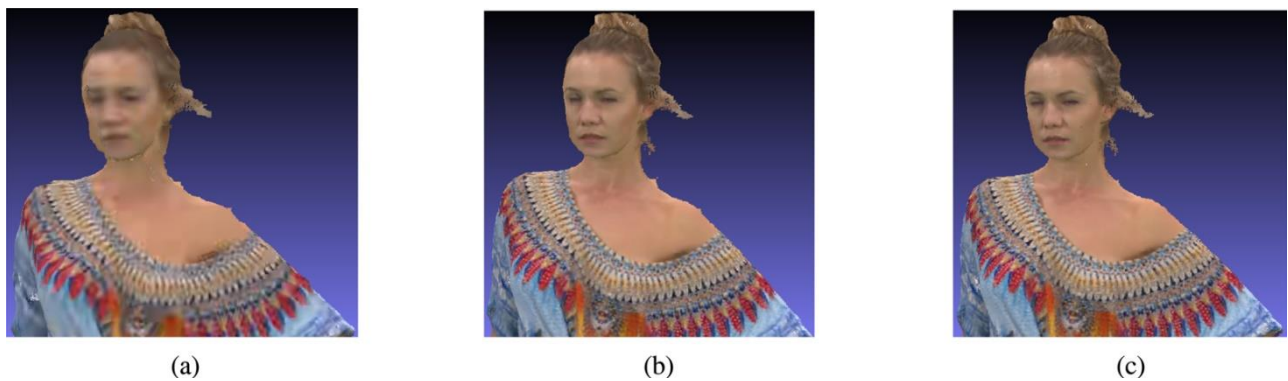
Za V-PCC se prikazuje izvedba kompresije TMC2 za kodiranje sekvenci dinamičkog oblaka točaka, kao što su sekvence duge haljine, crvene i crne i vojnika. Svaki okvir oblaka točaka ima otprilike 1 milijun točaka s vokseliziranim položajima i RGB atributima boja. Od početka procesa standardizacije performanse referentnog softvera značajno su poboljšane, kao što je prikazano na grafikonima prikazanim na slici 31. Na primjer za niz dugih haljina dobitak postignut s TMC2v8.0 u usporedbi s TMC2v1.0 je oko 60% uštede u BD-stopi za metriku od točke do točke ( $D 1$ ) i oko 40% uštede u BD-brzini za atribut luma.



Slika 31. TMC2v1.0 u odnosu na TMC2v8.0, (a) izobličenje geometrije od točke do točke ( $D_1$ ), (b) izobličenje geometrije od točke do ravnine ( $D_2$ ) (c), izobličenje atributa Luma [11]

Oba mjerila točka-točka i točka-ravnina pokazuju učinkovitost usvojenih tehnika. Nadalje poboljšanje performansi RD za attribute također se može provjeriti. Budući da se TMC2 oslanja na postojeći video koder za izvedbu kompresije, poboljšanje uglavnom dolazi od optimizacije kodera kao što je raspodjela zakrpa među okvirima na način vremenske dosljednosti ili smjernice za procjenu kretanja za video koder iz generatora zakrpa.

Slika 32. prikazuje vizualizaciju kodiranih sekvenci korištenjem MPEG PCC Reference Rendering Software-a. S najnovijom TMC2 verzijom 8 bitstream-a od 10–15 Mbps pruža dobru vizualnu kvalitetu preko 70 dB dok je početnom TM v1 bilo potrebno 30–40 Mbps za istu kvalitetu.

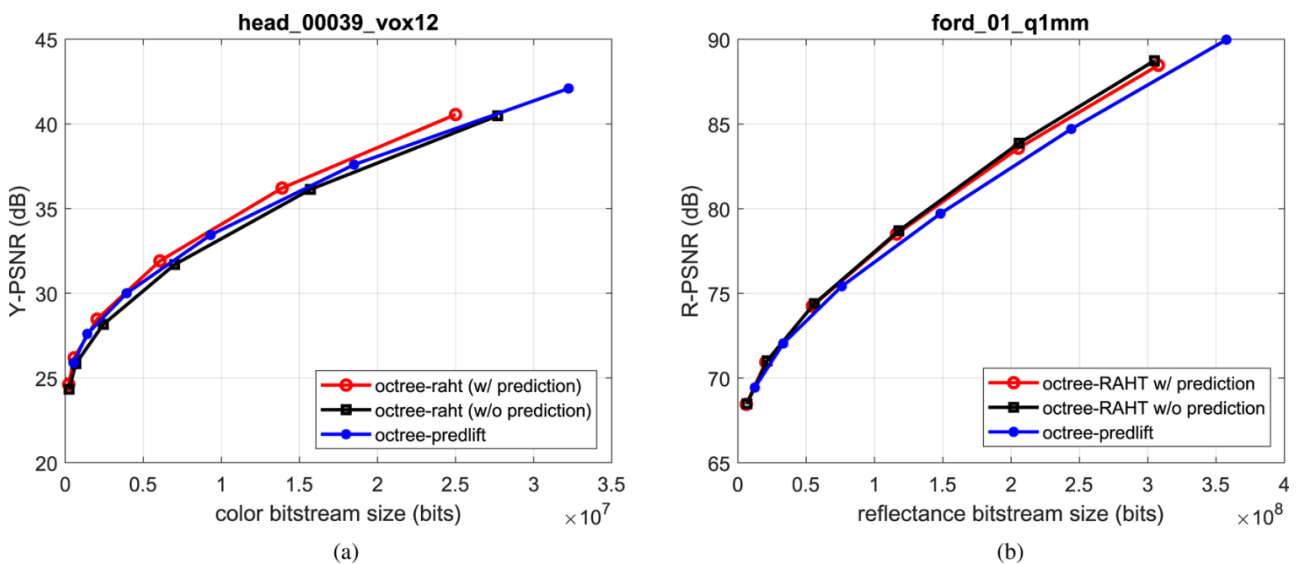


Slika 32. Subjektivna kvaliteta pomoću PCC softvera za renderiranje, (a) Duga haljina pri 2,2 Mbps, (b) Duga haljina pri 10,9 Mbps, (c) Duga haljina pri 25,2 Mbps [11]

Za G-PCC postoje različiti scenariji kodiranja koji uključuju predstavljanje geometrije: „octree“ ili „trisoup“, transformaciju atributa (RAHT, Predicting ili Lifting Transforms), kao i metode kompresije (s gubicima, bez gubitaka i gotovo bez gubitaka). Ovisno o slučaju korištenja odabire se podskup ovih alata.

Obe metode koristeći geometriju „octree-a“ bez gubitaka i sheme kompresije atributa s gubitkom. Prikazani su rezultati za dva oblaka točaka koji predstavljaju specifične slučajeve uporabe. Prvi oblak točaka (Glava\_00039\_vox12, oko 14 milijuna točaka) primjer je primjene kulturne baštine. U ovom slučaju upotrebe, oblaci točaka su gusti i poželjno je očuvanje atributa boje. Drugi oblak točaka (ford\_q1 mm, oko 80 tisuća točaka po okviru) zapravo je niz oblaka točaka, i predstavlja aplikaciju za autonomnu vožnju.

U ovom slučaju upotrebe, oblaci točaka su vrlo rijetki i općenito je poželjno očuvanje informacija o refleksiji. Slika 33. a) prikazuje dijagrame PSNR osvjetljenja (Y-PSNR) za Glava\_00039\_vox12 kodiran s RAHT-om (sa i bez predviđanja domene transformacije) i transformacijom podizanja. Slika 33. b) prikazuje dijagrame refleksije PSNR (R-PSNR) za ford\_01\_q1 mm kodirane s istim koderima atributa. Kao što je prije navedeno, u oba slučaja, geometrija je kodirana bez gubitaka.



Slika 33. Primjeri izvedbe G-PCC kodiranja [11]

PSNR dijagrami pokazuju da za head\_00039\_vox12 RAHT s predviđanjem domene transformacije nadmašuje RAHT bez predviđanja i Lifting Transform u smislu stope osvjetljenja-izobličenja. Za ford\_01\_q1 mm, RAHT nadmašuje podizanje; međutim, isključivanje predviđanja domene transformacije daje nešto bolje rezultate.

Što se tiče geometrije bez gubitaka i kompresije atributa bez gubitaka, koristi se predviđanje transformacije. U ovom slučaju, uzimajući u obzir i geometriju i attribute, omjeri kompresije za head\_00039\_vox12 i ford\_01\_q1 mm su oko 3 prema 1 odnosno 2 prema 1. Uzimajući u obzir sve oblake točaka u CTC-u, prosječni omjer kompresije je oko 3 prema 1.

## 9. Primjena oblaka točaka

Postoje mnoge razne aplikacije koje koriste oblake točaka kao preferirani format za hvatanje podataka. Spomenuti će se neke najčešće slučajeve primjene, a to su VR/AR, telekomunikacije, autonomno vozilo, svjetska baština itd.

VR/AR – dinamički nizovi oblaka točaka mogu pružiti korisniku mogućnost da vidi pokretni sadržaj s bilo kojeg gledišta, značajka koja se još zove 6 stupnjeva slobode (6DoF). Kao primjer imamo aplikacije za vizualizaciju oblaka točaka pomoću mobilnih uređaja. Korištenjem dostupnih video dekodera i GPU resursa prisutnih u mobilnom telefonu V-PCC kodirani oblaci točaka se dekodirani i rekonstruirani u stvarnom vremenu. Kada se kombinira s AR okvirom slijed oblaka točaka može se prekriti u stvarnom svijetu putem mobilnog uređaja što je prikazano na slici 34.a).

Telekomunikacije – zbog visoke učinkovitosti kompresije V-PCC omogućuje prijenos videa u oblaku točaka preko mreže s ograničenim opsegom. Zbog toga se može koristiti za aplikacije prisutnosti na daljinu. Primjer korisnika koji nosi uređaj za prikaz na glavi, bit će u mogućnosti komunicirati s virtualnim svijetom na daljinu slanjem/primanjem oblaka točaka kodiranih s V-PCC što se vidi na slici 34.b).



a)



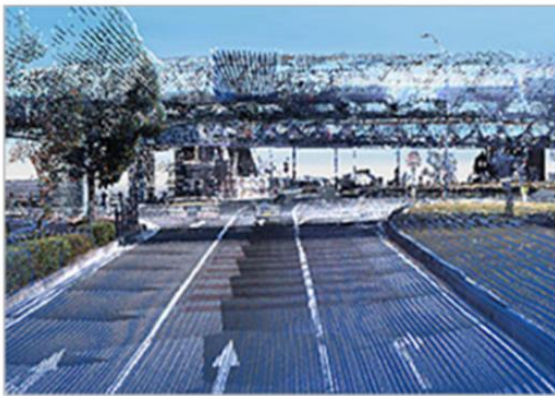
b)

Slika 34. Primjer slučajeve korištenja oblaka točaka za a )VR/AR i b) telekomunikacije [11]



Autonomno vozilo – vozila za autonomnu vožnju koriste oblake točaka za prikupljanje informacija o svojoj okolini kako bi izbjegli neželjene sudare kao što je prikazano na slici 35. a). Danas se za dobivanje 3D informacija na vozila montiraju razni vizualni senzori. LIDAR senzor je jedan primjer od takvih senzora, snima svoje okruženje kao vremenski promjenjivi slijed rijetkih oblaka točaka. G-PCC može komprimirati ovu rijetku sekvencu i stoga pomoći u poboljšanju protoka podataka unutar vozila laganim i učinkovitim algoritmom.

Za arhiv kulturne baštine objekt se skenira 3D senzorom u statički oblak točaka visoke razlučivosti što se vidi na slici 35. b). Mnogi istraživački projekti generiraju visokokvalitetne oblake točaka povijesne arhitekture ili objekata kako bi ih sačuvali i stvorili digitalne kopije za virtualni svijet. Laserski skener raspona ili struktura iz pokreta su dvije tehnike koje se koriste u procesu generiranja sadržaja. Uz to G-PCC se može koristiti za komprimiranje generiranih oblaka točaka bez gubitaka, smanjujući zahtjeve za pohranu uz očuvanje točnih mjera.



a)



b)

Slika 35. Primjer slučajeva korištenja oblaka točaka za a) autonomno vozilo [11]

## 10. Zaključak

Primjena oblaka točaka sve je češća, pruža brojne mogućnosti, ali brojne primjene i velike količine podataka predstavljaju izazov za prijenos i pohranu. Iz tog razloga je bitna kompresija oblaka točaka. Međutim sa sve većom potražnjom za oblacima točaka s većom preciznošću kompresija postaje veoma bitno pitanje što rezultira uvođenjem odgovarajućih aplikacija. Glavni problem pri rješavanju ovog pitanja odnosi se na potpuno nestrukturiranu prirodu 3D oblaka točaka.

U ovome radu su objašnjeni glavni koncepti vezani uz MPEG PCC te je predstavljen pregled alata za kodiranje koji se koriste u trenutnim modelima za MPEG V-PCC i MPEG G-PCC standarde. V-PCC je temeljen na pristupu projiciranja svakog oblaka točaka na skup ravnina nakon čega slijedi njihovo 2D kodiranje, a G-PCC na geometriji. Očekuje se da dva PCC standarda pružaju konkurentna rješenja za novo tržište, zadovoljavajući pritom razne zahtjeve primjene ili slučajevne upotrebe. U budućnosti MPEG također razmatra proširenje PCC standarda na nove slučajevne upotrebe kao što je dinamička kompresija mreže za V-PCC i uključenje novih alata kao što je *inter* kodiranje za G-PCC.

Objašnjene su neke od metoda za duboko učenje koje su se pokazale boljima u odnosu na spomenute tradicionalne metode kompresije. Opisane su dvije arhitekture temeljene na dubokom učenju za kompresiju geometrije oblaka točaka: metodu s gubicima i metodu bez gubitaka. U oba slučaja, upotreba vokselskih konvolucija daje značajne dobitke u odnosu na referentno G-PCC rješenje za guste oblake točaka. Ova prednost dana je sposobnošću predstavljanja temeljne geometrijske strukture koja nije obuhvaćena jednostavnim pristupima koji se temelje na „octree-u“. S druge strane, kada je oblak točaka rjeđi ili oskudan kao za LiDAR podatke, tehnike temeljene na vokselim ne funkcioniraju, a druge vrste pristupa su prikladnije, kao što su konvolucije temeljene na točkama i grafikonima.

Pristupi temeljeni na dubokom učenju pokazali su odlične karakteristike na kompresiji geometrije i postali usporedivi s najnovijim dostignućima pristupa kompresiji atributa. Uz pomoć tehnologija otkrivanja i klasifikacije temeljenih na dubokom učenju mogu se odrediti različiti algoritmi i konfiguracije kompresije za svaku vrstu točaka, radi čega tehnologije dubokog učenja zaslužuju daljnje istraživanje i nadogradnje za oblake točaka.

## 11. LITERATURA

- [1] Chao Cao, Marius Preda, and Titus Zaharia. 2019. 3D Point Cloud Compression: A Survey. In Web3D '19: The 24th International Conference on 3D Web Technology (Web3D '19), July 26.–28. 2019. Los Angeles, CA, USA. ACM, New York, NY, USA  
<https://doi.org/10.1145/3329714.3338130>
- [2] Chao Cao, 3D point cloud compression, doktorski rad, 'Institut Polytechnique de Paris, 2021.
- [3] Maurice Quach, Jiahao Pang, Tian Dong, Giuseppe Valenzise, Frédéric Dufaux. Survey on Deep Learning-based Point Cloud Compression. *Frontiers in Signal Processing*, Frontiers, 2022. 2
- [4] Emil Dumic, Anamaria Bjelopera and Andreas Nüchter, Dynamic Point Cloud Compression Based on Projections, Surface Reconstruction and Video Compression, *Sensors*, 2022.
- [5] Giuseppe Valenzise, Maurice Quach, Dat-Thuan Nguyen, Frédéric Dufaux. Voxel-based Deep Point Cloud Geometry Compression. CORESA (COmpression et REprésentation des Signaux Audiovisuels), Nov 2021. Sophia-Antipolis, France.
- [6] Fotogrametrija interijera i 3D skeniranje, dostupno na:  
<https://www.mojkvart.hr/Zapresic/Geodezija-geotehnika/GEOVID/Fotogrametrija-interijera-i-3D-skeniranje-S80627> Pristupljeno: 02.01.2023.
- [7] Slika TLS skenera, Dostupno na:  
<http://icydebrisfans.blogs.bucknell.edu/files/2015/05/P1290247.jpg> pristupljeno: 02.01.2023.
- [8] Slika ALS skenera, Dostupno na:  
<http://artemavdeenko.weebly.com/uploads/5/0/6/3/50634105/8965779.jpg?254> pristupljeno: 02.01.2023.
- [9] Slika MLS skenera, Dostupno na:  
<https://headingassociates.com.au/wpcontent/uploads/elementor/thumbs/mobile-laser-scaled-ozb3cvzdfh5g6j5v7z0929b0269qp38c00zep5mglk.jpg> pristupljeno: 02.01.2023.
- [10] "Lossy i Lossless" kompresija slike, Dostupno na: [https://www.hostinger.com/tutorials/lossy-vs-lossless#Lossy\\_Image\\_Compression](https://www.hostinger.com/tutorials/lossy-vs-lossless#Lossy_Image_Compression) pristupljeno: 02.01.2023.
- [11] Pregled standardizacije kompresije oblaka točaka: na temelju videa i na temelju geometrije, Dostupno na: <https://www.cambridge.org/core/journals/apsipa-transactions-on-signal-and-information-processing/article/an-overview-of-ongoing-point-cloud-compression-standardization-activities-videobased-vpcc-and-geometrybased-gpcc/56FCAF660DD44348BCB1BCA9B5EC56CF> pristupljeno: 02.01.2023.
- [12] Leonardo Belemečić, Kompresija oblaka točaka, Seminar, Odjel za elektrotehniku Sveučilište u Dubrovniku, lipanj 2022.
- [13] E. Camuffo, D. Mari, S. Milani, Recent Advancements in Learning Algorithms for Point Clouds: An Updated Overview. *Sensors* 2022. 22, 1357.

[14] Weiping Liu , Jia Sun , Wanyi Li , Ting Hu and Peng Wang, Deep Learning on Point Clouds and Its Application: A Survey, 26. rujna 2019.

[15] Emil Dunic, and Luis A. da Silva Cruz, Point Cloud Coding Solutions, Subjective Assessment and Objective Measures: A Case Study, Instituto de Telecomunicações, 3030-290 Coimbra, Portugal, 26. studenog 2020.

## 12. POPIS SLIKA

Slika 1. Primjer TLS skenera [7] .....	3
Slika 2. Primjer zračnog ALS skenera [8] .....	4
Slika 3. Primjer mobilnog MLS skenera [9] .....	5
Slika 4. Fotogrametrija interijera i 3D skeniranje[6] .....	6
Slika 5. Oblak točaka dobiven isključivo iz fotografija iz zraka[10] .....	6
Slika 6. Cjelokupni 3D prikaz[10] .....	7
Slika 7. Primjeri nekih skupova podataka: (a) ShapeNet, (b) MPEG, (c) 8iVFB, (d) S3DIS, (e) SemanticKITTI, (f) SynthCity .....	11
Slika 8. (a) 3D zakrpa, (b) 3D karta zakrpe zauzetosti, (c) 3D slika geometrije zakrpe, (d) 3D slika teksture zakrpe[10] .....	13
Slika 9. Dijagram V-PCC kodera [11] .....	14
Slika 10. Generiranje zakrpa [11] .....	15
Slika 11. Primjer pakiranja zakrpa [11] .....	17
Slika 12. Ispuna teksture mip mapa slike s rijetkom linearnom optimizacijom [11] .....	20
Slika 13. Dijagram G-PCC referentnog kodera [2] .....	22
Slika 14. Dijagram kodera geometrije G-PCC-a [2] .....	24
Slika 15. Prva dva koraka u konstrukciji „octree“ procesa [11] .....	24
Slika 16. Primjer planarnog načina rada gdje su zauzeti čvorovi u istoj ravnini [2] .....	26
Slika 17. Smanjenje koda popunjenosti pomoću kodiranja s QT/BT [2] .....	26
Slika 18. Izvođenje „Trisoup“ točke na dekoderu [11] .....	27
Slika 19. Primjer načina predviđanja geometrije. [2] .....	28
Slika 20. Konfiguracija susjeda NC (lijevo) i primjer za NC=15 (desno) [2] .....	29
Slika 21. Iskorištavanje već kodiranih dječjih čvorova za daljnju učinkovitost kompresije. [2] .....	29
Slika 22. Određivanje stvarno ili lažno okupiranog susjeda. [2] .....	30
Slika 23. Dijagram kodera atributa G-PCC. [2] .....	31
Slika 24. Proces transformacije bloka $2 \times 2 \times 2$ . [11] .....	31
Slika 25. Proces generiranja LOD-a. [2] .....	32
Slika 26. Jednostavna arhitektura autoenkodera za kompresiju. [3] .....	36
Slika 27. Usporedba kodera za kompresiju geometrije. [3] .....	37
Slika 28. Primjer prije i poslije kompresije slike s gubitkom. [12] .....	38
Slika 29. Primjer prije i poslije kompresije slike bez gubitka. [12] .....	39
Slika 30. Set za testiranje oblaka točaka:a)Duga haljina b)Crveno i crno c)Vojnik d)Glava e) Ford	

[11].....	47
Slika 31. TMC2v1.0 u odnosu na TMC2v8.0, (a) izobličenje geometrije od točke do točke ( $D$ 1), (b) izobličenje geometrije od točke do ravnine ( $D$ 2) (c), izobličenje atributa Luma. [11] .....	48
Slika 32. Subjektivna kvaliteta pomoću PCC softvera za renderiranje, (a) Duga haljina pri 2,2 Mbps, (b) Duga haljina pri 10,9 Mbps, (c) Duga haljina pri 25,2 Mbps. [11] .....	48
Slika 33. Primjeri izvedbe G-PCC kodiranja [11] .....	49
Slika 34. Primjer slučajeve korištenja oblaka točaka za a )VR/AR i b) telekomunikacije .....	50
Slika 35. Primjer slučajeve korištenja oblaka točaka za a) autonomno vozilo.....	51

## IZJAVA

Izjavljujem pod punom moralnom odgovornošću da sam diplomski rad izradio samostalno, isključivo znanjem stečenim na studijima Sveučilišta u Dubrovniku, služeći se navedenim izvorima podataka i uz stručno vodstvo mentora doc.dr.sc. Anamarije Bjelopere, kojoj se još jednom srdačno zahvaljujem.

Ime i prezime studenta

*L. Beleručić*