ТЕХНИЧЕСКИ УНИВЕРСИТЕТ – СОФИЯ

Факултет по Телекомуникации
Катедра „Радиокомуникации и видеотехнологии“

Маг. инж. Ивайло Божидаров Божилов

КОДИРАНЕ И ВИЗУАЛИЗАЦИЯ НА 3D ОБЕКТИ, ЧРЕЗ АРХИТЕКТУРИ ЗА ДЪЛБОКО ОБУЧЕНИЕ

А В Т О Р Е Ф Е Р А Т

на дисертация за придобиване на образователна и научна степен
"ДОКТОР"

Област: 5. Технически науки

Професионално направление: 5.3. Комуникационна и компютърна техника
Научна специалност: Телевизионна и видеотехника

Научен ръководител: доц. д-р Агата Манолова

СОФИЯ, 2026 г.

Дисертационният труд е обсъден и насочен за защита от Катедрения съвет на катедра „Радиокомуникации и видеотехнологии“ към Факултет по Телекомуникации на ТУ-София на редовно заседание, проведено на 11.05.2026 г..

Публичната защита на дисертационния труд ще се състои на 20.07.2026 г. от 11:00 часа в Конферентната зала на БИЦ на Технически университет – София на открито заседание на научното жури, определено със заповед № ОЖ-5.3-45 / 22.05.2026 г. на Ректора на ТУ-София в състав:

1. Доц. д-р Никол Христова – председател
2. Доц. д-р Аделина Алексиева-Петрова – научен секретар
3. Проф. д-р Александър Бекярски
4. Проф. д-р Габриела Атанасова
5. Доц. д-р Страхил Соколов

Рецензенти:

1. Доц. д-р Никол Христова
2. Проф. д-р Александър Бекярски

Материалите по защитата са на разположение на интересуващите се в канцеларията на Факултет по Телекомуникации на ТУ-София, блок №1, кабинет № 1254.

Дисертантът е задочен докторант към катедра „Радиокомуникации и видеотехнологии“ на Факултет по Телекомуникации. Изследванията по дисертационната разработка са направени от автора, като някои от тях са подкрепени от научноизследователски проекти.

Автор: маг. инж. Ивайло Божилов
Заглавие: Кодиране и визуализация на 3D обекти, чрез архитектури за дълбоко обучение
Тираж: 20 броя
Отпечатано в ИПК на Технически университет – София

I. ОБЩА ХАРАКТЕРИСТИКА НА ДИСЕРТАЦИОННИЯ ТРУД

Актуалност на проблема

Развитието на системи за тримерно съдържание, разширена реалност, холографска комуникация и интерактивни мултимедийни приложения води до значително нарастване на изискванията към ефективното кодиране, предаване и визуализация на 3D данни. Облаците от точки и другите форми на тримерно представяне се характеризират с изключително голям обем от данни, което поставя сериозни ограничения върху комуникационните и изчислителните ресурси в съвременните системи. Класическите методи за компресия, въпреки високата си технологична зрялост, срещат затруднения при адаптация към сложната структура на 3D данните и динамичните условия на предаване. В този контекст обучаемите и семантично ориентираните подходи, базирани на дълбоко обучение, се очертават като перспективно направление за повишаване на ефективността на компресията, устойчивостта към шум и адаптивността на системите за предаване на 3D съдържание. Това определя актуалността на настоящия дисертационен труд, насочен към изследване и разработване на архитектури и методи за кодиране и визуализация на 3D обекти чрез дълбоко обучение.

Цел на дисертационния труд, основни задачи и методи за изследване

Основната цел на настоящата дисертация е да се изследват и разработят методи за интеграция на обучаеми и семантично ориентирани подходи в системи за заснемане, предаване и визуализация на 3D съдържание, с цел повишаване на ефективността, адаптивността и функционалността на процеса на кодиране.

За постигане на тази цел се формулират следните основни задачи:

1. Анализ на възможностите за интеграция на обучаеми и семантично ориентирани подходи за кодиране в системи за заснемане, предаване и визуализация на 3D съдържание.

2. Изследване и разработка на автоенкодерни архитектури за кодиране на 3D източници.

3. Изследване и разработка на автоенкодерни архитектури за съвместно кодиране източник–канал на 3D съдържание, с цел постигане на устойчивост към канални смущения и ефективност при крайни дължини на блоковете.

4. Реализация, експериментално изследване и сравнителен анализ на предложените методи и архитектури.

Научна новост

Научната новост на дисертационния труд се състои в разработването и изследването на обучаеми и семантично ориентирани методи за кодиране и предаване на тримерно съдържание, базирани на автоенкодерни архитектури и дълбоко съвместно кодиране източник–канал. Предложена е теоретична постановка за ентропийно кодиране с несъгласувани вероятностни модели, позволяваща използването на по-сложни модели в кодера чрез въвеждане на странична информация. Разработени са нови архитектурни решения за компресия и предаване на разредени облаци от точки, включително метод за фазово-инвариантно декодиране при предаване на динамични облаци от точки, който намалява зависимостта от синхронизацията между предавателя и приемника. Получените резултати разширяват приложението на обучаемите методи в областта на компресията и

комуникацията на 3D съдържание и показват възможности за повишаване на ефективността и устойчивостта на системите при реалистични канални условия.

Практическа приложимост

Практическата приложимост на дисертационния труд се изразява в разработването на програмни реализации и методи за ефективно кодиране, предаване и визуализация на 3D съдържание, приложими в системи за разширена реалност, холографска комуникация, телеприсъствие и интерактивни мултимедийни среди. Предложените автоенкодерни архитектури и методи за дълбоко съвместно кодиране източник–канал позволяват намаляване на необходимата скорост за предаване при запазване на добро качество на реконструкцията и устойчивост към шум в комуникационния канал. Реализирани са програмни системи за компресия на RGB-D изображения и облаци от точки, както и експериментални реализации на AEPCC и DPCT архитектурите, които могат да бъдат използвани като основа за бъдещи научни изследвания и практически системи за обработка и предаване на тримерно съдържание. Резултатите от дисертацията могат да намерят приложение в съвременни комуникационни системи, базирани на 5G/6G мрежи, облачни инфраструктури и XR платформи.

Публикации

Основните постижения и резултати от дисертационния труд са публикувани в 6 научни публикации, от които 3 са публикувани в международни научни списания, а останалите 3 са представени на международни научни конференции. Една от публикациите е самостоятелна.

Международните научни списания са: IEEE Access 2024 и 2025 и MDPI Sensors 2023.

Международните научни конференции са: IEEE International Scientific Conference on Information, Communication and Energy Systems and Technologies (ICEST) 2025, IEEE International Symposium on Wireless Personal Multimedia Communications (WPMC) 2025 и Joint International Conference on Digital Arts, Media and Technology with ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunication Engineering (ECTI DAMT & NCON) 2026.

Структура и обем на дисертационния труд

Дисертационният труд е в обем от **151** страници, като включва увод, **5** глави за решаване на формулираните основни задачи, списък на основните приноси, списък на публикациите по дисертацията, списък на програмните реализации и използвана литература. Цитирани са общо **97** литературни източници, като **94** са на латиница, а останалите са интернет адреси. Работата включва общо **49** фигури и **8** таблици. Номерата на фигурите и таблиците в автореферата съответстват на тези в дисертационния труд.

Дисертационният труд е структуриран в пет глави. В Глава 1 е направен анализ на състоянието на проблема и са разгледани класически, обучаеми и семантични методи за кодиране на 3D съдържание. В Глава 2 е въведен операционен модел на системи за заснемане, предаване и визуализация на 3D съдържание и е анализирана интеграцията на обучаеми методи в различните слоеве на системата. В Глава 3 са разгледани автоенкодерни архитектури за кодиране на геометричната структура на разредени облаци от точки. В Глава 4 е изследвано използването на тези архитектури за компресия чрез квантуване и вторично кодиране на латентните представяния, а в Глава 5 е разгледано дълбоко съвместно кодиране източник–канал за предаване на облаци от точки през канали с шум.

II. СЪДЪРЖАНИЕ НА ДИСЕРТАЦИОННИЯ ТРУД

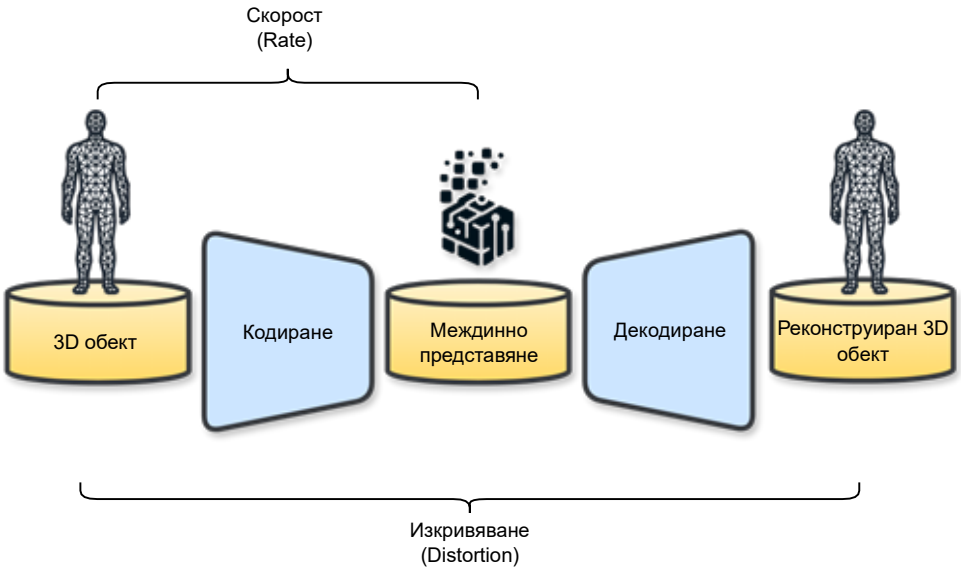
ГЛАВА 1. АНАЛИЗ НА СЪСТОЯНИЕТО НА ПРОБЛЕМА ПО ЛИТЕРАТУРНИ ДАННИ

Първа глава разглежда основните принципи и съвременните подходи за кодиране на 3D източници, като поставя акцент върху развитието на методите за компресия на тримерно съдържание. Представени са основните категории методи — класически, обучаеми и семантични — които се различават по начина на формиране и интерпретация на междинното представяне на данните. Анализирани са техните теоретични основи, предимства и ограничения, както и връзката им с компромиса скорост–изкривяване и статистическото моделиране на 3D данни. Целта е да се изгради концептуална рамка за сравнение на различните подходи и да се очертаят тенденциите в развитието на съвременните системи за компресия и предаване на 3D съдържание.

1.3 Основи на кодирането на 3D източници

1.3.1 Категоризация на методи за кодиране на 3D източници

В резултат на изложените теоретични постановки може да се формулира обобщена дефиниция на кодирането на източника като операция, която осигурява компактно междинно представяне на данните, позволяващо възстановяване на оригиналния входен сигнал с контролирано изкривяване, при значително по-ниски изисквания към пропускателната способност на канала или необходимия обем за съхранение. На Фиг. 1.4 е илюстрирана обобщена схема на процеса на компресия, като са обозначени и основните метрики за оценка на качеството, както и етапите, в които те се измерват.



Фигура 1.4: Обобщена схема на процеса на компресия с обозначени метрики за оценка

Анализът в областта на компресията на 3D съдържание в [A1] групира методите за кодиране в три категории. Тези категории се разграничават според формата на междинното представяне и средствата, чрез които то се получава. Идентифицирани са следните три категории методи: класически, обучаеми и семантични, които са разгледани в тази глава съответно в раздели 1.4, 1.5 и 1.7.

1. **Класически методи за компресия.** При този клас методи междинното представяне обикновено е вектор от неинтерпретируеми за човека признаци или битова последователност с висока ентропия. Компресията се постига чрез класически техники описани в 1.1. Тези подходи са дълбоко вкоренени в класическата теория на информацията и цифровата обработка на сигнали и формират основата на утвърдените стандарти за компресия на 3D съдържание.

2. **Обучаеми методи за компресия**. При обучаемите методи междинното представяне също е вектор от неинтерпретируеми за човека признаци, но средствата за получаването му са базирани на методи от областта на машинното обучение и дълбокото обучение. Най-често се използват невронни мрежи, автоенкодерни архитектури и други обучаеми модели, които се оптимизират спрямо критерии от тип скорост–изкривяване. За разлика от класическите подходи, тук преобразуването и вероятностното моделиране се научават директно от данните, което позволява по-добро адаптиране към сложната структура на 3D източниците.

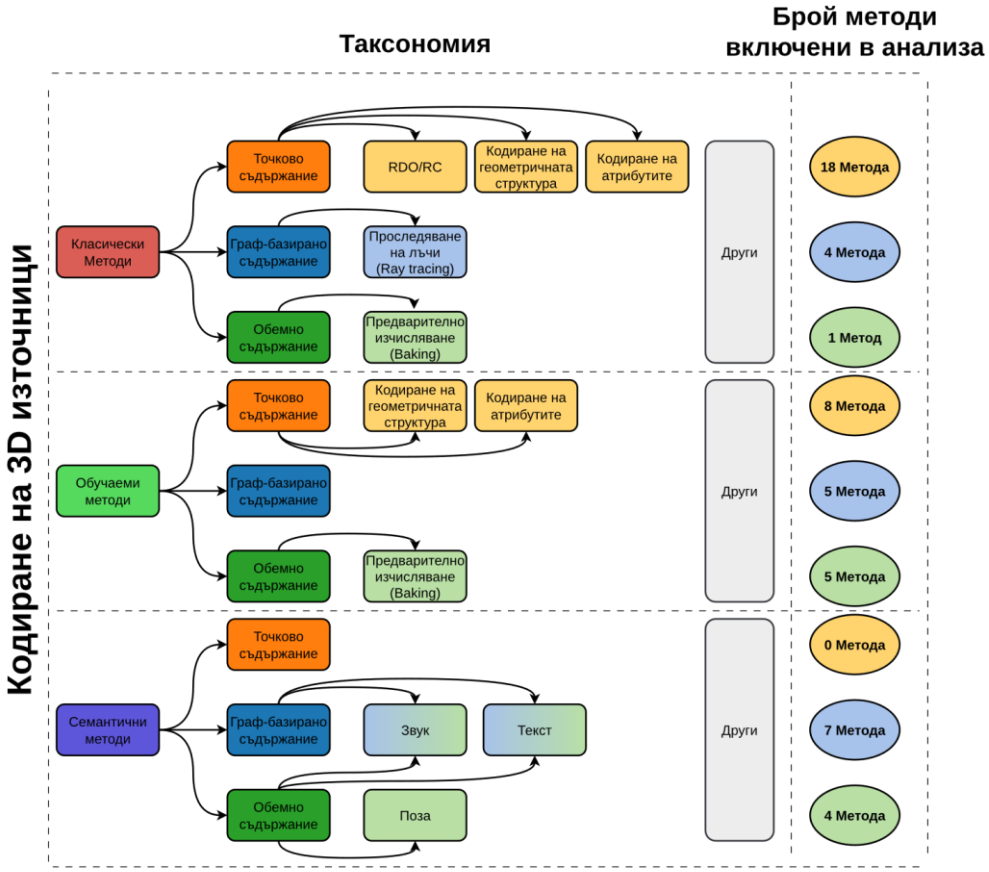
3. **Семантични методи за компресия**. Семантичната компресия се отличава с това, че междинното представяне е съставено от интерпретируеми за човека признаци, които носят смислова информация за сцената или обектите. Този подход цели запазване на контекста и значението на съдържанието. Семантичните методи използват техники от дълбокото обучение, машинното обучение и извличане на семантични признаци, като позволяват директна интерпретация или последваща обработка на компресираното представяне без пълна реконструкция на оригиналния сигнал.

Таксономията, изложена в систематичния обзор [A1], отразява разпределението и основните направления на съвременните методи за компресия на 3D съдържание. На Фиг. 1.5 е представена графична илюстрация на тази таксономия, която ще служи като концептуална рамка при анализа и сравнението на различните класове методи в следващите раздели на дисертацията.

1.8 Изводи

Извършеният анализ на състоянието на проблема показва, че кодирането на визуално, и в частност на 3D съдържание, се основава на добре установена теоретична рамка, произтичаща от класическата теория на информацията и трансформационното кодиране, при която ключова роля играят компромисът скорост–изкривяване и ефективното моделиране на статистическите зависимости в данните. Класическите методи, представени чрез стандарти като видео-базирана компресия на облаци от точки (Video-based Point Cloud Compression) (V-PCC) и геометрично-базирана компресия на облаци от точки (Geometry-based Point Cloud Compression) (G-PCC), демонстрират висока зрялост и ефективност, особено при точково съдържание, което се потвърждава както от широкото им приложение, така и от доминиращото им присъствие в литературата [A1].

От друга страна, обучаемите методи разширяват тази парадигма чрез използване на невронни мрежи за автоматично извличане на компактни представяния и вероятностно моделиране, което позволява по-добро адаптиране към сложната структура на 3D данните



Фигура 1.5: Таксономията на методите за кодиране на 3D източници, представена в [A1].

и води до съществени подобрения в ефективността на кодирането [A1]. В контекста на съвместното кодиране източник-канал (Joint Source-Channel Coding) (JSCC), тези методи позволяват директна оптимизация на представянето спрямо характеристиките на канала и крайната задача, което води до по-устойчиво поведение при шум и крайни дължини на блоковете в сравнение с класическото разделно кодиране [55].

Въпреки това, обучаемите подходи се характеризират с по-висока степен на неопределеност, свързана както с обучението, така и с генерализацията към множество данни, което ги поставя в позиция на компромис между ефективност и надеждност спрямо класическите методи. Паралелно с това се наблюдава развитие към използване на интерпретируеми междинни представяния, при които компресираното съдържание може да бъде анализирано, модифицирано или използвано без необходимост от пълна реконструкция [A1, A2]. Такива представяния намират практически приложения, например при търсене, редактиране, взаимодействие със сцени или интеграция с други системи, което ги прави интересни от инженерна гледна точка.

Както е обобщено на Фиг. 1.17, адаптирана по [A1], съвременните подходи могат да се разглеждат като еволюция от класически към обучаеми и интерпретируеми, при която се откриват нови възможности за повишаване на ефективността на кодирането, съпроводени с по-ниска технологична зрялост и по-голяма степен на неопределеност. Това очертава необходимостта от разработване на нови методи, които съчетават предимствата на класическите и обучаемите подходи с използването на интерпретируеми междинни представяния, с цел постигане на по-висока ефективност, гъвкавост и адаптивност при кодиране и визуализация на 3D съдържание.

Фигура 1.17 Обобщение на направленията в компресията на 3D съдържание, адаптирано по [A1].

1.9 Дефиниране на целта и основните задачи на настоящата дисертация

Основната цел на настоящата дисертация е да се изследват и разработят методи за интеграция на обучаеми и семантично ориентирани подходи в системи за заснемане, предаване и визуализация на 3D съдържание, с цел повишаване на ефективността, адаптивността и функционалността на процеса на кодиране.

За постигане на тази цел се формулират следните основни задачи:

1. Анализ на възможностите за интеграция на обучаеми и семантично ориентирани подходи за кодиране в системи за заснемане, предаване и визуализация на 3D съдържание;

2. Изследване и разработка на автоенкодерни архитектури за кодиране на 3D източници;

3. Изследване и разработка на автоенкодерни архитектури за JSCC на 3D съдържание, с цел постигане на устойчивост към канални смущения и ефективност при крайни дължини на блоковете;

4. Реализация, експериментално изследване и сравнителен анализ на предложените методи и архитектури.

ГЛАВА 2. ИНТЕГРАЦИЯ НА ОБУЧАЕМИ МЕТОДИ ЗА КОДИРАНЕ В СИСТЕМИ ЗА ТРИМЕРНО СЪДЪРЖАНИЕ

Програмната реализация на подходите за кодиране в слоя за заснемане може да бъде намерена в [B1]: https://github.com/Teleinfrastructure-Research-Lab/rgbd-fusion

В Глава 2 е въведен унифициран 4-слоен операционен модел за системи за заснемане, предаване и визуализация на 3D съдържание, чрез който се анализира разпределението на изчислителните и комуникационните ресурси в различните части на системата. На тази основа са разгледани практически подходи за кодиране в слоя за заснемане, включително компресия на RGB-D данни чрез оцветяване, мултиплексиране и семантично-ориентирана обработка, които позволяват намаляване на комуникационния товар при ниска изчислителна сложност. Освен това е формулирана теоретична постановка за кодиране в слоя за визуализация, при която кодерът използва по-точен и по-сложен вероятностен модел от декодера, а несъответствието между двата модела се компенсира чрез странична информация, което поставя основата за изследване на връзката между точност на модела, сложност и необходима скорост на предаване.

2.1 Операционен модел на системи за заснемане, предаване и визуализация на тримерно съдържание

За да бъде анализирана интеграцията на обучаеми методи за кодиране в реални системи, е необходимо да се въведе общ операционен модел на система за заснемане, предаване и визуализация на тримерно съдържание. В [A3] е предложен 4-слоен модел, който проследява потока на данните в системи за холографска комуникация (Holographic Type Communication) (HTC). Той може да бъде разглеждан в по-общ контекст, тъй като описва основните операции върху 3D данните: заснемане, обработка, пренасяне и визуализация. В настоящата дисертация този модел се използва като универсален операционен модел, като HTC се разглежда като негов частен случай.

С съществено предимство на модела от [A3] е, че той е “data-centric”, т.е. фокусиран е около преобразуването и предаването на данните, което позволява ясно разграничаване между изчислително и комуникационно ``натоварване". Компромисът между тези два ресурса е определящ за архитектурата на системата и интеграцията на методите за кодиране.

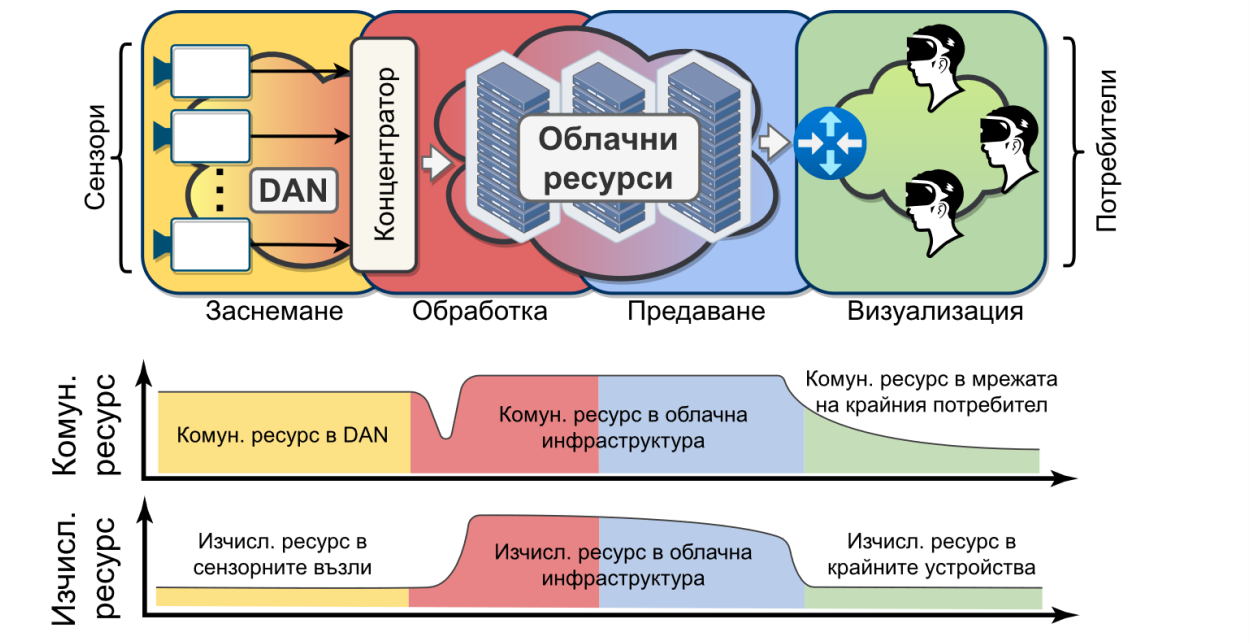
Моделът от [A3] е адаптиран към общия случай за системи за заснемане, предаване и визуалзиация на 3D съдържание и е илюстриран на Фиг. 2.1. Той включва следните слоеве:

1. Заснемане - Преобразува физическата сцена в цифрови данни чрез сензори и базова предварителна обработка.

2. Обработка - Преобразува суровите данни в структурирано 3D представяне чрез операции като реконструкция и сегментация.

3. Предаване - Осигурява ефективно и надеждно пренасяне на данните между възлите на системата.

4. Визуализация - Възстановява и визуализира сцената чрез крайни устройства като очила за виртуална или добавена реалност (Head-Mounted Display) (HMD).



Фигура 2.1: Операционен модел на системи за заснемане, предаване и визуализация на тримерно съдържание.

2.1.2 Разпределение на ресурсите по слоеве

Един от основните изводи в [A3] е, че в системите за 3D съдържание изчислителните и комуникационните ресурси са съвместно определящи за цялостната ефективност. Това следва от факта, че данните не се предават директно, а преминават през последователност от операции по преобразуване (напр. реконструкция, сегментация, кодиране), които изискват значителен изчислителен ресурс. Следователно ефективността не може да се характеризира единствено чрез обема на пренесените данни, а чрез разпределението на ресурса между изчисление и комуникация. Формално, системното проектиране включва избор на алгоритми, които минимизират комуникационния товар при дадени ограничения върху изчислителната сложност, или обратно [4, 67]. В рамките на настоящата дисертация фокусът е върху комуникационния аспект на този компромис. По-конкретно, разглеждат се методи за кодиране, които намаляват необходимата скорост, при даден изчислителен ресурс, без да се анализират в детайли останалите етапи на обработка (напр. реконструкция, сегментация, калибрация др.), характерни за една система за 3D съдържание или HTC система.

Разпределението на изчислителните и комуникационните ресурси не е еднородно по слоевете на системата. Всеки слой се характеризира със специфичен профил на наличните ресурси, което определя допустимата сложност на алгоритмите и ефективните стратегии за кодиране (Фиг. 2.1 и Табл. 2.2).

В обобщение, разпределението на ресурсите по слоеве е нееднородно: изчислителният капацитет нараства при преминаване от слоя за заснемане към слоя за обработка, докато комуникационните ограничения се проявяват най-силно при преноса на данни между различните страни, участващи в комуникацията, през публична мрежа (локална среда на заснемане → облачни изчислителни клъстери → крайни устройства). Това налага съвместна оптимизация на изчислителните и комуникационните ресурси при проектиране и интеграция на методи за кодиране.

Таблица 2.2: Обобщение на ресурсните ограничения по слоеве

Слой	Изчислителни ресурси	Комуникационни ресурси
Заснемане	Ниски (крайни устройства)	Високи (локална мрежа)
Обработка	Високи (облачни ресурси)	Високи (вътрешна инфраструктура)
Предаване	Умерени до високи	Високи (вътрешна инфраструктура)
Визуализация	Ниски (крайни устройства, HMD)	Ниски до умерени (безжични връзки)

2.3 Предизвикателства при предаването на 3D съдържание

Системите за заснемане, предаване и визуализация на тримерно съдържание се характеризират с изключително високи изисквания към обема на данните и закъсненията при предаване, което поражда специфични предизвикателства в различните слоеве на операционния модел [4]. При заснемането, основен проблем е възникването на трафичното претоварване в рамките на локалната мрежа [A3]. Дори при сравнително ограничен брой сензори, генерираният поток от данни може да достигне порядъка на 10^9 бита в секунда. Например, при три сензора с честота 30 кадъра в секунда, обемът на данните надвишава 2 Gbps [A3]. В този контекст използването на надеждни транспортни протоколи като TCP води до допълнителни ефекти, като повторно предаване на пакети и контрол на претоварването, които могат да увеличат закъснението и да доведат до вариации в честотата на постъпване на кадрите. Това води до несъответствие между генерирания и реално обработваемия поток от данни и налага внимателно управление на буферирането и синхронизацията.

При визуализацията, се наблюдава противоположен, но също толкова критичен проблем. Крайните устройства (напр. HMD) разполагат с ограничена комуникационна свързаност и често използват безжични връзки с ограничена честотна лента. В същото време изискванията към качеството на възпроизвежданото съдържание и закъсненията са изключително строги, като допустимото закъснение е от порядъка на десетки милисекунди [4]. Това създава фундаментален конфликт между необходимостта от пренос на големи обеми 3D данни, ограниченията на комуникационния канал и ниския изчислителен капацитет на крайните устройства.

В обобщение, системите за 3D съдържание са ограничени едновременно от претоварване в локалните мрежи в етапа на заснемане и от ограничена пропускателна способност в мрежата на крайните устройства при визуализация. Това налага използването на ефективни методи за компресия и адаптивно предаване, които да минимизират комуникационния товар при запазване на изискванията за ниско закъснение.

2.1.4 Място и интеграция на обучаемите методи за кодиране

Както беше установено в Глава 1, обучаемите методи за кодиране постигат по-висока ефективност спрямо класическите подходи чрез адаптиране към статистическата структура на данните. Това е особено важно при 3D съдържанието, където зависимостите са сложни и трудно могат да се дефинират с аналитични модели. Тази ефективност, обаче, обикновено се постига за сметка на повишена изчислителна сложност, която в общия случай нараства с мащаба на модела [68]. В рамките на операционния модел от [A3], интеграцията на такива методи следва да се разглежда като задача за съвместна оптимизация между изчислителни и комуникационни ресурси. По-конкретно, методите за кодиране могат да бъдат

разглеждани като механизъм за намаляване на комуникационния товар чрез използване на допълнителен изчислителен ресурс. Практическата им приложимост зависи от това дали необходимата сложност може да бъде поддържана в съответния слой на системата.

Следователно, изборът и разположението на обучаеми кодери не са универсални, а зависят от ресурсните ограничения на конкретния слой (устройства за заснемане, облачна инфраструктура, крайни устройства за визуализация). В този смисъл интеграцията на обучаеми методи е специфична за всеки слой и изисква съобразяване както с наличния изчислителен капацитет, така и с ограниченията върху пропускателната способност.

2.4 Изводи

В настоящата глава беше въведен унифициран операционен модел на системи за заснемане, предаване и визуализация на тримерно съдържание, базиран на [A3], който позволява систематичен анализ на компромиса между изчислителни и комуникационни ресурси. Показано беше, че този компромис не е равномерно разпределен по слоевете на системата, като слоят за визуализация се характеризира едновременно с ограничена пропускателна способност и ограничен изчислителен капацитет. Това го превръща в критична точка за интеграция на ефективни методи за кодиране.

В контекста на слоя за заснемане беше анализирана компресията на Red-Green-Blue-Depth (RGB-D) данни чрез оцветяване и използване на класически схеми за кодиране на изображения [A4], като беше показано, че чрез подходящи предварителни преобразования и мултиплексиране може да се постигне съществено намаляване на комуникационния товар при минимални изчислителни разходи. Това демонстрира практическа стратегия за адаптиране на кодиращите методи към ограниченията на конкретния слой.

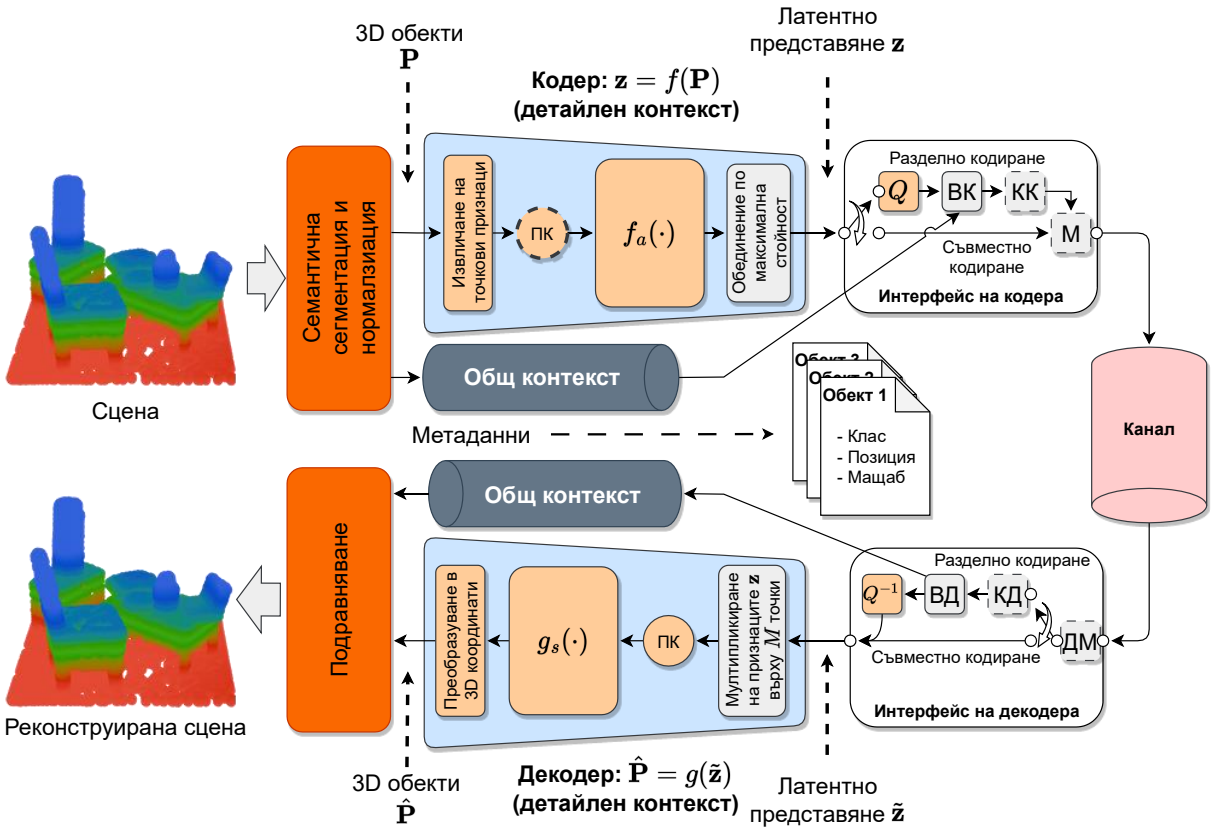
Накрая беше формулиран теоретичен проблем за кодиране на източника в слоя за визуализация, свързан с използването на несъгласувани вероятностни модели. Показано беше, че класическото ентропийно кодиране ограничава сложността на модела от ресурсите на декодера, което противоречи на желаната системна архитектура. В този смисъл беше дефинирана постановка, при която кодерът използва по-точен модел, а несъответствието се компенсира чрез странична информация, като беше изведено основното условие за ефективност на такава схема. Това поставя основата за по-нататъшно изследване на връзката между точност на модела, сложност и необходима скорост на предаване.

ГЛАВА 3. АВТОЕНКОДЕРНИ АРХИТЕКТУРИ ЗА КОДИРАНЕ НА ГЕОМЕТРИЧНАТА СТРУКТУРА НА РАЗРЕДЕНИ ОБЛАЦИ ОТ ТОЧКИ

Програмните реализации на архитектурите и методите от Глава 3, могат да бъдат намерени в [B2]: https://github.com/Teleinfrastructure-Research-Lab/aepcc

Настоящата глава разглежда системата за кодиране на геометричната структура на разредени облаци от точки, предложена в [A5] (виж Фиг. 3.1), която е реализирана в смисъла на концептуалната рамка, въведена в раздел 1.7. В частност, системата използва виртуален канал за предаване на общ контекст, както е формулиран в раздел 1.15 и приложен към семантични сценарии в раздел 1.7.3. За разлика от класическите подходи за кодиране на облаци от точки, като тези в G-PCC, които разчитат на пространствено разделяне чрез блокове и срезове (виж раздел 1.4.2), разглежданата система прилага семантична сегментация върху сцената. Това позволява използването на интерпретируемо

междинно представяне на общия контекст, описващо разположението на обектите в тримерното пространство. По този начин сцената се моделира като съвкупност от семантично значими обекти, вместо като равномерно дискретизирано пространство. След сегментацията всеки обект се нормализира чрез сферична нормализация, като параметрите на нормализацията се съхраняват като метаданни (клас, позиция и мащаб на обекта). За описанието на детайлния контекст, характеризиращ външния вид и геометрията, нормализираните обекти се подават към автоенкодер, който генерира неинтерпретируемо междинно представяне под формата на латентен вектор. Заедно с метаданните тези вектори формират общото междинно представяне, структурирано като множество обектни записи, състоящи се от латентен вектор, позиция и мащаб на обекта в сцената. В [A5] и [A6] са предложени различни автоенкодерни архитектури, които работят върху геометричната структура на разредени облаци от точки. Те ще бъдат анализирани и сравнени в настоящата и следващите глави.



Фигура 3.1: Блокова схема на системата от [A5].

Q – квантуване; ВК – вторичен кодер; КК – канален кодер; М – модулатор; ДМ – демодулатор; КД – канален декодер; ВД – вторичен декодер.

За нормализиран облак от точки $\boldsymbol{P} = \{\boldsymbol{p}_n\}_{n=1}^{N}, \boldsymbol{p}_n \in \mathbb{R}^3$, представящ геометричната структура на отделен обект, автоенкодерът реализира анализиращо (право) преобразуване $\boldsymbol{z} = f(\boldsymbol{P})$ и синтезиращо (обратно) преобразуване $\widehat{\boldsymbol{P}} = g(\boldsymbol{z})$. Всички разглеждани архитектури следват общ, нейерархичен (плосък) принцип на работа (подобен на този в [76]). На първия етап координатите на точките се използват за извличане на начални точкови характеристики. В зависимост от конкретната реализация може да бъде приложено позиционно кодиране върху тези характеристики. След това се моделират зависимостите между точките чрез функция f_a, която обогатява представянето с междуточков контекст,

извлечен въз основа на свързаността и топологията на облака от точки. Тази свързаност може да бъде дефинирана експлицитно (напр. чрез kNN граф), или да бъде имплицитно извлечена чрез механизми за внимание (attention) [44], които адаптивно моделират зависимостите между точките спрямо техните характеристики. Накрая се прилага операция на глобално агрегиране (напр. обединяване по максимална стойност), чрез която се получава един глобален латентен вектор z, представящ целия обект. В декодера този глобален вектор се репликира M пъти, и върху получените M точки с еднакви признаци се прилага позиционно кодиране. След това се използва функция g_s, която трансформира позиционно кодираните характеристики, и накрая те се преобразуват обратно в тримерни координати. Възстановените обекти се мащабират и позиционират в сцената чрез използване на общия контекст.
В [A5] са разгледани два основни работни режима за използване на латентното представяне. Първият е класическият подход на разделно кодиране, при който латентният вектор z се квантува, след което се прилага вторично кодиране (напр. ентропийно или речниково кодиране). Този режим се разглежда в Глава 4. В рамките на този сценарий могат да бъдат включени и канално кодиране и модулация за предаване през комуникационен канал. Вторият режим е JSCC, при който латентният вектор z се използва директно за модулация на носещ сигнал и се предава без квантуване. Вторият режим е разгледан в Глава 5.

В обобщение, настоящата глава се фокусира върху анализа и сравнението на различни автоенкодерни архитектури, стратегиите за тяхното обучение и изчислителна им сложност, с цел идентифициране на подходящи класове функции и архитектурни решения за ефективно извличане на значими признаци от облаци от точки. Тези изследвания са направени в рамките на унифицирана система за кодиране на геометричната структура, при която се комбинират интерпретируемо представяне на структурата на сцената и компактни латентни описания на отделните обекти. Системата, илюстрирана на Фиг. 3.1, накратко ще бъде наричана Autoencoder-based Point Cloud Coding (AEPCC), както в [A5].

3.6 Обучение

В [A5] и [A6] всяка архитектура се обучава при три различни размерности на латентното пространство: 128, 256 и 512. Тази вариация позволява да се изследва влиянието на капацитета на латентното пространство върху качеството на реконструкцията и поведението при наличие на шум в канала. Всички модели от [A5] (това са FoldingNet, Graph Autoencoder (GAE) и Transformer Graph Autoencoder (TGAE)) са обучени и валидирани върху синтетичната база данни **SYNTH**, описана в раздел 3.5. За разлика от тях, в [A6], архитектурите се обучават върху базата данни, въведена в [79], която е разделена на обучаваща, валидационна и тестова извадка в съотношение 80%:10%:10%. С цел намаляване на неопределеността на задачата в [A6], глобалната ротация на обектите е премахната, така че всички модели да бъдат ориентирани по един и същи начин. Общият брой параметри на кодера и декодера за всяка архитектура, разгледана в тази глава от дисертационния труд, е обобщен в Таблица 3.1. В [A5], както и в [A6], като функция на загубата се използва Chamfer разстоянието, изчислявано между оригиналния и реконструирания облак от точки $\boldsymbol{P}$ и $\widehat{\boldsymbol{P}}$, както е дадено в (3.11), където $N = |\boldsymbol{P}|$ и $M = |\widehat{\boldsymbol{P}}|$.

Таблица 3.1: Брой параметри за кодера и декодера при различните архитектури и време за обучение.

Арх.	**Модул**	**F = 128**	**F = 256**	**F = 512**	**Време за обучение**
FoldingNet	Кодер	100,8K	282K	939,5K	6,11ч./6,91ч./8,28ч.
	Декодер	68,6K	268,3K	1M	
GAE	Кодер	767,8K	2M	5,9M	21,33ч./28,46ч./60,06ч.
	Декодер	939,8K	2,6M	7,9M	
TGAE	Кодер	1,1M	4,2M	16,5M	17,13ч./30,38ч./111,18ч
	Декодер	1,5M	5,8M	23,1M	
SEPT	Кодер	2,3M	2,3M	2,3M	4,21ч./5,67ч./6,50ч.
	Декодер	17,8M	17,8M	17,8M	
DPCT	Кодер	5,4M	5,4M	5,6M	6,17ч./6,14ч./6,20ч.
	Декодер	74,5M	74,6M	74,7M	

Всички архитектури са обучени върху графичен процесор RTX 4090 24GB, с изключение на TGAE при F = 512, който поради ограничения в паметта е обучен върху NVIDIA RTX A6000 ADA 48GB. Времената за обучение за FoldingNet, GAE и TGAE са измерени върху SYNTH [A5], а за SEPT и DPCT — върху набора от данни от [79], аналогично на [A6].

$$L_{CD}(\boldsymbol{P}, \widehat{\boldsymbol{P}}) = \frac{1}{N}\sum_{\boldsymbol{p}\in\boldsymbol{P}} \min_{\widehat{\boldsymbol{p}}\in\widehat{\boldsymbol{P}}} |\boldsymbol{p}-\widehat{\boldsymbol{p}}|^2 + \frac{1}{M}\sum_{\widehat{\boldsymbol{p}}\in\widehat{\boldsymbol{P}}} \min_{\boldsymbol{p}\in\boldsymbol{P}} |\widehat{\boldsymbol{p}}-\boldsymbol{p}|^2 \tag{3.11}$$

В [A5] всички модели са обучени с оптимизатора Adaptive Moment Estimation (ADAM) в продължение на 300 епохи. Скоростта на обучение е зададена на 10^{-4}, с коефициент на затихване на теглата 10^{-6}. Използван е планировчик за намаляване на скоростта на обучение, който прилага стъпково намаляване на всеки 60 епохи с коефициент 0.5. Размерът на партидата е 64. Предвид това, че някои облаци от точки са допълнени с нули, се използва бинарна маска на ниво точки (описана в раздел 3.5), чрез която се изключват допълнените точки от изчисляването на загубата. За GAE архитектурата маскирането се прилага както върху входа, така и върху изхода, тъй като моделът запазва съответствие между входните и изходните точки, както е описано в раздел 3.3.

В [A6] обучението се извършва по сходен, но различен начин. При Dynamic Point Cloud Transmission (DPCT) архитектурата, се използва същия размер на партидата, но обучението се параметризира с различни стойности на нивото на шум в канала $\mathrm{SNR}_{\text{об.}} \in \{0\,\mathrm{dB}, 5\,\mathrm{dB}, 10\,\mathrm{dB}\}$. При обучението на архитектурите FoldingNet [76] и Semantic Point Cloud Transmission (SEPT) [57] в [A6] се използва начална скорост на обучение 10^{-3}, докато за DPCT се използва 10^{-4}, което отразява по-високата сложност на архитектурата и необходимостта от по-стабилна оптимизация. Обучена е и една допълнителна вариация на DPCT с фазово-инвариантно декодиране, разгледано в раздел 5.4, като там се използват $\mathrm{SNR}_{\text{об.}} = 5\mathrm{dB}$ и $F = 512$. В [A6] всички архитектури са обучават за 200 епохи.

Кривите на Chamfer загубата при обучение и валидация за архитектурите от [A5] са показани в раздел 3.6 на дисертационния труд. Всички модели демонстрират стабилна сходимост, като наблюдаваните разлики в поведението на загубата корелират както с капацитета на модела, така и с особеностите на архитектурата и условията на обучение. За експериментите в следващите глави се използват теглата от епохата с минимална валидационна загуба.

Взимайки предвид трите архитектури, разгледани в [A5] и трите размерностти на латентния вектор, при които те се обучават, в [A5] са обучени общо 9 различни модела. В [A6] се обучават три архитектури (FoldingNet, SEPT [57] и DPCT), отново с три вариации на

размерността на латентното пространство и три вариации на отношението сигнал-шум (Signal-to-Noise Ratio) (SNR) в канала - $SNR_{об.}$. Тоест в [A6] са обучени общо 27 модела, плюс един допълнителен модел, който изследва прилагането на фазово-инвариантно декодиране, разгледано в Глава 5. Тези модели ще бъдат използвани за експерименти при компресия и дълбоко съвместно кодиране източник-канал (Deep Joint Source-Channel Coding) (DJSCC) на облаци от точки в следващите глави.

3.6 Изводи

В настоящата глава бяха разгледани автоенкодерни архитектури за кодиране на геометричната структура на разредени облаци от точки в рамките на системата AEPCC. Показано беше, че общата постановка, базирана на семантична сегментация, интерпретируемо представяне на общия контекст и компактни латентни описания на отделните обекти, позволява обединяване на различни архитектурни подходи в единна система за компресия и предаване на 3D съдържание.

Анализираните архитектури обхващат широк спектър от методи за обработка на облаци от точки - от FoldingNet-базирани декодери с геометрично позиционно кодиране, през граф-конволюционни автоенкодери, до хибридни архитектури със самовнимание и 3D конволюции. Това позволява да се изследва влиянието на различни механизми за моделиране на междуточковите зависимости, различни форми на позиционно кодиране и различни стратегии за декодиране върху качеството на реконструкцията, сложността и приложимостта им в задачи по компресия и DJSCC на облаци от точки.

ГЛАВА 4. КОМПРЕСИЯ НА РАЗРЕДЕНИ ОБЛАЦИ ОТ ТОЧКИ, ЧРЕЗ АВТОЕНКОДЕРНИ АРХИТЕКТУРИ

Програмните реализации на методите от Глава 4, могат да бъдат намерени в [B2]: https://github.com/Teleinfrastructure-Research-Lab/aepcc

Настоящата глава разглежда използването на автоенкодерните архитектури, анализирани в Глава 3, в класическия режим на разделно кодиране. В този сценарий автоенкодерът изпълнява ролята на обучаемо преобразуване, което извлича компактен латентен вектор z, описващ геометричната структура на обекта. В този смисъл, разглежданата постановка съответства на класическата схема за кодиране на визуално съдържание при която се комбинират три основни етапа: (обучаемо) преобразуване, квантуване и ентропийно или речниково кодиране. По смисъла на блоковата схема от Фиг. 3.1, тази глава се фокусира върху блоковете за квантуване и вторично кодиране на латентното представяне, без да се разглеждат каналното кодиране и модулацията.

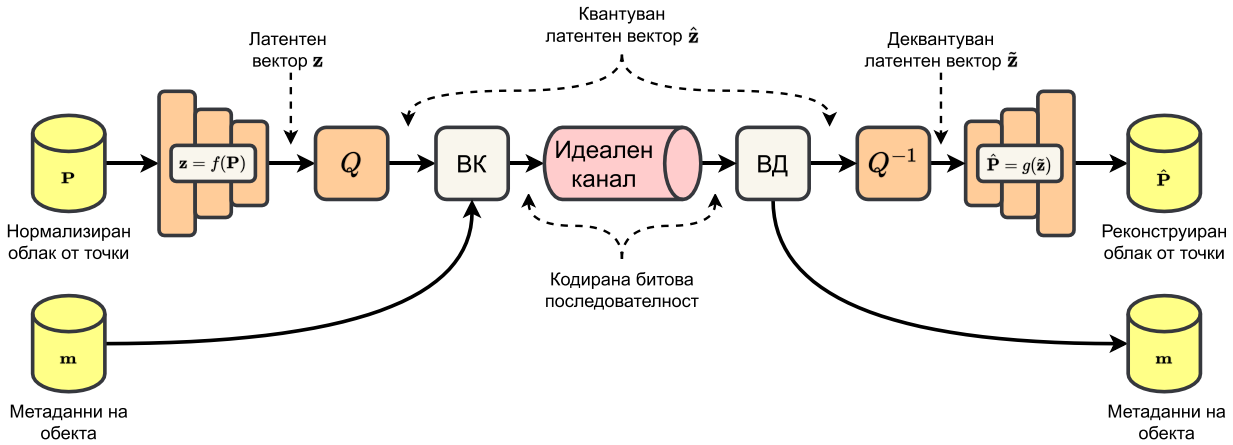
Основният въпрос, разглеждан в настоящата глава, е доколко латентните представяния, извлечени от различните архитектури, са подходящи за ефективна компресия на геометрията. За тази цел се анализира влиянието на сложността на архитектурата, размерността на латентното пространство и стъпката на квантуване върху компромиса между скорост и изкривяване. В рамките на тази глава се приема, че каналът за предаване е идеален, така че грешки при пренос не се разглеждат. По този начин анализът е фокусиран изцяло върху компресия (кодиране на източника) на геометричната структура. Това позволява ясно да се отдели ефектът от квантуването на z от ефектите, свързани с предаването през канал с шум, които са предмет на Глава 5.

По-конкретно, главата разглежда квантуването на латентните вектори, формирането на сериализирано междинно представяне и прилагането на вторично кодиране върху получените дискретни символи. Получените резултати се оценяват чрез зависимости скорост -изкривяване, които позволяват сравнение както между различните автоенкодерни архитектури, така и спрямо класически подходи за кодиране на геометрията.

4.1 Постановка на задачата при разделно кодиране

Схема на разглежданата работна постановка е показана на Фиг. 4.1. Нека $\boldsymbol{P} = \{\boldsymbol{p}_n\}_{n=1}^{N}, p_n \in R^{\mathbb{3}}$, обозначава нормализиран облак от точки, представящ геометричната структура на отделен обект. Както беше разгледано в Глава 3, кодерът реализира обучаемо преобразуване $\boldsymbol{z} = f(\boldsymbol{P})$, където $\boldsymbol{z} \in R^F$ е латентният вектор, а F е размерността на латентното пространство. В общия случай $\mathbf{z}$ съдържа непрекъснати стойности и не може директно да бъде представен чрез краен битов поток. Поради това при разделното кодиране се въвежда етап на квантуване, чрез който се формира дискретно латентно представяне $\hat{\boldsymbol{z}} = Q(\boldsymbol{z})$, където $Q(\cdot)$ е оператор на квантуване. Полученото дискретно представяне $\hat{\boldsymbol{z}}$ може да бъде сериализирано и представено чрез компактен битов поток. В този смисъл автоенкодерът дефинира преобразуването $\boldsymbol{P} \mapsto \boldsymbol{z}$, а последващите етапи осигуряват представяне на $\mathbf{z}$ в дискретна форма, подходяща за ефективно вторично кодиране. Декодерът реконструира геометричната структура чрез $\widehat{\boldsymbol{P}} = g\big(Q^{-1}(\hat{\boldsymbol{z}})\big)$, където $g(\cdot)$ е синтезиращото преобразуване.

В разглеждания сценарий скоростта зависи от три основни фактора: размерността F на латентния вектор, стъпката на квантуване Δ и ефективността на вторичното кодиране. Увеличаването на F обикновено повишава капацитета на представянето и може да подобри качеството на реконструкцията, но едновременно с това увеличава броя кодирани символи.



Фигура 4.1: Схема на разглежданата работна постановка.

Намаляването на Δ води до по-прецизно представяне на латентните вектори, но обикновено увеличава ентропията на символния поток и съответно необходимата скорост. От своя страна, вторичното кодиране елиминира информационния излишък в $\hat{\boldsymbol{z}}$, за да намали дължината на битовото представяне без допълнителна загуба на информация.

Важна особеност на разглежданата постановка е, че се приема идеален канал, т.е. след вторичното декодиране се възстановява точно същият квантуван латентен вектор $\hat{\boldsymbol{z}}$, който е бил формиран в кодера. Следователно единствените източници на загуба в тази глава са неидеалностите на автоенкодерната архитектура и квантуването на латентното

представяне, а не грешките при предаване. Това позволява експериментално да се анализира доколко различните автоенкодерни архитектури водят до латентни представяния, които са едновременно компактни и устойчиви на квантуване.

По смисъла на системата от Фиг. 3.1, за всеки обект се кодира двойката $(\hat{\boldsymbol{z}}, \boldsymbol{m})$, където **m** обозначава съпътстващите метаданни, свързани с позицията, мащаба и класа на обекта. В настоящата глава основният фокус е върху компресията на латентния вектор $\hat{\boldsymbol{z}}$, тъй като именно той носи основната информация за детайлната геометрична структура. Метаданните се разглеждат като допълнително сериализирано описание, необходимо за подреждане на обектите в сцената, но не променят основните зависимости между размерността на латентното пространство, квантуването и качеството на реконструкцията. На тази основа в следващите раздели се анализират конкретните решения за квантуване и вторично кодиране, както и получените зависимости скорост -изкривяване за разгледаните в Глава 3 архитектури.

4.6 Резултати при компресия на геометричната структура на 3D сцени

Резултатите при компресия на сцени, докладвани в [A5], са получени в два експериментални сценария. В първия се анализират характеристиките на скорост-изкривяване (Rate-Distortion) (R-D) върху сцени от базата **REAL**. Наблюдаваните зависимости са подобни на тези при компресия на отделни обекти (раздел 4.5), като се потвърждават следните основни тенденции: (i) TGAE демонстрира най-добра R-D ефективност; (ii) размерът на модела оказва съществено влияние при силно квантуване, но влиянието му намалява при по-високи скорости; и (iii) най-добрите резултати не се постигат непременно от най-големите модели.

Във втория експеримент, R-D характеристиките се измерват върху четири самостоятелни сцени — две от базата SceneNN и две, заснети от авторите на [A5]. Количествените и качествените резултати са представени в раздел 4.6 на дисертационния труд. Представени са входните сцени, както и реконструкциите, получени от различните методи, заедно със съответните R-D криви за всяка сцена. От представените резултати се вижда, че нито Draco, нито G-PCC достигат ниските скорости, постигнати от AEPCC, независимо от използваната архитектура. Поради ограниченото припокриване между AEPCC и класическите кодеци в нискоскоростния режим, Bjøntegaard Delta PSNR (BD-PSNR) не може да бъде надеждно изчислен. Вместо това, при наличие на достатъчно припокриване по отношение на пиковото отношение сигнал-шум (Peak Signal-to-Noise Ratio) (PSNR), се използва метриката Bjøntegaard Delta Rate (BD-Rate) с G-PCC като референтен метод, като резултатите са представени в Таблица 4.3.

4.7 Изводи

В настоящата глава беше изследвано използването на разгледаните в Глава 3 автоенкодерни архитектури в класически режим на разделно кодиране, при който латентното представяне се квантува и подлага на вторично кодиране. Показано беше, че автоенкодерните архитектури формират латентни представяния, които могат да бъдат ефективно компресирани, като компромисът скорост-изкривяване зависи едновременно от архитектурата, размерността на латентното пространство и стъпката на квантуване. Резултатите показват, че при компресия както на ниво отделен обект, така и на ниво сцена, архитектурата TGAE постига най-добра R-D ефективност, като същевременно демонстрира добра способност за генерализация при преминаване от синтетични към реални данни.

Резултатите при компресия на сцени показват, че предложеният подход AEPCC превъзхожда класическите кодиращи схеми като Draco и G-PCC и обучаемия метод CRCIR в нискоскоростния режим при разредени облаци от точки, особено при архитектурите GAE и TGAE. Това потвърждава, че обучаемите преобразования са ефективен подход за компресия на геометричната структура на разредени облаци от точки.

Таблица 4.3: BD-Rate сравнение на кодиращите схеми, като за референтен кодек е използван G-PCC.

		Сцена 1		Сцена 2		Сцена 3		Сцена 4		Средно	
Кодек	F	BD-Rate [%]	Време [s]	BD-Rate [%]	Време [s]	BD-Rate [%]	Време [s]	BD-Rate [%]	Време [s]	BD-Rate [%]	Време [s]
FoldingNet	128	-85.69	0.30 ± 0.006	-83.01	0.25 ± 0.004	-72.69	0.51 ± 0.011	-82.88	0.62 ± 0.012	-81.07	0.42 ± 0.151
FoldingNet	256	-81.53	0.30 ± 0.004	-77.23	0.25 ± 0.004	-68.15	0.51 ± 0.009	-74.94	0.63 ± 0.015	-75.46	0.42 ± 0.156
FoldingNet	512	-69.59	0.30 ± 0.006	-73.90	0.25 ± 0.004	22706	0.52 ± 0.013	-62.25	0.63 ± 0.020	-50.53	0.42 ± 0.158
GAE	128	-87.45	0.45 ± 0.006	-74.98	0.34 ± 0.005	-77.59	0.83 ± 0.011	-84.54	1.05 ± 0.014	-81.14	0.67 ± 0.289
GAE	256	-81.99	0.46 ± 0.005	-66.24	0.35 ± 0.006	-71.02	0.86 ± 0.011	-76.97	1.09 ± 0.013	-74.06	0.69 ± 0.299
GAE	512	26207	0.47 ± 0.007	-52.58	0.36 ± 0.006	-53.62	0.91 ± 0.012	-62.35	1.15 ± 0.016	-39.46	0.72 ± 0.323
TGAE	128	-87.81	0.39 ± 0.006	-86.34	0.32 ± 0.004	-81.09	0.66 ± 0.012	-84.51	0.81 ± 0.015	-84.94	0.55 ± 0.201
TGAE	256	-76.29	0.40 ± 0.006	-82.81	0.33 ± 0.005	-74.20	0.69 ± 0.015	-77.32	0.85 ± 0.019	-77.65	0.57 ± 0.214
TGAE	512	-73.35	0.43 ± 0.009	-73.13	0.34 ± 0.008	-58.79	0.79 ± 0.016	-63.62	0.99 ± 0.020	-67.22	0.64 ± 0.265
CRCIR	—	-34.34	0.04 ± 0.003	12571	0.20 ± 0.006	-28.74	0.06 ± 0.013	-45.59	0.09 ± 0.025	-25.58	0.10 ± 0.066
Draco	—	-33.23	0.54 ± 0.166	-11.83	0.27 ± 0.065	0.12	1.25 ± 0.416	-25.70	1.69 ± 0.597	-17.66	0.94 ± 0.674
G-PCC	—	—	0.71 ± 0.374	—	0.33 ± 0.132	—	2.22 ± 1.371	—	0.97 ± 0.736	—	1.11 ± 1.096

Клетките, оцветени в тъмносиво, показват най-добрия BD-Rate за съответната сцена, а светлосивите клетки обозначават втория най-добър. Времената за обработка (най-доброто е подчертано) са измерени върху система с AMD Ryzen 9 7950X и 64 GB RAM и NVIDIA RTX 4090.

ГЛАВА 5. ДЪЛБОКО СЪВМЕСТНО КОДИРАНЕ ИЗТОЧНИК-КАНАЛ ЗА ОБЛАЦИ ОТ ТОЧКИ

Програмните реализации на методите от Глава 5, могат да бъдат намерени в [B3]: https://github.com/Teleinfrastructure-Research-Lab/lb-dpct

В предходната глава беше разгледана задачата за компресия на облаци от точки чрез обучаеми автоенкодери в рамките на класическия подход за разделно кодиране. В този режим компресията на сцените се реализира чрез квантуване на латентния вектор и вторично кодиране, като в четвърта глава беше разгледано предаването на получената битова последователност през идеален канал без шум. При наличие на реален комуникационен канал, в който присъства шум, е необходимо въвеждането на допълнително канално кодиране, с цел защита на предаваната информация от грешки. Класическият подход за разделно кодиране е теоретично оптимален при безкрайни дължини на блоковете и при стационарни канални условия, съгласно теоретичната постановка изложена в раздел 1.6 на дисертационния труд. В практическите системи, обаче, където се работи с крайни дължини на кодовите думи и при наличие на динамично променящи се канали, този подход води до съществени ограничения. Най-характерното от тях е т.нар. *cliff-effect*, при който при малко влошаване на SNR настъпва рязък спад в качеството на реконструкцията, което е особено неблагоприятно при предаване на 3D съдържание в реално време, където плавната деградация е критично изискване. Този недостатък на разделното кодиране се проявява именно в режима на крайни дължини на

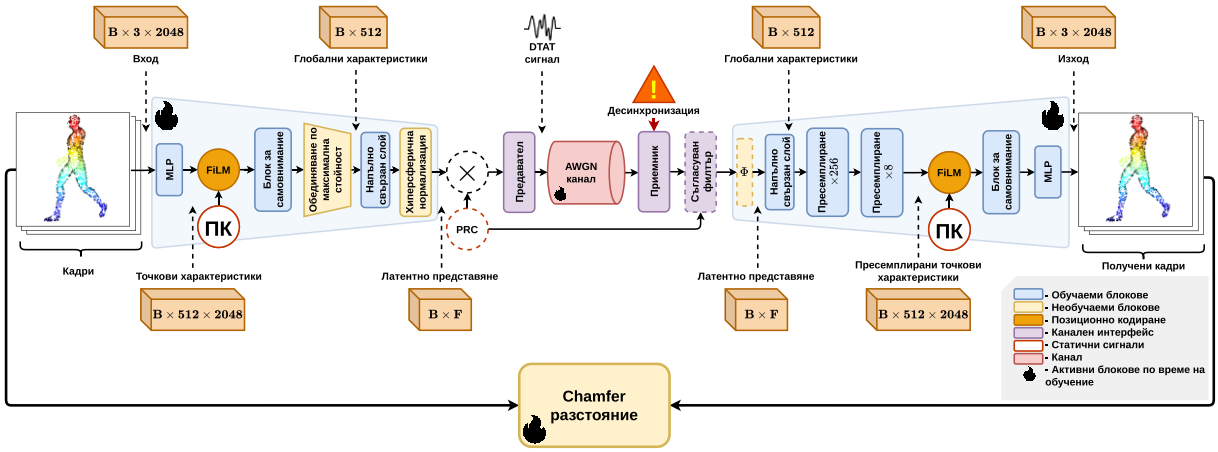
блоковете, където JSCC представлява по-подходяща алтернатива [A5, 57, 55]. В този контекст, системата AEPCC, въведена в [A5] и разгледана в Глава 3, поддържа JSCC/DJSCC режим на работа, при който латентните представяния могат да бъдат използвани директно за модулация на носеща и предаване през канал с шум, без необходимост от квантуване и канално кодиране.

Предимствата на JSCC подхода са особено значими при динамични 3D данни, които се превръщат в ключов тип съдържание за приложения като разширена реалност (Extended Reality) (XR), телеприсъствие [A3, 4] и мобилна роботика, но чието предаване през безжични канали остава предизвикателство поради високата им размерност и сложност [4]. Класическите подходи с разделно кодиране страдат от *cliff-effect* и ограничена адаптивност в режимите с къси блокове, характерни за съвременните приложения [92]. За разлика от тях, DJSCC реализирано чрез директно предаване на непрекъснати латентни представяния, осигурява плавна деградация на качеството и по-голяма гъвкавост спрямо каналните условия. В тази глава се анализира чувствителността към шум на автоенкодерите, предложени в [A5] и разгледани в Глава 3, а именно FoldingNet, GAE и TGAE, като се изследва тяхното поведение при наличие на адитивен бял гаусов шум (Additive White Gaussian Noise) (AWGN) в латентното пространство. Този анализ служи като преход от режима изследван в Глава 4, към реалистична комуникационна постановка. След това се въвежда и формализира експерименталната постановка от [A6], която разглежда поведението на DPCT архитектурата в JSCC режим, при който латентните вектори се използват директно за модулация и предаване през канал с шум. В допълнение се разглеждат специфичните предизвикателства, възникващи при предаването на динамични облаци от точки в JSCC режим, включително проблемите със синхронизацията и чувствителността към времеви измествания, както и предложените в [A6] решения за тяхното преодоляване.

5.2 Съвместно кодиране източник -канал и предаване през канали с шум

При втория режим на работа на AEPCC системата от Фиг. 3.1, а именно JSCC режима, при който компонентите на латентния вектор се изпозлват за модулиране на носеща и предаване през зашумен канал, е изследвана DPCT архитектурата от [A6]. Експерименталната постановка от [A6] е представена на Фиг. 5.3, която в общи линии съответства на втория режим на AEPCC с добавени блокове, изпълняващи конкретни функции свързани с предаването на сигнала през канал с шум. В този режим предаваните символи съответстват на елементите на латентен вектор, генериран от кодера, съгласно архитектурата, описана в раздел 3.4.2 и в предходни работи [76, 57, 95]. Латентният вектор $\boldsymbol{z} \in R^F$ се състои от F реални компонента с амплитуда, която се приема за непрекъсната стойност, а де факто е 32-битово чисто с плаваща запетая. За предаване на z през физически канал се използва Discrete Time Analog Transmission (DTAT) [95]. Нека $\boldsymbol{z} = [z_0, \dots, z_{F-1}]^\top$ обозначава латентния вектор, генериран от DJSCC кодера, като на всеки елемент $\boldsymbol{z}_k$ отговаря символ $a[k]$ (реално число). Символите се разглеждат като импулси, разположени през интервал T, и се оформят чрез Root-Raised-Cosine (RRC) филтър. Полученият DTAT сигнал е формализиран математически в 5.6, където $g_{\mathrm{RRC}}(t)$ е импулсната характеристика на RRC филтъра, $k = 0, \dots, F-1$, а T е символният интервал. Това импулсно оформяне гарантира, че предаваният сигнал е ограничен по честотна лента, като при прилагане на съгласуван RRC филтър в приемника и дискретизация в моментите на символите се минимизира междусимволна интерференция (Inter-Symbol Interference) (ISI):

$$s(t) = \sum_k a[k]\ g_{\mathrm{RRC}}(t - kT) \tag{5.6}$$



Фигура 5.3: Схема на постановката при DPCT [A6].

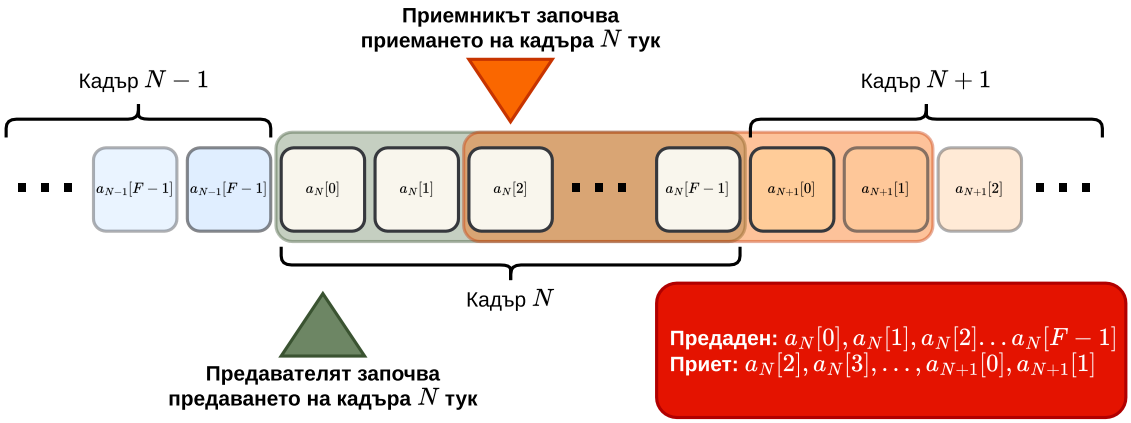
В рамките на тази постановка възникват две основни технически предизвикателства, които се разглеждат в следващите раздели:

1. **Ефективно декодиране в канал с шум:** Предаденият облак от точки трябва да бъде реконструиран от зашумения латентен вектор с минимална загуба на качество. Както беше показано в раздел 5.1 и в [A5], автоенкодерите демонстрират съществена устойчивост към шум в латентното пространство, дори когато отсъства такъв шум по време на обучението. Въпреки това, в [57] е показано, че вкарването на AWGN в латентния вектор по време на обучението подобрява устойчивостта към шум в канала. Също така, използването на съгласуван RRC филтър в приемника максимизира изходното SNR и е оптимално при AWGN канал, което го прави естествен избор за извличане на латентните символи [95].

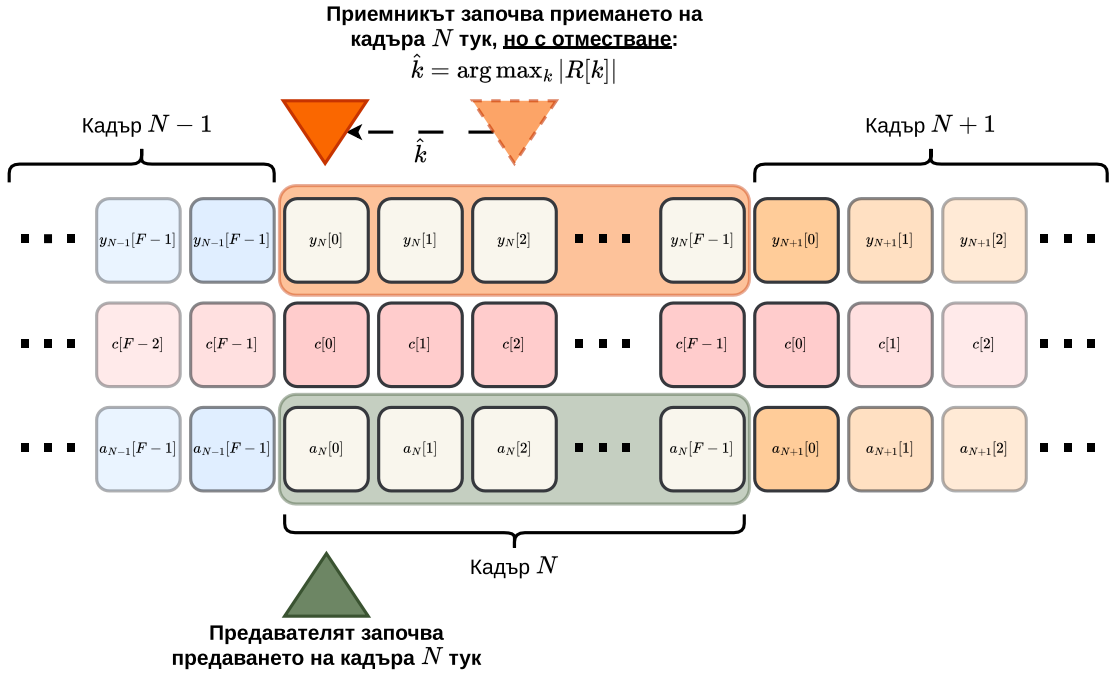
2. **Синхронизация на кадрите:** При предаване на динамични облаци от точки, множество латентни вектори се предават последователно, като всеки от тях съответства на отделен кадър. Приемникът трябва да определи границите между тези последователни вектори, за да може правилно да реконструира отделните кадри. За разлика от цифровите системи, където тази задача обикновено се решава чрез добавяне на заглавия или предварително известни преамбюли [96], при аналоговите DJSCC комуникационни системи подобни структури не са налични. Това прави проблема за синхронизация на кадрите значително по-сложен и той остава слабо изследван в литературата [97]. Визуална илюстрация на този проблем е представена на Фиг. 5.4А, където се показва как десинхронизацията на приемника води до грешно възстановяване на латентния вектор.

5.3 Синхронизация чрез PRC и съгласуван филтър

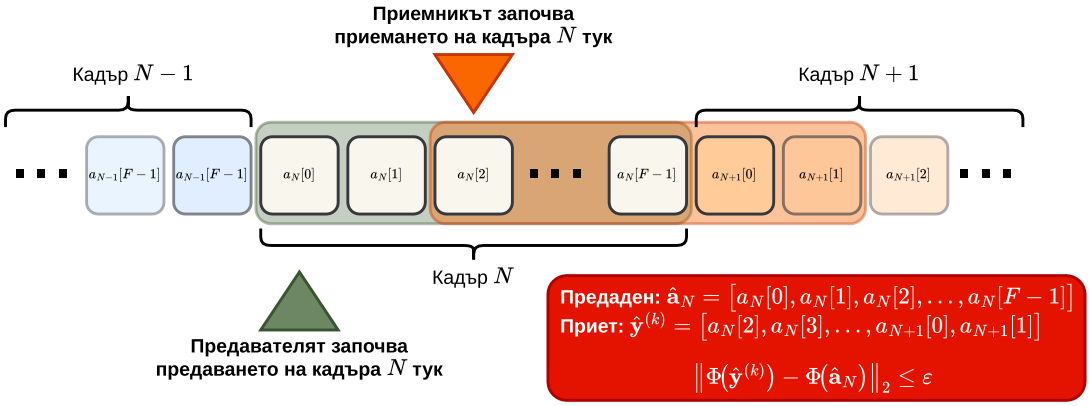
Предложената схема за синхронизация използва псевдослучаен код (Pseudorandom Code) (PRC) и съгласуван филтър за надеждно определяне на началото на кадъра при предаване през AWGN канал. Изходната последователност се умножава поелементно с PRC кода, след което сигналът се подлага на импулсно оформяне чрез RRC филтър и се предава. В приемника полученият сигнал се филтрира, дискретизира и корелира със спрегнатия код в честотната област. Позицията на максимума на корелационната функция определя оценката на времевото отместване на кадъра, което позволява правилно подравняване и възстановяване на синхронизираната последователност. Механизмът, чрез който съгласуваният филтър и корелацията с PRC последователността позволяват възстановяване на синхронизацията, е илюстриран на Фиг. 5.4Б.



(А) Проблемът с десинхронизацията при използване на латентните вектори за модулация и предаване през канал с шум.



(Б) Възстановяване на синхронизацията чрез съгласуван филтър.



(В) Фазово-инвариантно декодиране (PID), при което не се изисква възстановяване на синхронизацията.

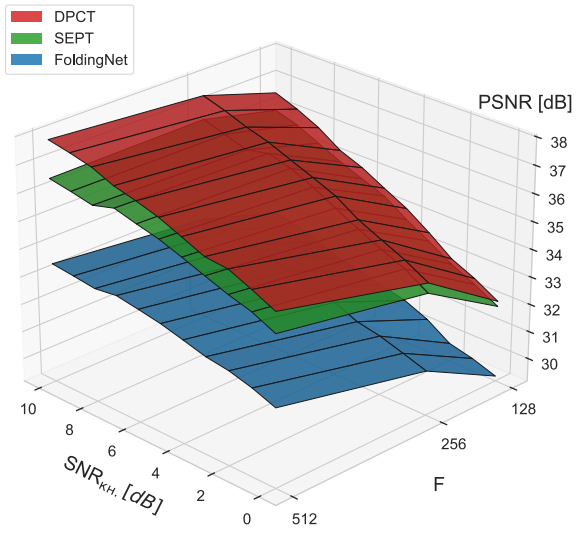
Фигура 5.4: Проблемът с десинхронизацията и предложените решения в [А6].

5.4 Фазово-инвариантно декодиране

Предложената схема за фазово-инвариантно декодиране (виж фиг. 5.4B) премахва зависимостта от времевото отместване чрез използване на амплитудния спектър на дискретно преобразование на Фурие (Discrete Fourier Transform) (DFT), който е инвариантен спрямо циклични измествания. Приетите латентни последователности се преобразуват чрез унитарно DFT, като декодерът работи единствено с амплитудите на спектралните коефициенти. Анализът показва, че дори когато извлеченият блок пресича границата между два последователни кадъра, полученото представяне остава близко до това на целевия кадър при условие за малки междукадрови изменения. Това позволява надеждно възстановяване на облака от точки без необходимост от експлицитна синхронизация, като фазово-инвариантният слой се добавя преди декодера и осигурява устойчивост спрямо десинхронизация между предавателя и приемника.

5.5 Резултати при DJSCC и предаване в канали с шум

В този раздел се разглеждат резултатите от експериментите в [A6], които оценяват качеството на реконструкцията при различни състояния на канала, в рамките на експерименталната постановка, описана в раздел 5.2. На Фиг. 5.5 е представена зависимостта между размера на латентния вектор F, SNR в канала при тестване — $\mathrm{SNR}_{\text{кн.}}$, и PSNR на реконструкцията. За този експеримент $\mathrm{SNR}_{\text{об.}}$ се избира така, че да съвпада максимално с тестовото условие, т.е. $\mathrm{SNR}_{\text{об.}} = \arg\min_{\mathrm{SNR}_{\text{об.}}} |\mathrm{SNR}_{\text{об.}} - \mathrm{SNR}_{\text{кн.}}|$, където $\mathrm{SNR}_{\text{об.}} \in \{0\,\mathrm{dB}, 5\,\mathrm{dB}, 10\,\mathrm{dB}\}$. Както беше описано в раздел 5.2, латентният вектор се разглежда като последователност от реални символи, които формират DTAT сигнал за предаване. Следователно броят на предаваните символи за един кадър от облака от точки — и съответно необходимата честотна лента — нараства с увеличаване на F. Резултатите показват, че качеството на реконструкция се подобрява както при увеличаване на честотната лента (по-голямо F), така и при по-добри канални условия (по-високо $\mathrm{SNR}_{\text{кн.}}$). Сред разглежданите архитектури в [A6], предложеният метод превъзхожда както SEPT, така и FoldingNet за целия диапазон от стойности на F и $\mathrm{SNR}_{\text{кн.}}$. Въпреки че FoldingNet не е първоначално разработен като DJSCC метод, той е включен за пълнота, тъй като демонстрира сравнително добра производителност спрямо броя параметри. Въпреки това, той отстъпва на SEPT и DPCT в повечето конфигурации.



Фигура 5.5: Зависимост на PSNR от честотната лента (F) и $\mathrm{SNR}_{\text{кн.}}$.

5.7 Изводи

В настоящата глава беше разгледано дълбокото съвместно кодиране източник -канал за облаци от точки като алтернатива на класическия подход за разделно кодиране, анализиран в Глава 4. Показано беше, че латентните представяния, генерирани от разгледаните автоенкодери, притежават вградена устойчивост към шум в латентното пространство, което ги прави подходящи както за квантуване, така и за директно предаване през AWGN канал. Резултатите показват, че DJSCC подходите осигуряват плавна деградация на качеството при влошаване на каналните условия и по този начин избягват характерния *cliff-effect*, наблюдаван при класическите цифрови схеми.

Освен това беше показано, че архитектурата DPCT постига по-добра производителност спрямо разгледаните референтни методи в DJSCC режим, като съчетава по-високо качество на реконструкцията с по-добра честотна ефективност [A6]. Допълнително бяха анализирани и два подхода за синхронизация на кадрите при предаване на динамични облаци от точки - чрез PRC и съгласуван филтър, както и чрез фазово-инвариантно декодиране. Резултатите показват, че съгласуваният филтър осигурява практически оптимална синхронизация, докато PID предлага работоспособна алтернатива в случаи на малки междукадрови разминавания и позволява декодиране без изрично възстановяване на синхронизацията.

Същевременно, предложената схема за фазово-инвариантно декодиране по своята същност води до загуба на информация, тъй като премахва фазовата компонента чрез използване на амплитудния спектър на DFT. Това ограничава представителната способност на автоенкодера и намалява капацитета му да моделира зависимостите в данните, което може да доведе до по-ниска граница на достижимото качество на реконструкцията. От друга страна, при практическа реализация на синхронизация чрез съгласуван филтър е необходимо въвеждането на допълнителна постояннотокова съставка в сигнала, която осигурява надеждно откриване на границите на кадрите. Амплитудата на тази съставка зависи от нивото на шума и изискваната надеждност на синхронизацията, като при по-неблагоприятни канални условия е необходимо тя да бъде по-голяма. При наличие на ограничение върху общата предавана мощност това води до намаляване на мощността, отделена за полезния сигнал, и съответно до ефективно влошаване на качеството на реконструкцията. В този смисъл възниква фундаментален компромис между двата подхода: PID избягва необходимостта от постояннотокова съставка, но за сметка на намалена представителна способност поради загуба на фазова информация, докато синхронизацията чрез съгласуван филтър запазва пълната информация, но изисква въвеждането на допълнителна съставка, която при ограничения на мощността намалява ефективния ресурс за кодиране на полезния сигнал. Поради това, в бъдеща работа е необходимо по-задълбочено изследване на този компромис, включително анализ дали намалената представителна способност при PID може да бъде компенсирана от по-ефективното използване на ресурса при реалистични канални условия.

III. ЗАКЛЮЧЕНИЕ И ПРИНОСИ НА ДИСЕРТАЦИОННИЯ ТРУД

Дисертационният труд разглежда ефективното кодиране на тримерно съдържание чрез обучаеми и семантични подходи за системи за заснемане, предаване и визуализация. Предложени са архитектури за компресия и директно предаване на облаци от точки, които използват автоенкодерни архитектури, които подобряват ефективността при ниски скорости и предаване през канали с шум. Анализирани са ограниченията на класическото ентропийно кодиране и е въведена постановка с несъгласувани вероятностни модели, позволяваща използването на по-сложни модели от страната на кодера. Показано е също, че предложените методи осигуряват устойчивост към шум и плавна деградация на качеството при предаване през канал с шум. Като бъдещи направления се очертават разработването на по-ефективни обучаеми архитектури, интеграцията на вероятностни модели за ентропийно кодиране, разширяването към по-сложни динамични сцени и изследването на компромисите при използване на различни методи за синхронизация.

НАУЧНО-ПРИЛОЖНИ И ПРИЛОЖНИ ПРИНОСИ

Научните приноси са:

1. Разработена е теоретична постановка за ентропийно кодиране с несъгласувани вероятностни модели и е изведено условие за ефективност, свързващо ползата от по-точен модел в кодера с необходимата странична информация.

2. Предложен е метод за фазово-инвариантно декодиране (PID) при предаване на динамични облаци от точки, който премахва необходимостта от изрично възстановяване на синхронизацията на кадрите чрез използване на представяне, инвариантно спрямо циклични измествания, за което е изведена граница на грешката.

Научно-приложните приноси са:

1. Извършени са систематизация и анализ на методите за кодиране на 3D съдържание, въз основа на които е предложена таксономия според използваните технологични принципи и интерпретируемостта на междинните представяния (фиг. 1.5). Формулиран е и концептуален модел за семантична компресия чрез разделяне на информацията на глобален и детайлен семантичен контекст.

2. Прилагане на 4-слоен операционен модел (фиг. 2.1) като универсална рамка за анализ на системи за заснемане, предаване и визуализация на тримерно съдържание, с акцент върху разпределението на изчислителните и комуникационните ресурси.

3. Изследван е широк спектър от методи за обработка на облаци от точки, включително FoldingNet-базирани, граф-конволюционни, архитектури с механизми за самовнимание и 3D-конволюционни архитектури, като са предложени архитектурите GAE, TGAE и DPCT. Обучени и сравнени са общо 37 модела при различни размерности на латентното пространство и условия на обучение.

4. Изследвана е компресията на разредени облаци от точки чрез автоенкодерни архитектури, като е анализирано влиянието на архитектурата, броя на параметрите и размерността на латентното пространство върху зависимостта скорост - изкривяване. Извършено е сравнение с G-PCC [11], Draco [90] и Context-based Residual Coding and Implicit neural representation based Refinement (CRCIR) [91], като

резултатите (табл. 4.3) показват, че AEPCC, и по-специално TGAE, постига най-добра R-D ефективност в нискоскоростния режим.

5. Изследвана е DPCT архитектурата в режим на дълбоко съвместно кодиране източник -канал за облаци от точки, като е показано подобрение спрямо референтни методи (SEPT [57], FoldingNet [76] и G-PCC [11]) по отношение на качеството на реконструкцията и честотната ефективност при наличие на шум в предавателния канал (фиг. 5.5). Разработени и сравнени са подходи за синхронизация на кадрите, включително фазово-инвариантно декодиране, като е анализирано поведението им при различни нива на шум и десинхронизация.

Приложните приноси са:

1. Анализ и имплементация на методи за компресия на RGB-D изображения чрез оцветяване и използване на кодиращи схеми за цветни изображения, включително оценка на влиянието на различни стратегии за оцветяване мултиплексиране и семантично-ориентирана обработка върху R-D характеристиките (програмната реализация на методите може да бъде намерена в [B1]).

2. Разработена е програмна реализация на система за кодиране на геометричната структура на разредени облаци от точки (AEPCC), включваща имплементация на автоенкодерни архитектури от различен тип, механизми за позиционно кодиране, процедури за подготовка и разширяване на данните, както и пълен цикъл за обучение, валидация и тестване. Програмната реализация на автоенкодерите от [A5] може да бъде открита в [B2], а за автоенкодера от [A6] - в [B3].

3. Разработена е самостоятелна програмна реализация на кодираща схема за разделно кодиране на геометричната структура в рамките на AEPCC, включваща покомпонентно скаларно квантуване на латентните вектори, деквантуване, пакетиране, сериализация чрез Concise Binary Object Representation (CBOR) и вторично кодиране чрез DEFLATE. Програмната реализация може да бъде намерена в [B2].

ИЗПОЛЗВАНИ СЪКРАЩЕНИЯ

СЪКРАЩЕНИЕ	ОПРЕДЕЛЕНИЕ	СЪКРАЩЕНИЕ	ОПРЕДЕЛЕНИЕ
3D	тримерни	DJSCC	дълбоко съвместно кодиране източник-канал (Deep Joint Source-Channel Coding)
V-PCC	видео-базирана компресия на облаци от точки (Video-based Point Cloud Compression)	R-D	скорост-изкривяване (Rate-Distortion)
G-PCC	геометрично-базирана компресия на облаци от точки (Geometry-based Point Cloud Compression)	BD-Rate	Bjøntegaard Delta Rate
JSCC	съвместно кодиране източник-канал (Joint Source-Channel Coding)	BD-PSNR	Bjøntegaard Delta PSNR
HTC	холографска комуникация (Holographic Type Communication)	PSNR	пиково отношение сигнал-шум (Peak Signal-to-Noise Ratio)
HMD	очила за виртуална или добавена реалност (Head-Mounted Display)	XR	разширена реалност (Extended Reality)
RGB-D	Red-Green-Blue-Depth	AWGN	адитивен бял гаусов шум (Additive White Gaussian Noise)
kNN	k-най-близки съседи (k-Nearest Neighbours)	DTAT	Discrete Time Analog Transmission
AEPCC	Autoencoder-based Point Cloud Coding	RRC	Root-Raised-Cosine
GAE	Graph Autoencoder	ISI	междусимволна интерференция (Inter-Symbol Interference)
TGAE	Transformer Graph Autoencoder	PRC	псевдослучаен код (Pseudorandom Code)
ADAM	Adaptive Moment Estimation	DFT	дискретно преобразование на Фурие (Discrete Fourier Transform)
DPCT	Dynamic Point Cloud Transmission	PID	фазово-инвариантно декодиране (Phase-Invariant Decoding)
SEPT	Semantic Point Cloud Transmission	CRCIR	Context-based Residual Coding and Implicit neural representation based Refinement
SNR	отношение сигнал-шум (Signal-to-Noise Ratio)	CBOR	Concise Binary Object Representation

СПИСЪК НА ПУБЛИКАЦИИТЕ ПО ДИСЕРТАЦИОННИЯ ТРУД

[A1] I. Bozhilov, R. Petkova, K. Tonchev и A. Manolova, „A Systematic Survey Into Compression Algorithms for Three-Dimensional Content," IEEE Access, т. 12, с. 141 604—141 624, 2024. DOI: 10.1109/ACCESS.2024.3469549.

[A2] I. Bozhilov, „Semantic Compression for 3D Content: A Unified Conceptual Framework and Survey of Advanced Solutions," в 2026 Joint International Conference on Digital Arts, Media and Technology with ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunication Engineering (ECTI DAMT & NCON), 2026, с. 710—715. DOI: 10.1109/ECTIDAMTNCON67592.2026.11459979.

[A3] I. Bozhilov, R. Petkova, K. Tonchev, A. Manolova и V. Poulkov, „HOLOTWIN: A Modular and Interoperable Approach to Holographic Telepresence System Development," Sensors, т. 23, №21, 2023, iSSN: 1424-8220. DOI: 10.3390/s23218692. url: https://www.mdpi.com/1424-8220/23/21/8692.

[A4] I. B. Bozhilov, R. R. Petkova, K. T. Tonchev и A. H. Manolova, „Exploring Semantic-Aware Compression of RGBD Images Using Conventional Codecs," в 2025 60th International Scientific Conference on Information, Communication and Energy Systems and Technologies (ICEST), IEEE, 2025, с. 1—4.

[A5] I. Bozhilov, R. Petkova, K. Tonchev, A. Manolova, V. Poulkov и H. Vincent Poor, „Autoencoder Architectures for Low-Rate Sparse Point Cloud Geometry Coding," IEEE Access, т. 13, с. 214 122—214 140, 2025. DOI: 10.1109/ACCESS.2025.3646031.

[A6] I. Bozhilov, R. Petkova, K. Tonchev и A. Manolova, „Learning-Based Dynamic Point Cloud Transmission," в 2025 28th International Symposium on Wireless Personal Multimedia Communications (WPMC), 2025, с. 1—6. DOI: 10.1109/WPMC67460.2025.11351024.

ПРОГРАМНИ РЕАЛИЗАЦИИ

[B1] I. B. Bozhilov, R. R. Petkova, K. T. Tonchev и A. H. Manolova, Програмна реализация: Exploring Semantic-Aware Compression of RGBD Images Using Conventional Codecs, https://github.com/Teleinfrastructure-Research-Lab/rgbd-fusion, Достъпен: 2026-04-07, 2025.

[B2] Bozhilov, Ivaylo and Petkova, Radostina and Tonchev, Krasimir and Manolova, Agata and Poulkov, Vladimir and Vincent Poor, H., AEPCC: Програмна реализация на системата, https://github.com/Teleinfrastructure-Research-Lab/aepcc, Достъпен: 2026-04-07, 2026.

[B3] Bozhilov, Ivaylo and Petkova, Radostina and Tonchev, Krasimir and Manolova, Agata, DPCT: Програмна реализация на системата, https://github.com/Teleinfrastructure-Research-Lab/lb-dpct, Достъпен: 2026-04-21, 2026.

ИЗПОЛЗВАНА ЛИТЕРАТУРА

[4] R. Petkova, I. Bozhilov, A. Manolova, K. Tonchev и V. Poulkov, „On the Way to Holographic-Type Communications: Perspectives and Enabling Technologies," IEEE Access, т. 12, с. 59 236–59 259, 2024. DOI: 10.1109/ACCESS.2024.3393124.

[44] A. Vaswani и др., „Attention is All You Need," Advances in Neural Information Processing Systems, т. 30, 2017.

[55] J. Gao и др., „Finite-Blocklength Information Theory," Fundamental Research, 2026.

[57] C. Bian, Y. Shao и D. Gündüz, „Wireless Point Cloud Transmission," в Proceedings of the 2024 IEEE 25th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC), 2024, с. 851–855. DOI: 10.1109/SPAWC60668.2024.10694621.

[67] R. Petkova, V. Poulkov, A. Manolova и K. Tonchev, „Challenges in Implementing Low-Latency Holographic-Type Communication Systems," Sensors, т. 22, № 24, с. 9617, 2022.

[68] J. Kaplan и др., „Scaling Laws for Neural Language Models," arXiv preprint arXiv:2001.08361, 2020.

[76] Y. Yang, C. Feng, Y. Shen и D. Tian, „FoldingNet: Point Cloud Auto-Encoder via Deep Grid Deformation," в Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, с. 206–215.

[79] I. Bozhilov, K. Tonchev, A. Manolova и R. Petkova, „3D Human Body Models Compression and Decompression Algorithm Based on Graph Convolutional Networks for Holographic Communication," в Proceedings of the 2022 25th International Symposium on Wireless Personal Multimedia Communications (WPMC), 2022, с. 532–537. DOI: 10.1109/WPMC55625.2022.10014791.

[92] M. Gastpar, B. Rimoldi и M. Vetterli, „To Code, or Not to Code: Lossy Source-Channel Communication Revisited," IEEE Transactions on Information Theory, т. 49, № 5, с. 1147–1158, 2003. DOI: 10.1109/TIT.2003.810631.

[95] Y. Shao и D. Gunduz, „Semantic Communications With Discrete-Time Analog Transmission: A PAPR Perspective," IEEE Wireless Communications Letters, т. 12, № 3, с. 510–514, 2023. DOI: 10.1109/LWC.2022.3232946.

[96] M. Liu, W. Chen, J. Xu и B. Ai, „Real-Time Implementation and Evaluation of SDR-Based Deep Joint Source-Channel Coding," в 2022 IEEE 96th Vehicular Technology Conference (VTC2022-Fall), 2022, с. 1–5. DOI: 10.1109/VTC2022-Fall57202.2022.10012971.

[97] G. Fernandes, H. Fontes и R. Campos, Semantic Communications: The New Paradigm Behind Beyond 5G Technologies, arXiv:2406.00754, 2024.

TECHNICAL UNIVERSITY OF SOFIA

FACULTY OF TELECOMMUNICATIONS

DEPARTMENT "RADIOCOMMUNICATIONS AND VIDEOTECHNOLOGIES"

Ivaylo Bozhidarov Bozhilov, MSc

ENCODING AND VISUALIZATION OF 3D OBJECTS USING DEEP LEARNING ARCHITECTURES

ABSTRACT of Ph.D THESIS

This thesis investigates the encoding and visualization of 3D objects using deep learning architectures, with a focus on efficient compression, transmission, and reconstruction of sparse point clouds. It analyzes classical, learning-based, and semantic approaches for 3D content coding and introduces an operational model for systems for acquisition, transmission, and visualization of three-dimensional content.

The work proposes and evaluates autoencoder-based architectures for point cloud geometry coding in both separate source coding and deep joint source-channel coding scenarios. The results demonstrate efficient low-rate compression, robustness to channel noise, and graceful degradation of reconstruction quality. The dissertation also addresses synchronization in dynamic point cloud transmission and proposes methods based on pseudo-random coding and phase-invariant decoding.

TECHNICAL UNIVERSITY OF SOFIA

Faculty of Telecommunications
Department "Radiocommunications and Videotechnologies"

MSc Eng. Ivaylo Bozhidarov Bozhilov

ENCODING AND VISUALIZATION OF 3D OBJECTS USING DEEP LEARNING ARCHITECTURES

A B S T R A C T

of a dissertation for the acquisition of the educational and scientific degree
"DOCTOR OF PHILOSOPHY"

Field: 5. Technical Sciences

Professional field: 5.3. Communication and Computer Engineering

Scientific specialty: Television and Video Technology

Scientific supervisor: Assoc. Prof. Agata Manolova, PhD.

SOFIA, 2026

The dissertation has been reviewed and approved for defense by the Department Council of the Department of "Radiocommunications and Videotechnologies" at the Faculty of Telecommunications of the Technical University of Sofia at a regular meeting held on May 11, 2026.

The public defense of the dissertation will take place on July 20, 2026 at 11:00 in the Conference Hall of the Library Information Center at the Technical University of Sofia, at an open session of the scientific jury appointed by Order No. OJ-5.3-45 / 22.05.2026 of the Rector of the Technical University of Sofia, composed of:

1. Assoc. Prof. Dr. Nicole Christoff – Chair
2. Assoc. Prof. Dr. Adelina Aleksieva-Petrova – Scientific Secretary
3. Prof. Dr. Alexander Bekyarski
4. Prof. Dr. Gabriela Atanasova
5. Assoc. Prof. Dr. Strahil Sokolov

Reviewers:

1. Assoc. Prof. Dr. Nicole Christoff
2. Prof. Dr. Alexander Bekyarski

The materials related to the defense are available to interested parties at the office of the Faculty of Telecommunications at the Technical University of Sofia, Block 1, Room 1254.

The PhD candidate is a doctoral student at the Department of "Radiocommunications and Videotechnologies" of the Faculty of Telecommunications. The research in the dissertation has been conducted by the author, with some parts supported by research projects.

Author: MSc Eng. Ivaylo Bozhilov
Title: Encoding and visualization of 3D objects using deep learning architectures
Print run: 10 copies
Printed at the Publishing House of the Technical University of Sofia

I. GENERAL CHARACTERISTICS OF THE DISSERTATION

Relevance of the Problem

The development of systems for 3D content, extended reality, holographic communication, and interactive multimedia applications has led to a significant increase in the requirements for efficient coding, transmission, and visualization of 3D data. Point clouds and other forms of three-dimensional representation are characterized by extremely large data volumes, which impose serious constraints on the communication and computational resources of modern systems. Classical compression methods, despite their high technological maturity, encounter difficulties in adapting to the complex structure of 3D data and dynamically changing transmission conditions. In this context, learned and semantic-oriented approaches based on deep learning are emerging as a promising direction for improving compression efficiency, robustness to noise, and the adaptability of 3D content transmission systems. This determines the relevance of the present dissertation, which is focused on the investigation and development of architectures and methods for coding and visualization of 3D objects using deep learning.

Objective of the Dissertation, Main Tasks, and Research Methods

The main objective of the present dissertation is to investigate and develop methods for integrating learned and semantic-oriented approaches into systems for capture, transmission, and visualization of 3D content, with the aim of improving the efficiency, adaptability, and functionality of the coding process.

To achieve this objective, the following main tasks are formulated:

1. Analysis of the possibilities for integrating learned and semantic-oriented coding approaches into systems for capture, transmission, and visualization of 3D content.
2. Investigation and development of autoencoder architectures for coding 3D sources.
3. Investigation and development of autoencoder architectures for Joint Source–Channel Coding (JSCC) of 3D content, with the aim of achieving robustness to channel impairments and efficiency under finite block-length conditions.
4. Implementation, experimental evaluation, and comparative analysis of the proposed methods and architectures.

Scientific Contribution

The scientific novelty of the dissertation lies in the development and investigation of learned and semantic-oriented methods for coding and transmission of 3D content, based on autoencoder architectures and Deep Joint Source–Channel Coding (DJSCC). A theoretical framework for entropy coding with mismatched probabilistic models is proposed, enabling the use of more complex models at the encoder through the introduction of side information. Novel architectural solutions for compression and transmission of sparse point clouds are developed, including a phase-invariant decoding method for the transmission of dynamic point clouds, which reduces the dependence on synchronization between the transmitter and receiver. The obtained results extend the application of learned methods in the field of 3D content compression and communication, and demonstrate opportunities for improving the efficiency and robustness of such systems under realistic channel conditions.

Practical Applicability

The practical applicability of the dissertation is reflected in the development of software implementations and methods for efficient coding, transmission, and visualization of 3D content, applicable to systems for extended reality, holographic communication, telepresence, and interactive multimedia environments. The proposed autoencoder architectures and Deep Joint Source–Channel Coding (DJSCC) methods enable reduction of the required transmission bitrate while maintaining good reconstruction quality and robustness to channel noise. Software systems for compression of RGB-D images and point clouds have been implemented, together with experimental implementations of the AEPCC and DPCT architectures, which can serve as a foundation for future scientific research and practical systems for processing and transmission of 3D content. The results of the dissertation may find applications in modern communication systems based on 5G/6G networks, cloud infrastructures, and XR platforms.

Publications

The main achievements and results of the dissertation have been published in 6 scientific publications, of which 3 are published in international scientific journals and the remaining 3 are presented at international scientific conferences. One of the publications is single-authored.

The international scientific journals are: IEEE Access (2024 and 2025) and MDPI Sensors (2023).

The international scientific conferences are: IEEE International Scientific Conference on Information, Communication and Energy Systems and Technologies (ICEST) 2025, IEEE International Symposium on Wireless Personal Multimedia Communications (WPMC) 2025, and Joint International Conference on Digital Arts, Media and Technology with ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunication Engineering (ECTI DAMT & NCON) 2026.

Structure and Scope of the Dissertation

The dissertation consists of **151** pages and includes an introduction, **5** chapters addressing the formulated research objectives, a list of the main contributions, a list of publications related to the dissertation, a list of software implementations, and references. A total of **97** references are cited, of which **94** are in Latin script and the remaining are internet sources. The dissertation contains **49** figures and **8** tables. The numbering of figures and tables in the abstract corresponds to that in the dissertation.

The dissertation is structured into five chapters. Chapter 1 presents an analysis of the state of the art and reviews classical, learning-based, and semantic methods for 3D content coding. Chapter 2 introduces an operational model for systems for acquisition, transmission, and visualization of 3D content and analyzes the integration of learning-based methods within the different system layers. Chapter 3 investigates autoencoder architectures for coding the geometric structure of sparse point clouds. Chapter 4 studies the use of these architectures for compression through quantization and secondary coding of latent representations, while Chapter 5 addresses deep joint source–channel coding for the transmission of point clouds over noisy communication channels.

II. CONTENT OF THE DISSERTATION

CHAPTER 1. ANALYSIS OF THE STATE OF THE PROBLEM BASED ON LITERATURE REVIEW

The first chapter examines the fundamental principles and contemporary approaches to 3D source coding, with emphasis on the development of methods for compression of three-dimensional content. The main categories of methods are presented — classical, learned, and semantic — which differ in the way intermediate data representations are formed and interpreted. Their theoretical foundations, advantages, and limitations are analyzed, together with their relation to the rate–distortion tradeoff and the statistical modeling of 3D data. The objective is to establish a conceptual framework for comparing the different approaches and to outline the development trends in modern systems for compression and transmission of 3D content.

1.3 Fundamentals of 3D Source Coding

1.3.1 Categorization of 3D Source Coding Methods

Based on the presented theoretical formulations, a generalized definition of source coding can be established as an operation that provides a compact intermediate representation of data, enabling reconstruction of the original input signal with controlled distortion while significantly reducing the required channel bandwidth or storage capacity. Figure 1.4 illustrates a generalized compression pipeline, together with the main quality assessment metrics and the processing stages at which they are evaluated.

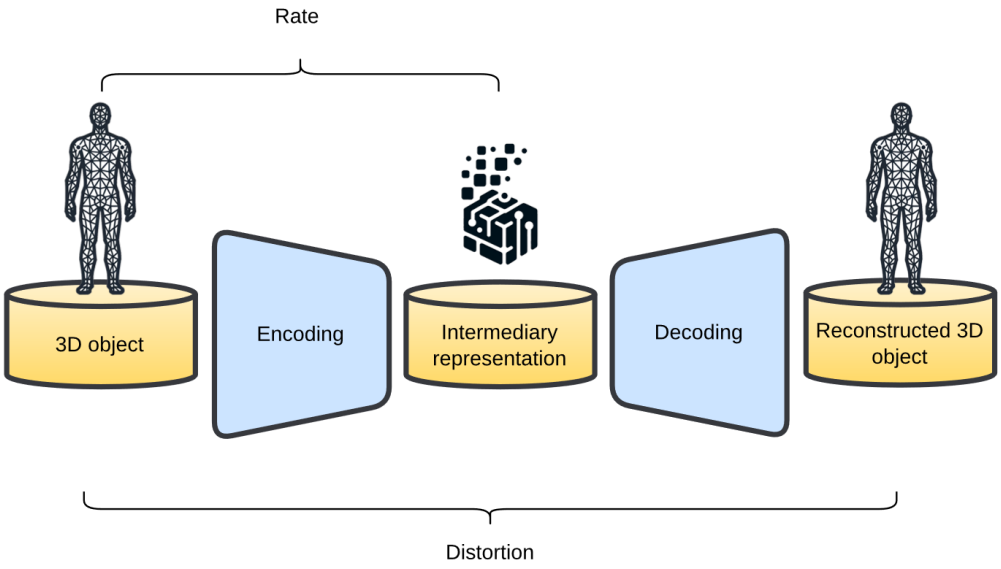


Figure 1.4: General compression pipeline with indicated evaluation metrics.

The analysis in the field of 3D content compression presented in [A1] groups coding methods into three categories. These categories are distinguished according to the form of the intermediate representation and the means by which it is obtained. The following three categories of methods are identified: classical, learned, and semantic, which are discussed in this chapter in Sections 1.4, 1.5, and 1.7, respectively.

1. **Classical compression methods.** In this class of methods, the intermediate representation is typically a vector of features that are not interpretable by humans, or a high-entropy bitstream. Compression is achieved through the classical techniques described in Section 1.1. These approaches are deeply rooted in classical information theory and digital signal processing and form the basis of established standards for 3D content compression.

2. **Learned compression methods.** In learned methods, the intermediate representation is also a vector of features that are not interpretable by humans, but the means used to obtain it are based on machine learning and deep learning techniques. Neural networks, autoencoder architectures, and other trainable models are most commonly employed and optimized according to rate–distortion criteria. Unlike classical approaches, both the transformation and probabilistic modeling are learned directly from data, enabling better adaptation to the complex structure of 3D sources.

3. **Semantic compression methods.** Semantic compression differs in that the intermediate representation consists of human-interpretable features carrying semantic information about the scene or objects. This approach aims to preserve the context and meaning of the content. Semantic methods employ techniques from deep learning, machine learning, and semantic feature extraction, enabling direct interpretation or subsequent processing of the compressed representation without requiring full reconstruction of the original signal.

The taxonomy presented in the systematic survey [A1] reflects the distribution and the main directions of contemporary 3D content compression methods. Figure 1.5 presents a graphical illustration of this taxonomy, which will serve as a conceptual framework for the analysis and comparison of the different classes of methods in the following sections of the dissertation.

1.8 Conclusions

The performed analysis of the state of the problem shows that the coding of visual content, and in particular 3D content, is based on a well-established theoretical framework derived from classical information theory and transform coding, in which the rate–distortion tradeoff and the efficient modeling of statistical dependencies in the data play a central role. Classical methods, represented by standards such as Video-based Point Cloud Compression (V-PCC) and Geometry-based Point Cloud Compression (G-PCC), demonstrate high maturity and efficiency, especially for point cloud content, as confirmed both by their widespread practical use and by their dominant presence in the literature [A1].

On the other hand, learned methods extend this paradigm through the use of neural networks for automatic extraction of compact representations and probabilistic modeling, enabling better adaptation to the complex structure of 3D data and leading to substantial improvements in coding efficiency [A1]. In the context of Joint Source–Channel Coding (JSCC), these methods enable direct optimization of the representation with respect to the channel characteristics and the target task, resulting in more robust behavior under noisy conditions and finite block lengths compared to classical separate coding approaches [55].

Nevertheless, learned approaches are characterized by a higher degree of uncertainty associated both with training and with generalization across diverse datasets, placing them in a tradeoff between efficiency and reliability relative to classical methods. At the same time, there is an increasing trend toward the use of interpretable intermediate representations, in which the compressed content can be analyzed, modified, or utilized without requiring

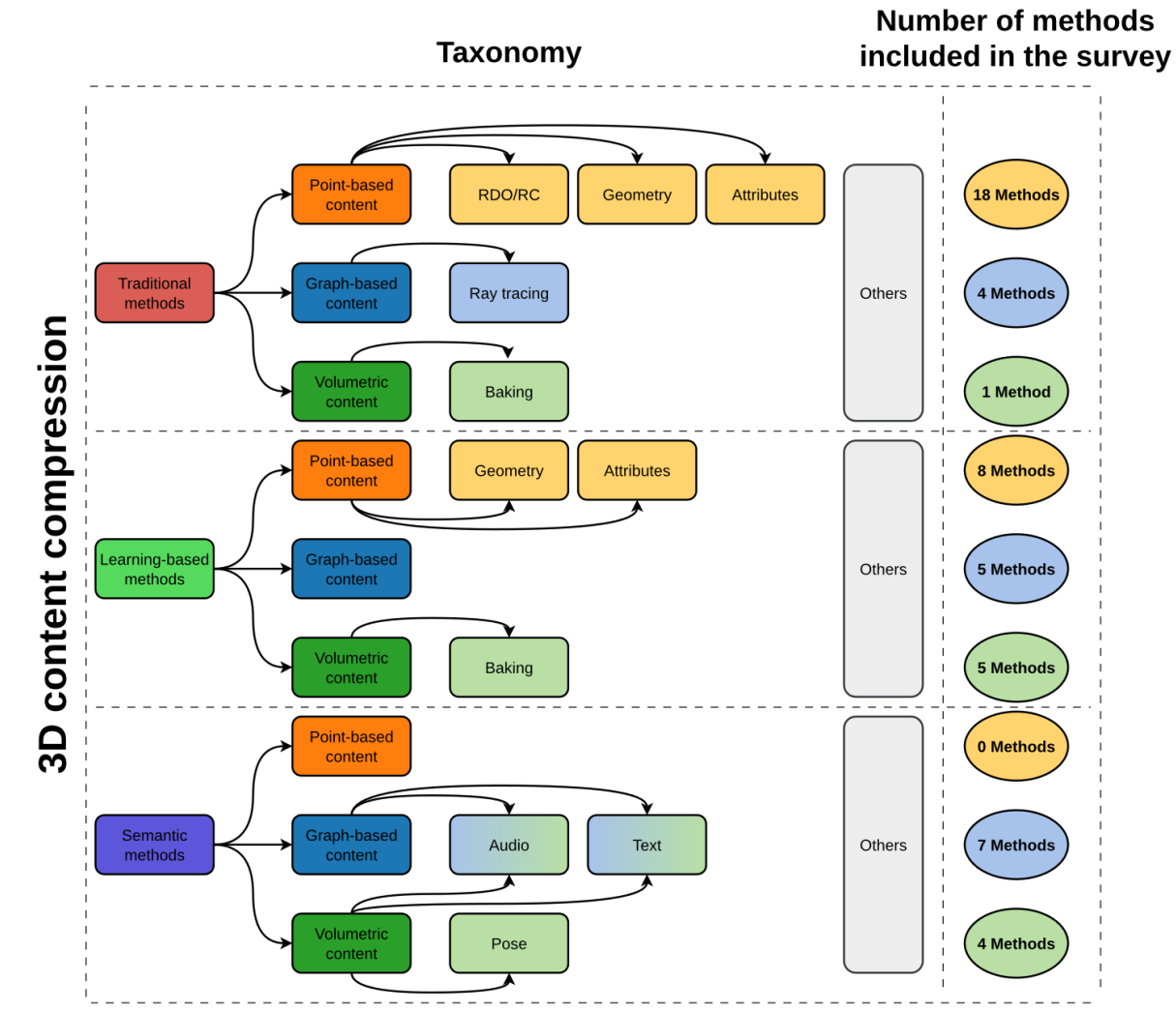


Figure 1.5: Taxonomy of 3D source coding methods presented in [A1].

full reconstruction [A1, A2]. Such representations have practical applications in tasks such as search, editing, scene interaction, and integration with other systems, making them particularly attractive from an engineering perspective.

As summarized in Figure 1.17, adapted from [A1], contemporary approaches can be viewed as an evolution from classical toward learned and interpretable methods, where new opportunities for improving coding efficiency emerge alongside lower technological maturity and a higher degree of uncertainty. This highlights the need for the development of new methods that combine the advantages of classical and learned approaches with the use of interpretable intermediate representations, aiming to achieve higher efficiency, flexibility, and adaptability in the coding and visualization of 3D content.

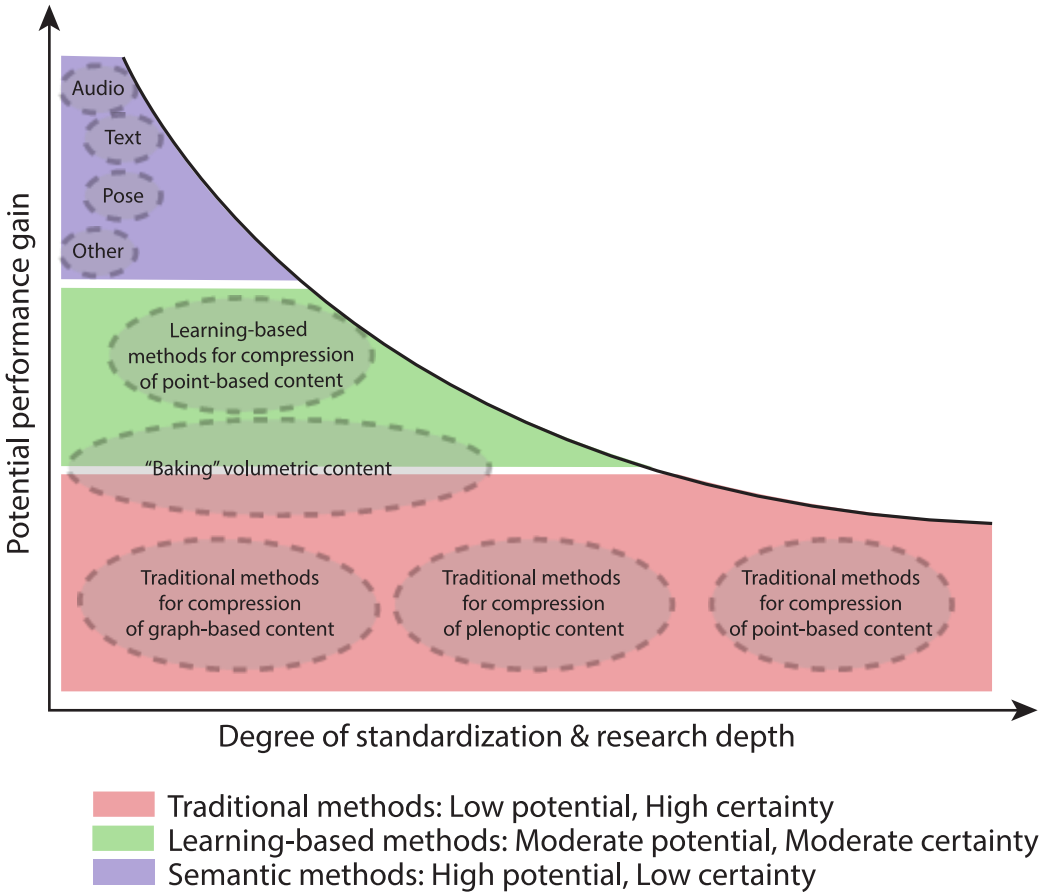


Figure 1.17: Summary of the directions in 3D content compression, adapted from [A1].

1.9 Definition of the Objective and Main Tasks of the Dissertation

The main objective of this dissertation is to investigate and develop methods for integrating learned and semantically oriented approaches into systems for acquisition, transmission, and visualization of 3D content, with the aim of improving the efficiency, adaptability, and functionality of the coding process.

To achieve this objective, the following main tasks are formulated:

1. Analysis of the possibilities for integrating learned and semantically oriented coding approaches into systems for acquisition, transmission, and visualization of 3D content;

2. Investigation and development of autoencoder architectures for coding 3D sources;

3. Investigation and development of autoencoder architectures for Joint Source–Channel Coding (JSCC) of 3D content, aiming to achieve robustness to channel impairments and efficiency under finite block lengths;

4. Implementation, experimental evaluation, and comparative analysis of the proposed methods and architectures.

CHAPTER 2. INTEGRATION OF LEARNED CODING METHODS INTO 3D CONTENT SYSTEMS

The software implementation of the coding approaches in the acquisition layer can be found in [B1]: https://github.com/Teleinfrastructure-Research-Lab/rgbd-fusion

Chapter 2 introduces a unified four-layer operational model for systems for acquisition, transmission, and visualization of 3D content, which is used to analyze the distribution of computational and communication resources across the different parts of the system. Based on this model, practical coding approaches in the acquisition layer are examined, including RGB-D data compression through coloring, multiplexing, and semantic-aware processing, enabling the reduction of communication load while maintaining low computational complexity. In addition, a theoretical framework for coding in the visualization layer is formulated, in which the encoder employs a more accurate and more complex probabilistic model than the decoder, while the mismatch between the two models is compensated through side information. This establishes the foundation for investigating the relationship between model accuracy, complexity, and the required transmission rate.

2.1 Operational Model of Systems for Acquisition, Transmission, and Visualization of 3D Content

In order to analyze the integration of learned coding methods into real-world systems, it is necessary to introduce a general operational model for systems for acquisition, transmission, and visualization of 3D content. In [A3], a four-layer model was proposed to describe the data flow in Holographic-Type Communication (HTC) systems. The model can also be considered in a broader context, since it describes the fundamental operations performed on 3D data: acquisition, processing, transmission, and visualization. In this dissertation, the model is used as a universal operational framework, while HTC is treated as a particular case.

A significant advantage of the model proposed in [A3] is that it is data-centric, i.e., focused on the transformation and transmission of data, which enables a clear distinction between computational and communication "load." The tradeoff between these two resources is fundamental to the system architecture and to the integration of coding methods.

The model from [A3] has been adapted to the general case of systems for acquisition, transmission, and visualization of 3D content and is illustrated in Figure 2.1. It includes the following layers:

1. Acquisition – Transforms the physical scene into digital data through sensors and basic preprocessing.

2. Processing – Converts the raw data into a structured 3D representation through operations such as reconstruction and segmentation.

3. Transmission – Provides efficient and reliable transfer of data between system nodes.

4. Visualization – Reconstructs and visualizes the scene through end devices such as Head-Mounted Displays (HMDs) for virtual or augmented reality.

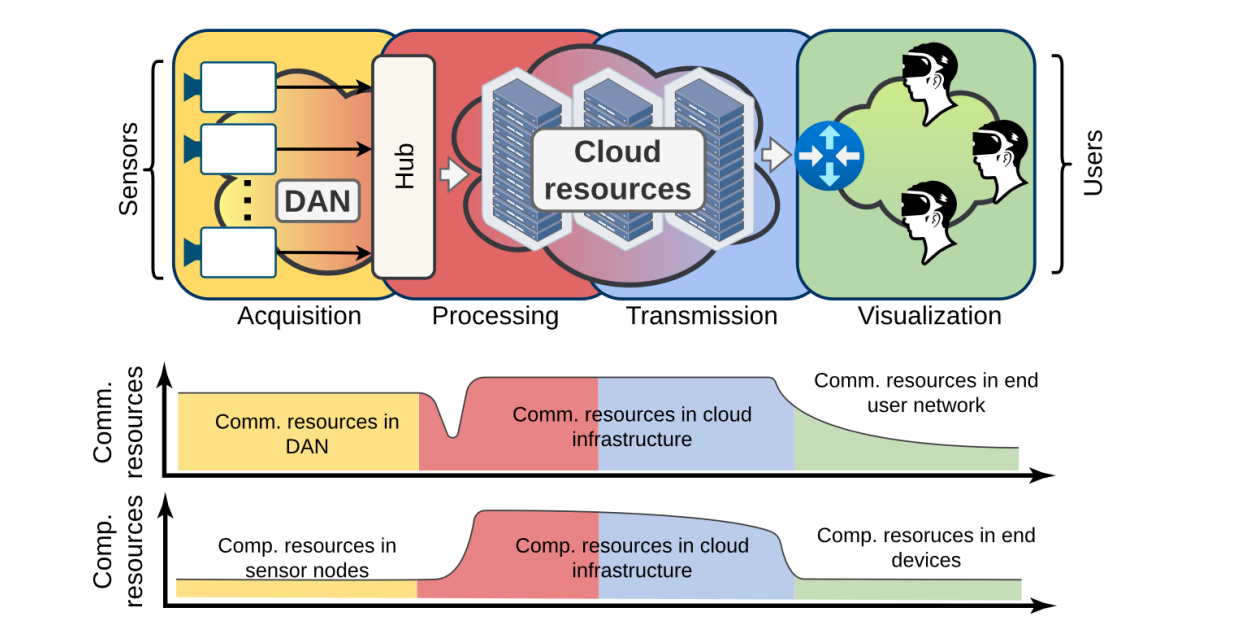


Figure 2.1: Operational model of systems for acquisition, transmission, and visualization of 3D content.

2.1.2 Distribution of Resources Across the Layers

One of the main conclusions in [A3] is that, in 3D content systems, computational and communication resources jointly determine the overall system efficiency. This follows from the fact that data are not transmitted directly, but instead pass through a sequence of transformation operations (e.g., reconstruction, segmentation, coding), which require significant computational resources. Consequently, efficiency cannot be characterized solely by the amount of transmitted data, but rather by the distribution of resources between computation and communication. Formally, system design involves selecting algorithms that minimize communication load under given computational complexity constraints, or vice versa [4, 67]. Within the scope of this dissertation, the focus is placed on the communication aspect of this tradeoff. More specifically, coding methods are considered that reduce the required transmission rate for a given computational resource budget, without analyzing in detail the remaining processing stages (e.g., reconstruction, segmentation, calibration, etc.) characteristic of 3D content or HTC systems.

The distribution of computational and communication resources is not uniform across the system layers. Each layer is characterized by a specific resource profile, which determines the admissible algorithmic complexity and the effective coding strategies (Figure 2.1 and Table 2.2).

In summary, the distribution of resources across the layers is heterogeneous: computational capacity increases from the acquisition layer toward the processing layer, whereas communication constraints are most pronounced during data transfer between the different entities participating in the communication process through a public network (local acquisition environment → cloud computing clusters → end-user devices). This necessitates joint optimization of computational and communication resources in the design and integration of coding methods.

Table 2.2: Summary of resource constraints across system layers.

Layer	Computational resources	Communication resources
Data Acquisition	Low (end devices)	High (LAN)
Processing	High (cloud resources)	High (cloud infrastructure)
Transmission	Moderate to High	High (cloud infrastructure)
Visualization	Low (end devices, HMD)	Low to Moderate (wireless connections)

2.3 Challenges in the Transmission of 3D Content

Systems for acquisition, transmission, and visualization of 3D content are characterized by extremely high requirements in terms of data volume and transmission latency, which creates specific challenges across the different layers of the operational model [4]. During acquisition, a major issue is traffic congestion within the local network [A3]. Even with a relatively small number of sensors, the generated data stream can reach the order of 10^9 bits per second. For example, with three sensors operating at 30 frames per second, the data volume exceeds 2 Gbps [A3]. In this context, the use of reliable transport protocols such as TCP introduces additional effects, including packet retransmissions and congestion control, which may increase latency and lead to variations in frame arrival rates. This creates a mismatch between the generated and the actually processable data stream, requiring careful management of buffering and synchronization.

At the visualization stage, an opposite but equally critical problem arises. End-user devices (e.g., HMDs) have limited communication connectivity and often rely on wireless links with constrained bandwidth. At the same time, the requirements for rendering quality and latency are extremely strict, with acceptable latency on the order of tens of milliseconds [4]. This creates a fundamental conflict between the need to transmit large volumes of 3D data, the limitations of the communication channel, and the limited computational capabilities of end-user devices.

In summary, 3D content systems are constrained both by congestion in local networks during the acquisition stage and by limited bandwidth in end-user networks during visualization. This necessitates the use of efficient compression and adaptive transmission methods that minimize communication load while maintaining low-latency requirements.

2.1.4 Placement and Integration of Learnable Coding Methods

As established in Chapter 1, learnable coding methods achieve higher efficiency compared to classical approaches by adapting to the statistical structure of the data. This is particularly important for 3D content, where dependencies are complex and difficult to describe using analytical models. However, this efficiency is typically achieved at the cost of increased computational complexity, which generally grows with the scale of the model [68]. Within the operational model proposed in [A3], the integration of such methods should therefore be viewed as a problem of joint optimization between computational and communication resources. More specifically, coding methods can be regarded as mechanisms for reducing communication load through the use of additional computational resources. Their practical applicability depends on whether the required complexity can be supported within the corresponding layer of the system.

Consequently, the selection and placement of learnable codecs are not universal, but depend on the resource constraints of the specific layer (capture devices, cloud infrastructure, end-user visualization devices). In this sense, the integration of learnable methods is layer-specific and requires consideration of both the available computational capacity and the bandwidth limitations.

2.4 Conclusions

In this chapter, a unified operational model for systems for acquisition, transmission, and visualization of 3D content was introduced, based on [A3], enabling a systematic analysis of the trade-off between computational and communication resources. It was shown that this trade-off is not uniformly distributed across the system layers, with the visualization layer being characterized simultaneously by limited bandwidth and limited computational capacity. This makes it a critical point for the integration of efficient coding methods.

In the context of the acquisition layer, the compression of Red-Green-Blue-Depth (RGB-D) data through colorization and the use of classical image coding schemes [A4] was analyzed. It was shown that through appropriate preprocessing and multiplexing, a substantial reduction of communication load can be achieved with minimal computational overhead. This demonstrates a practical strategy for adapting coding methods to the constraints of a specific system layer.

Finally, a theoretical problem related to source coding in the visualization layer was formulated, associated with the use of mismatched probabilistic models. It was shown that classical entropy coding constrains model complexity by the resources available at the decoder, which contradicts the desired system architecture. In this context, a framework was defined in which the encoder uses a more accurate model while the mismatch is compensated through side information, and the fundamental condition for the efficiency of such a scheme was derived. This establishes the basis for further investigation of the relationship between model accuracy, complexity, and required transmission rate.

CHAPTER 3. AUTOENCODER ARCHITECTURES FOR SPARSE POINT CLOUD GEOMETRY CODING

The software implementations of the architectures and methods presented in Chapter 3 can be found in [B2]: https://github.com/Teleinfrastructure-Research-Lab/aepcc

This chapter examines the system for coding the geometric structure of sparse point clouds proposed in [A5] (see Fig. 3.1), which is implemented within the conceptual framework introduced in Section 1.7. In particular, the system employs a virtual channel for the transmission of global context, as formulated in Section 1.15 and applied to semantic scenarios in Section 1.7.3. Unlike classical point cloud coding approaches, such as those used in G-PCC, which rely on spatial partitioning through blocks and slices (see Section 1.4.2), the considered system applies semantic segmentation to the scene. This enables the use of an interpretable intermediate representation of the global context, describing the spatial arrangement of objects in three-dimensional space. In this way, the scene is modeled as a collection of semantically meaningful objects rather than as a uniformly discretized space.

Following segmentation, each object is normalized through spherical normalization, while the normalization parameters are stored as metadata (object class, position, and scale). To describe the detailed context characterizing appearance and geometry, the normalized objects are processed by an autoencoder that generates a non-interpretable intermediate representation in the form of a latent vector. Together with the metadata, these vectors form the overall intermediate representation, structured as a set of object records consisting of a latent vector, object position, and object scale within the scene. In [A5] and [A6], several autoencoder architectures operating on the geometric structure of sparse point clouds are proposed. These architectures are analyzed and compared throughout this and the following chapters.

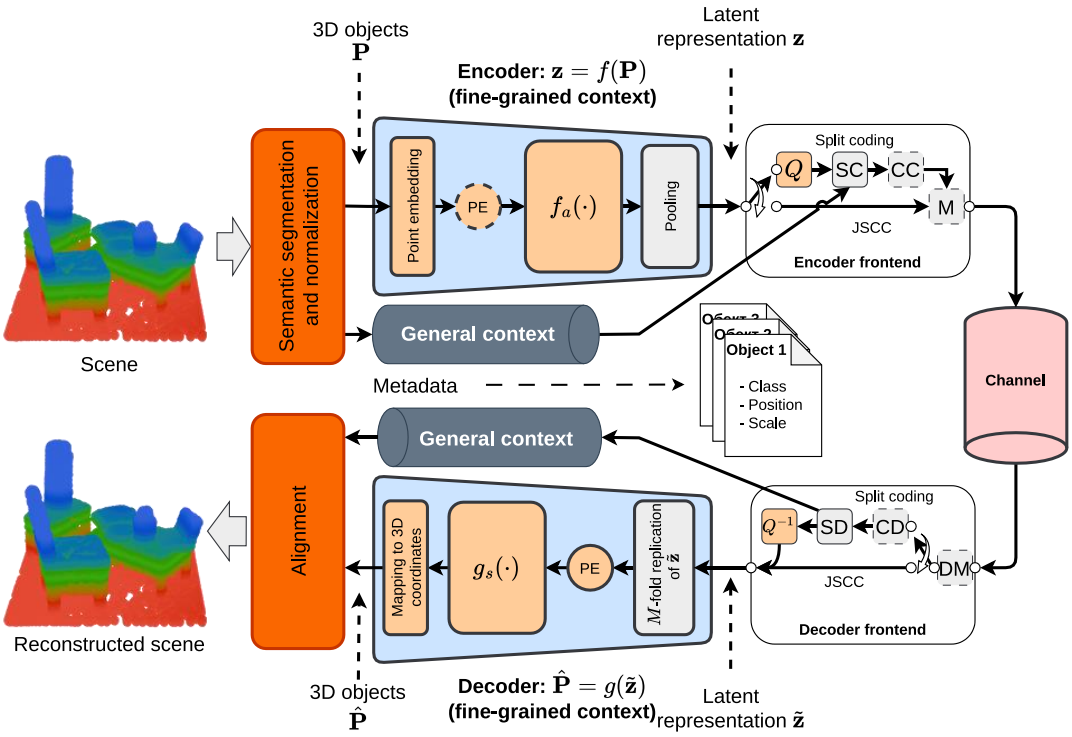


Figure 3.1: Block diagram of the system proposed in [A5].

Q – quantization; SC – secondary coder; CC – channel coder; M – modulator; DM – demodulator; CD – channel decoder; SD – secondary decoder.

For a normalized point cloud $\boldsymbol{P} = \{\boldsymbol{p}_n\}_{n=1}^{N}, \boldsymbol{p}_n \in \mathbb{R}^3$, representing the geometric structure of an individual object, the autoencoder implements an analysis (forward) transform $\boldsymbol{z} = f(P)$ and a synthesis (inverse) transform $\widehat{\boldsymbol{P}} = g(\boldsymbol{z})$ All considered architectures follow a common non-hierarchical (flat) processing principle, similar to that in [76]. In the first stage, the point coordinates are used to extract initial point-wise features. Depending on the specific implementation, positional encoding may be applied to these features. Next, dependencies between points are modeled through a function f_a, which enriches the representation with inter-point context extracted based on the connectivity and topology of the point cloud. This connectivity may be defined explicitly (e.g., through a kNN graph) or implicitly extracted using attention mechanisms [44], which adaptively model dependencies between points according to their features. Finally, a global aggregation operation (e.g., max pooling) is applied to obtain a single global latent vector $\boldsymbol{z}$, representing the entire object. In the decoder, this global vector is replicated (M) times, and positional encoding is applied to the resulting (M) points with identical features. A function g_s, is then used to transform the positionally encoded features, after which they are mapped back into three-dimensional coordinates. The reconstructed objects are finally scaled and positioned within the scene using the global context.

In [A5], two main operating modes for the use of the latent representation are considered. The first is the classical separate coding approach, in which the latent vector $\mathbf{z}$ is quantized, followed by secondary coding (e.g., entropy or dictionary coding). This mode is examined in Chapter 4. Within this scenario, channel coding and modulation for transmission over a communication channel may also be incorporated. The second mode is Joint Source-Channel Coding (JSCC), in which the latent vector $\mathbf{z}$ is used directly for carrier signal modulation and transmitted without quantization. This second mode is examined in Chapter 5.

In summary, the present chapter focuses on the analysis and comparison of different autoencoder architectures, their training strategies, and their computational complexity, with the goal of identifying suitable classes of functions and architectural solutions for efficient extraction of meaningful features from point clouds. These investigations are conducted within a unified geometry coding system that combines an interpretable representation of scene structure with compact latent descriptions of individual objects. The system illustrated in Fig. 3.1 will hereafter be referred to as Autoencoder-based Point Cloud Coding (AEPCC), following the terminology used in [A5].

3.6 Training scenarios

In [A5] and [A6], each architecture is trained using three different latent space dimensionalities: 128, 256, and 512. This variation enables the investigation of the influence of latent space capacity on reconstruction quality and behavior in the presence of channel noise. All models presented in [A5] — namely FoldingNet, Graph Autoencoder (GAE), and Transformer Graph Autoencoder (TGAE) — are trained and validated on the synthetic **SYNTH** dataset described in Section 3.5. In contrast, the architectures in [A6] are trained on the dataset introduced in [79], which is divided into training, validation, and test subsets in an 80%:10%:10% ratio. To reduce task uncertainty in [A6], the global rotation of objects is removed so that all models are consistently aligned. The total number of encoder and decoder parameters for each architecture considered in this chapter is summarized in Table 3.1. In both [A5] and [A6], the Chamfer distance is used as the loss function, computed between the original and reconstructed point clouds $\boldsymbol{P}$ and $\widehat{\boldsymbol{P}}$, as defined in (3.11), where $N = |\boldsymbol{P}|$ and $M = \left|\widehat{\boldsymbol{P}}\right|$.

Table 3.1: Number of encoder and decoder parameters for the different architectures and their training times.

Architecture	**Module**	**F = 128**	**F = 256**	**F = 512**	**Training time**
FoldingNet	Encoder	100,8K	282K	939,5K	6,11h./6,91h./8,28h.
	Decoder	68,6K	268,3K	1M	
GAE	Encoder	767,8K	2M	5,9M	21,33h./28,46h./60,06h.
	Decoder	939,8K	2,6M	7,9M	
TGAE	Encoder	1,1M	4,2M	16,5M	17,13h./30,38h./111,18h.
	Decoder	1,5M	5,8M	23,1M	
SEPT	Encoder	2,3M	2,3M	2,3M	4,21h./5,67h./6,50h.
	Decoder	17,8M	17,8M	17,8M	
DPCT	Encoder	5,4M	5,4M	5,6M	6,17h./6,14h./6,20h.
	Decoder	74,5M	74,6M	74,7M	

$$L_{CD}\left(\boldsymbol{P}, \widehat{\boldsymbol{P}}\right) = \frac{1}{N}\sum_{\boldsymbol{p}\in\boldsymbol{P}} \min_{\widehat{\boldsymbol{p}}\in\widehat{\boldsymbol{P}}} |\boldsymbol{p} - \widehat{\boldsymbol{p}}|^2 + \frac{1}{M}\sum_{\widehat{\boldsymbol{p}}\in\widehat{\boldsymbol{P}}} \min_{\boldsymbol{p}\in\boldsymbol{P}} |\widehat{\boldsymbol{p}} - \boldsymbol{p}|^2 \tag{3.11}$$

In [A5], all models are trained using the Adaptive Moment Estimation (ADAM) optimizer for 300 epochs. The learning rate is set to 10^{-4}, with a weight decay coefficient of 10^{-6}. A learning rate scheduler is employed, applying step-wise decay every 60 epochs with a factor of 0.5. The batch size is 64. Since some point clouds are padded with zeros, a point-level binary mask (described in Section 3.5) is used to exclude the padded points from the loss computation. For the GAE architecture, masking is applied both at the input and output, since the model preserves correspondence between input and output points, as described in Section 3.3.

In [A6], training is performed in a similar but modified manner. For the Dynamic Point Cloud Transmission (DPCT) architecture, the same batch size is used, but training is parameterized with different channel noise levels, $SNR_{train} \in \{0dB,5dB,10dB\}$. During the training of the FoldingNet [76] and Semantic Point Cloud Transmission (SEPT) [57] architectures in [A6], an initial learning rate of 10^{-3} is used, while DPCT is trained with a learning rate of 10^{-4}. This reflects the higher complexity of the architecture and the need for more stable optimization. An additional variation of DPCT with phase-invariant decoding, discussed in Section 5.4, is also trained using $SNR_{train.}$ = 5dB and F=512. In [A6], all architectures are trained for 200 epochs.

The Chamfer loss curves during training and validation for the architectures proposed in [A5] are presented in Section 3.6 of the dissertation. All models demonstrate stable convergence, while the observed differences in loss behavior correlate with both model capacity and the specific architectural and training characteristics. For the experiments in the following chapters, the weights from the epoch with the minimum validation loss are used.

Considering the three architectures examined in [A5] and the three latent vector dimensionalities used during training, a total of 9 different models are trained in [A5]. In [A6], three architectures (FoldingNet, SEPT [57], and DPCT) are trained, again with three variations of latent space dimensionality and three variations of channel Signal-to-Noise Ratio (SNR), denoted as SNR_{train}. Consequently, a total of 27 models are trained in [A6], in addition to one extra model investigating phase-invariant decoding, discussed in Chapter 5. These models are used in the following chapters for experiments on point cloud compression and Deep Joint Source-Channel Coding (DJSCC).

3.6 Conclusions

In this chapter, autoencoder architectures for coding the geometric structure of sparse point clouds within the AEPCC system were examined. It was shown that the general framework, based on semantic segmentation, interpretable representation of global context, and compact latent descriptions of individual objects, enables the integration of different architectural approaches into a unified system for compression and transmission of 3D content.

The analyzed architectures cover a broad range of point cloud processing methods — from FoldingNet-based decoders with geometric positional encoding, through graph-convolutional autoencoders, to hybrid architectures incorporating self-attention and 3D convolutions. This makes it possible to investigate the influence of different mechanisms for modeling inter-point dependencies, various forms of positional encoding, and different decoding strategies on reconstruction quality, computational complexity, and applicability in point cloud compression and DJSCC tasks.

CHAPTER 4. COMPRESSION OF SPARSE POINT CLOUDS USING AUTOENCODER ARCHITECTURES

The software implementations of the methods presented in Chapter 4 can be found in [B2]: https://github.com/Teleinfrastructure-Research-Lab/aepcc

This chapter examines the use of the autoencoder architectures analyzed in Chapter 3 within the classical separate coding framework. In this scenario, the autoencoder acts as a learnable transform that extracts a compact latent vector **z**, describing the geometric structure of the object. In this sense, the considered formulation corresponds to the classical visual content coding pipeline, in which three main stages are combined: (learnable) transformation, quantization, and entropy or dictionary coding. According to the block diagram in Fig. 3.1, this chapter focuses on the

quantization and secondary coding blocks of the latent representation, without considering channel coding and modulation.

The main question addressed in this chapter is to what extent the latent representations extracted by different architectures are suitable for efficient geometry compression. To this end, the influence of architectural complexity, latent space dimensionality, and quantization step size on the rate–distortion trade-off is analyzed. Within this chapter, the transmission channel is assumed to be ideal, meaning that transmission errors are not considered. Consequently, the analysis is focused entirely on geometry compression (source coding). This makes it possible to clearly separate the effects of quantization of **z** from the effects associated with transmission over a noisy channel, which are addressed in Chapter 5.

More specifically, the chapter investigates the quantization of latent vectors, the formation of a serialized intermediate representation, and the application of secondary coding to the resulting discrete symbols. The obtained results are evaluated through rate–distortion characteristics, enabling comparison both between the different autoencoder architectures and against classical geometry coding approaches.

4.1 Problem Formulation

The considered experimental setup is illustrated in Fig. 4.1. Let $\boldsymbol{P} = \{\boldsymbol{p}_n\}_{n=1}^{N}$, $\boldsymbol{p}_n \in R^{\mathbb{3}}$, denote a normalized point cloud representing the geometric structure of an individual object. As discussed in Chapter 3, the encoder implements a learnable transform $\boldsymbol{z} = f(\boldsymbol{P})$, where $\boldsymbol{z} \in R^F$ is the latent vector, and F is the dimensionality of the latent space. In the general case, **z** contains continuous values and cannot be directly represented as a finite bitstream. Therefore, in the separate coding framework, a quantization stage is introduced, through which a discrete latent representation $\hat{\boldsymbol{z}} = Q(\boldsymbol{z})$, where $Q(\cdot)$ denotes the quantization operator. The resulting discrete representation $\hat{\boldsymbol{z}}$ can then be serialized and represented by a compact bitstream. In this sense, the autoencoder defines the transformation $\boldsymbol{P} \mapsto \boldsymbol{z}$, while the subsequent stages provide a discrete representation of **z,** suitable for efficient secondary coding. The decoder reconstructs the geometric structure through $\hat{\boldsymbol{P}} = g\left(Q^{-1}(\hat{\boldsymbol{z}})\right)$, where $g(\cdot)$ is the synthesis transform.

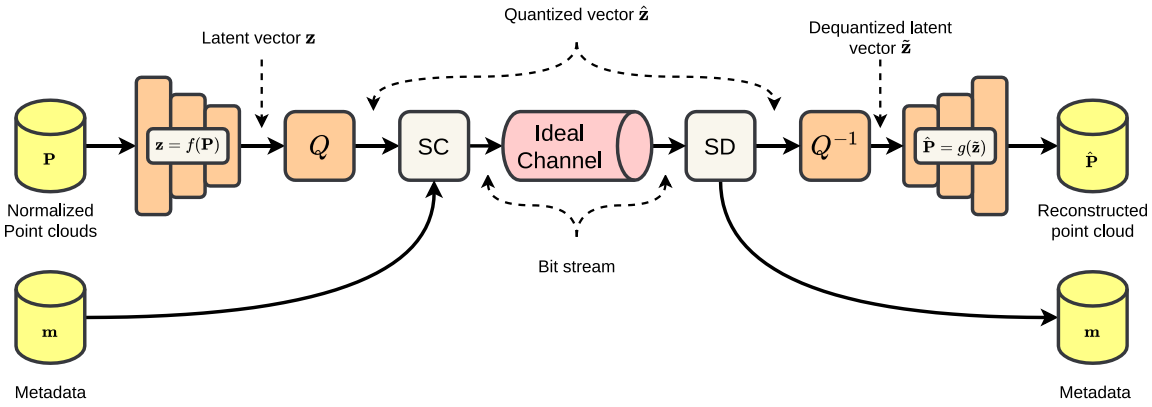


Figure 4.1: Diagram of the considered experimental setup.

In the considered scenario, the bitrate depends on three main factors: the latent vector dimensionality F, the quantization step size Δ, and the efficiency of the secondary coding scheme. Increasing F generally increases the representational capacity and may improve reconstruction quality, but simultaneously increases the number of coded symbols. Reducing Δ leads to a more precise representation of the latent vectors, but typically increases the entropy of the symbol stream

and consequently the required bitrate. In turn, the secondary coding stage removes statistical redundancy from $\hat{\boldsymbol{z}}$ in order to reduce the length of the bitstream without introducing additional information loss.

An important characteristic of the considered setup is the assumption of an ideal transmission channel, i.e., after secondary decoding, the exact same quantized latent vector $\hat{\boldsymbol{z}}$ generated by the encoder is perfectly reconstructed. Consequently, the only sources of distortion considered in this chapter are the imperfections of the autoencoder architecture and the quantization of the latent representation, rather than transmission errors. This makes it possible to experimentally analyze the extent to which different autoencoder architectures produce latent representations that are both compact and robust to quantization.

Within the meaning of the system shown in Fig. 3.1, each object is represented by the pair $(\hat{\boldsymbol{z}}, \boldsymbol{m})$, where **m** denotes the associated metadata related to the object's position, scale, and class. In this chapter, the primary focus is on the compression of the latent vector $\hat{\boldsymbol{z}}$, since it carries the main information describing the detailed geometric structure. The metadata are treated as an additional serialized description required for arranging the objects within the scene, but they do not alter the fundamental relationships between latent space dimensionality, quantization, and reconstruction quality. Based on this framework, the following sections analyze specific quantization and secondary coding strategies, as well as the resulting rate–distortion characteristics of the architectures introduced in Chapter 3.

4.6 Results for Compression of the Geometric Structure of 3D Scenes

The scene compression results reported in [A5] were obtained under two experimental scenarios. In the first scenario, the rate–distortion (R–D) characteristics were analyzed using scenes from the REAL dataset. The observed behavior is similar to that obtained for individual object compression (Section 4.5), confirming the following main trends: (i) TGAE demonstrates the best R–D efficiency; (ii) model size has a significant impact under aggressive quantization, but its influence decreases at higher bitrates; and (iii) the best performance is not necessarily achieved by the largest models.

In the second experiment, the R–D characteristics were evaluated on four independent scenes — two from the SceneNN dataset and two captured by the authors of [A5]. The quantitative and qualitative results are presented in Section 4.6 of the dissertation. The input scenes and the corresponding reconstructions generated by the different methods are shown together with the associated R–D curves for each scene.

The presented results show that neither Draco nor G-PCC are capable of reaching the very low bitrates achieved by AEPCC, regardless of the autoencoder architecture used. Due to the limited overlap between AEPCC and the classical codecs in the low-bitrate region, the Bjøntegaard Delta PSNR (BD-PSNR) metric cannot be reliably computed. Instead, when sufficient overlap exists in terms of Peak Signal-to-Noise Ratio (PSNR), the Bjøntegaard Delta Rate (BD-Rate) metric is used with G-PCC as the reference method, and the results are summarized in Table 4.3.

4.7 Conclusions

This chapter investigates the use of the autoencoder architectures introduced in Chapter 3 within the classical separate coding framework, where the latent representation is quantized and subjected to secondary coding. It was shown that the autoencoder architectures generate latent representations that can be efficiently compressed, with the rate–distortion tradeoff depending

Table 4.3: BD-Rate comparison of the coding schemes using G-PCC as the reference codec.

Codec	F	Scene 1		Scene 2		Scene 3		Scene 4		Average	
		BD-Rate [%]	Time [s]	BD-Rate [%]	Time [s]	BD-Rate [%]	Time [s]	BD-Rate [%]	Time [s]	BD-Rate [%]	Time [s]
FoldingNet	128	-85.69	0.30 ± 0.006	-83.01	0.25 ± 0.004	-72.69	0.51 ± 0.011	-82.88	0.62 ± 0.012	-81.07	0.42 ± 0.151
FoldingNet	256	-81.53	0.30 ± 0.004	-77.23	0.25 ± 0.004	-68.15	0.51 ± 0.009	-74.94	0.63 ± 0.015	-75.46	0.42 ± 0.156
FoldingNet	512	-69.59	0.30 ± 0.006	-73.90	0.25 ± 0.004	22706	0.52 ± 0.013	-62.25	0.63 ± 0.020	-50.53	0.42 ± 0.158
GAE	128	-87.45	0.45 ± 0.006	-74.98	0.34 ± 0.005	-77.59	0.83 ± 0.011	-84.54	1.05 ± 0.014	-81.14	0.67 ± 0.289
GAE	256	-81.99	0.46 ± 0.005	-66.24	0.35 ± 0.006	-71.02	0.86 ± 0.011	-76.97	1.09 ± 0.013	-74.06	0.69 ± 0.299
GAE	512	26207	0.47 ± 0.007	-52.58	0.36 ± 0.006	-53.62	0.91 ± 0.012	-62.35	1.15 ± 0.016	-39.46	0.72 ± 0.323
TGAE	128	-87.81	0.39 ± 0.006	-86.34	0.32 ± 0.004	-81.09	0.66 ± 0.012	-84.51	0.81 ± 0.015	-84.94	0.55 ± 0.201
TGAE	256	-76.29	0.40 ± 0.006	-82.81	0.33 ± 0.005	-74.20	0.69 ± 0.015	-77.32	0.85 ± 0.019	-77.65	0.57 ± 0.214
TGAE	512	-73.35	0.43 ± 0.009	-73.13	0.34 ± 0.008	-58.79	0.79 ± 0.016	-63.62	0.99 ± 0.020	-67.22	0.64 ± 0.265
CRCIR	—	-34.34	0.04 ± 0.003	12571	0.20 ± 0.006	-28.74	0.06 ± 0.013	-45.59	0.09 ± 0.025	-25.58	0.10 ± 0.066
Draco	—	-33.23	0.54 ± 0.166	-11.83	0.27 ± 0.065	0.12	1.25 ± 0.416	-25.70	1.69 ± 0.597	-17.66	0.94 ± 0.674
G-PCC	—	—	0.71 ± 0.374	—	0.33 ± 0.132	—	2.22 ± 1.371	—	0.97 ± 0.736	—	1.11 ± 1.096

The dark gray cells indicate the best BD-Rate for the corresponding scene, while the light gray cells denote the second-best result. Processing times (the best result is underlined) were measured on a system equipped with an AMD Ryzen 9 7950X CPU, 64 GB RAM, and an NVIDIA RTX 4090 GPU.

jointly on the architecture, the dimensionality of the latent space, and the quantization step size. The results demonstrate that, for compression at both the individual object level and the scene level, the TGAE architecture achieves the best rate–distortion (R–D) efficiency while also demonstrating good generalization capability when transitioning from synthetic to real-world data.

The analysis further shows that model capacity is particularly important under aggressive quantization, whereas at higher bitrates the architectural design has a more significant influence on reconstruction quality. In addition, the scene compression results indicate that the proposed AEPCC approach outperforms classical coding schemes such as Draco and G-PCC, as well as the learned CRCIR method, in the low-bitrate regime for sparse point clouds, especially when using the GAE and TGAE architectures. These findings confirm that learned transformations, combined with appropriate quantization and secondary coding, constitute an effective approach for compressing the geometric structure of sparse point clouds.

CHAPTER 5. DEEP JOINT SOURCE–CHANNEL CODING FOR POINT CLOUDS

The software implementations of the methods presented in Chapter 5 can be found in [B3]: https://github.com/Teleinfrastructure-Research-Lab/lb-dpct

The previous chapter examined the problem of point cloud compression using learned autoencoder architectures within the classical separate coding framework. In this mode, scene compression is achieved through quantization of the latent vector followed by secondary coding, while Chapter 4 considered the transmission of the resulting bitstream over an ideal noiseless channel. In the presence of a real communication channel affected by noise, additional channel coding is required in order to protect the transmitted information against transmission errors. The classical separate coding approach is theoretically optimal for infinite block lengths and stationary channel conditions, according to the theoretical framework discussed in Section 1.6 of the dissertation. In practical systems, however, where finite codeword lengths and dynamically varying channels are considered, this approach leads to significant limitations. The most characteristic of these is the

so-called *cliff-effect*, where a slight degradation in channel SNR results in a sharp drop in reconstruction quality. This behavior is particularly undesirable for real-time transmission of 3D content, where graceful degradation is a critical requirement. This limitation of separate coding manifests precisely in the finite block-length regime, where Joint Source–Channel Coding (JSCC) represents a more suitable alternative [A5, 57, 55]. In this context, the AEPCC system introduced in [A5] and discussed in Chapter 3 supports a JSCC/DJSCC operating mode, in which the latent representations can be directly used for carrier modulation and transmission over a noisy channel without the need for quantization and channel coding.

The advantages of the JSCC approach are particularly significant for dynamic 3D data, which are becoming a key type of content for applications such as Extended Reality (XR), telepresence [A3, 4], and mobile robotics, but whose transmission over wireless channels remains challenging due to their high dimensionality and complexity [4]. Classical separate coding approaches suffer from the *cliff-effect* and limited adaptability in the short block-length regimes characteristic of modern applications [92]. In contrast, Deep Joint Source–Channel Coding (DJSCC), implemented through the direct transmission of continuous latent representations, provides graceful quality degradation and greater adaptability to channel conditions. This chapter analyzes the noise sensitivity of the autoencoder architectures proposed in [A5] and introduced in Chapter 3, namely FoldingNet, GAE, and TGAE, by investigating their behavior in the presence of Additive White Gaussian Noise (AWGN) in the latent space. This analysis serves as a transition from the compression-oriented setup studied in Chapter 4 toward a realistic communication scenario. Subsequently, the experimental setup from [A6] is introduced and formalized, examining the behavior of the DPCT architecture in JSCC mode, where latent vectors are directly used for modulation and transmission over a noisy channel. In addition, the chapter discusses the specific challenges arising in the transmission of dynamic point clouds in the JSCC regime, including synchronization problems and sensitivity to temporal shifts, as well as the solutions proposed in [A6] to address these issues.

5.2 Joint Source–Channel Coding and Transmission over Noisy Channels

In the second operating mode of the AEPCC system shown in Fig. 3.1, namely the JSCC mode, where the components of the latent vector are used for carrier modulation and transmission over a noisy channel, the DPCT architecture proposed in [A6] is investigated. The experimental setup from [A6] is illustrated in Fig. 5.3 and generally corresponds to the second operating mode of AEPCC, extended with additional blocks responsible for specific transmission-related functions in a noisy communication channel.
In this mode, the transmitted symbols correspond directly to the elements of the latent vector generated by the encoder, according to the architecture described in Section 3.4.2 and in previous works [76, 57, 95]. The latent vector $\boldsymbol{z} \in R^F$ consists of F real components whose amplitudes are treated as continuous values, although in practice they are represented using 32-bit floating-point numbers. To transmit z over a physical channel, the Discrete Time Analog Transmission (DTAT) [95]. Let $z = [z_0, \dots, z_{F-1}]^\top d$enote the latent vector generated by the DJSCC encoder, where each element $\boldsymbol{z}_k$ corresponds to a transmitted symbol $a[k]$ (a real-valued amplitude). The symbols are interpreted as pulses spaced at intervals of T and shaped using a Root-Raised-Cosine (RRC) filter. The resulting DTAT signal is mathematically described in Eq. 5.6, where $g_{\mathrm{RRC}}(t)$ denotes the impulse response of the RRC filter, $k = 0, \dots, F - 1$, and T is the symbol interval. This pulse shaping ensures that the transmitted signal remains bandwidth-limited, while the use of a matched RRC filter at the receiver together with sampling at the symbol instants minimizes Inter-Symbol Interference (ISI).

$$s(t) = \sum_k a[k]\ g_{\mathrm{RRC}}(t - kT) \tag{5.6}$$

Within this framework, two main technical challenges arise, which are addressed in the following sections:

1. **Efficient decoding in a noisy channel:** The transmitted point cloud must be reconstructed from the noisy latent vector with minimal quality degradation. As shown in Section 5.1 and in [A5], autoencoders demonstrate significant robustness to noise in the latent space, even when such noise is absent during training. However, [57] shows that injecting Additive White Gaussian Noise (AWGN) into the latent vector during training improves robustness to channel noise. Furthermore, the use of a matched RRC filter at the receiver maximizes the output Signal-to-Noise Ratio (SNR) and is optimal for AWGN channels, making it a natural choice for recovering the latent symbols [95].

2. **Frame synchronization:** During the transmission of dynamic point clouds, multiple latent vectors are transmitted sequentially, each corresponding to an individual frame. The receiver must determine the boundaries between these consecutive vectors in order to correctly reconstruct the separate frames. Unlike digital communication systems, where this task is typically solved through packet headers or predefined preambles [96], such structures are not available in analog DJSCC communication systems. This makes the frame synchronization problem significantly more challenging, and it remains relatively unexplored in the literature [97]. A visual illustration of this problem is presented in Fig. 5.4A, showing how receiver desynchronization leads to incorrect reconstruction of the latent vector.

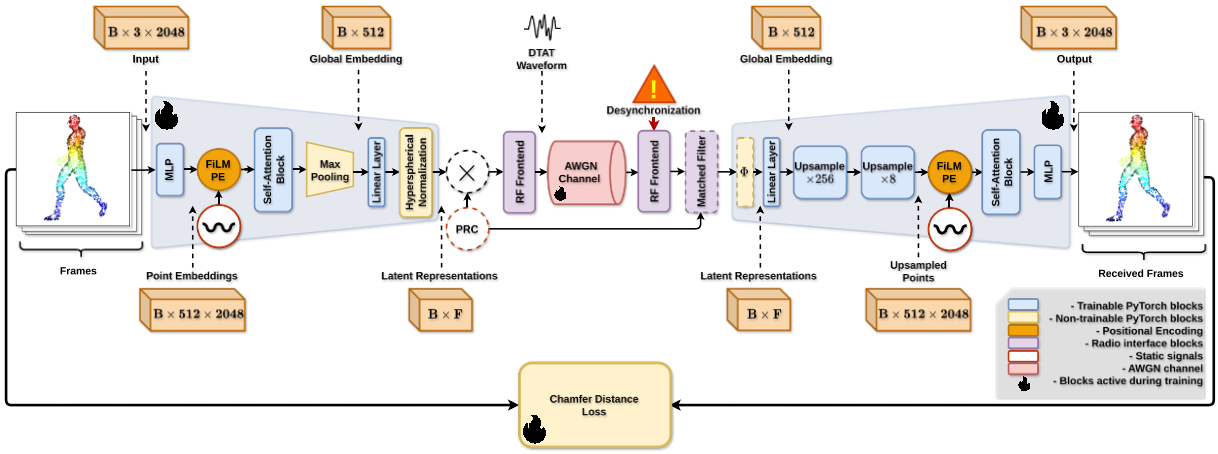


Figure 5.3: Block diagram of the experimental setup for DPCT [A6].

5.3 Synchronization Using PRC and Matched Filtering

The proposed synchronization scheme employs a Pseudorandom Code (PRC) together with a matched filter in order to reliably determine the frame start during transmission over an AWGN channel. The transmitted sequence is multiplied element-wise by the PRC sequence, after which the signal undergoes pulse shaping using an RRC filter and is transmitted through the channel. At the receiver, the received signal is filtered, sampled, and correlated with the conjugated PRC sequence in the frequency domain. The position of the maximum of the correlation function determines the estimated frame timing offset, allowing proper alignment and recovery of the synchronized sequence. The mechanism through which the matched filter and the correlation with the PRC sequence enable synchronization recovery is illustrated in Fig. 5.4B.

5.4 Phase-Invariant Decoding

The proposed phase-invariant decoding scheme (see Fig. 5.4C) removes the dependence on temporal alignment by using the amplitude spectrum of the Discrete Fourier Transform (DFT), which is invariant to cyclic shifts. The received latent sequences are transformed using a unitary DFT, while the decoder operates solely on the amplitudes of the spectral coefficients. The analysis shows that even when the extracted block overlaps the boundary between two consecutive frames, the resulting representation remains close to that of the target frame under the assumption of small inter-frame variations. This enables reliable reconstruction of the point cloud without the need for explicit synchronization. The phase-invariant layer is inserted before the decoder and provides robustness against desynchronization between the transmitter and receiver.

5.5 Results for DJSCC and Transmission over Noisy Channels

This section presents the experimental results from [A6], evaluating reconstruction quality under different channel conditions within the experimental setup described in Section 5.2. Figure 5.5 illustrates the relationship between the latent vector size F, the channel Signal-to-Noise Ratio during testing — $\mathrm{SNR}_{\mathrm{test}}$ — and the reconstruction PSNR. For this experiment, the training $\mathrm{SNR}_{\mathrm{train}}$ is selected to match the testing condition as closely as possible, i.e., $\mathrm{SNR}_{\mathrm{train}} = \arg\min_{\mathrm{SNR}_{\mathrm{train}}} |\mathrm{SNR}_{\mathrm{train}} - \mathrm{SNR}_{\mathrm{test}}|$, where $\mathrm{SNR}_{\mathrm{train}} \in \{0\,\mathrm{dB}, 5\,\mathrm{dB}, 10\,\mathrm{dB}\}$. As described in Section 5.2, the latent vector is treated as a sequence of real-valued symbols forming a DTAT signal for transmission. Consequently, the number of transmitted symbols per point cloud frame — and therefore the required bandwidth — increases with increasing F.

The results show that reconstruction quality improves both with increased bandwidth (larger F) and under better channel conditions (higher $\mathrm{SNR}_{\mathrm{test}}$). Among the architectures evaluated in [A6], the proposed method outperforms both SEPT and FoldingNet across the entire range of F and $\mathrm{SNR}_{\mathrm{test}}$ values. Although FoldingNet was not originally designed as a DJSCC method, it is included for completeness, since it demonstrates relatively good performance with respect to its number of parameters. Nevertheless, it is outperformed by SEPT and DPCT in most configurations.

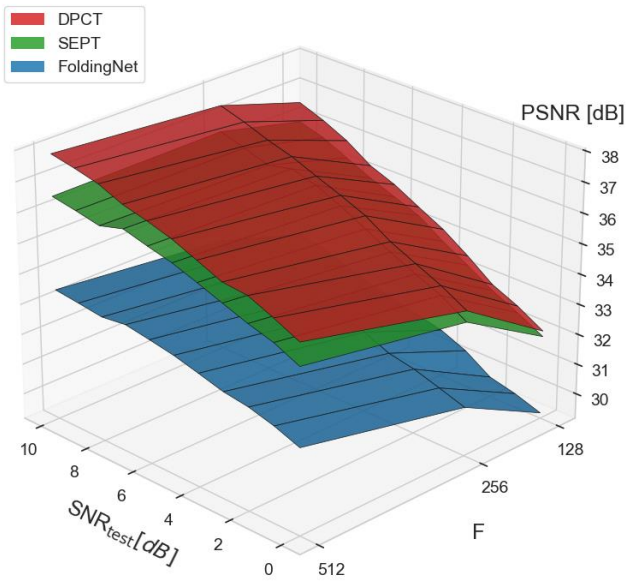
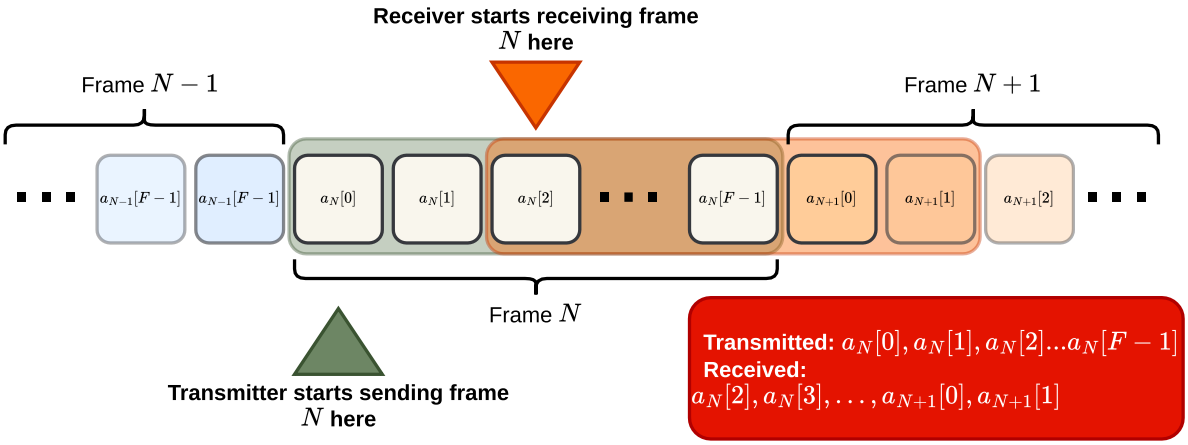
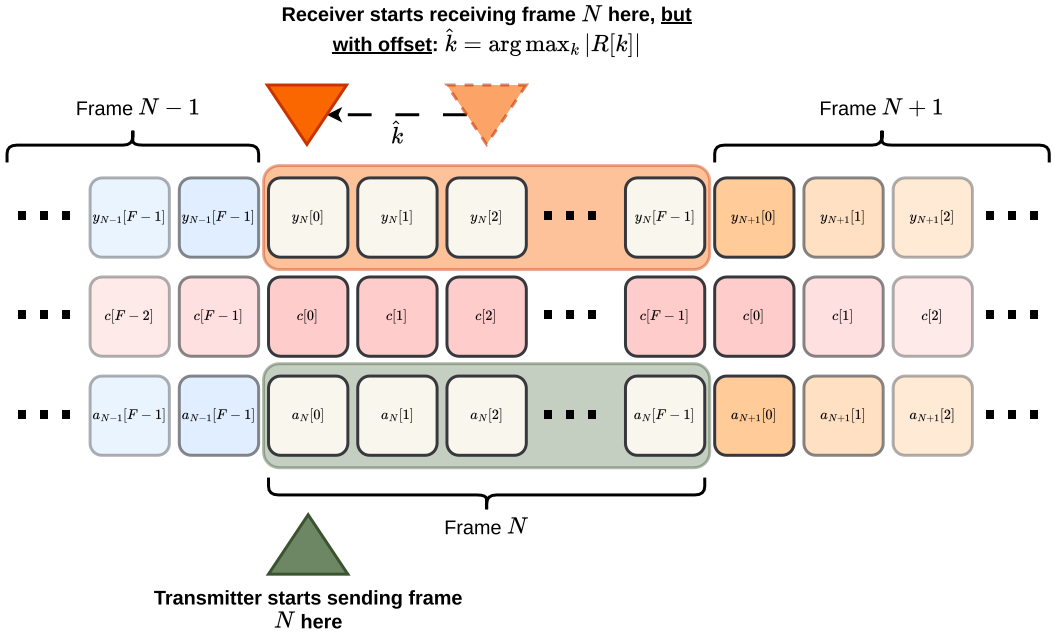


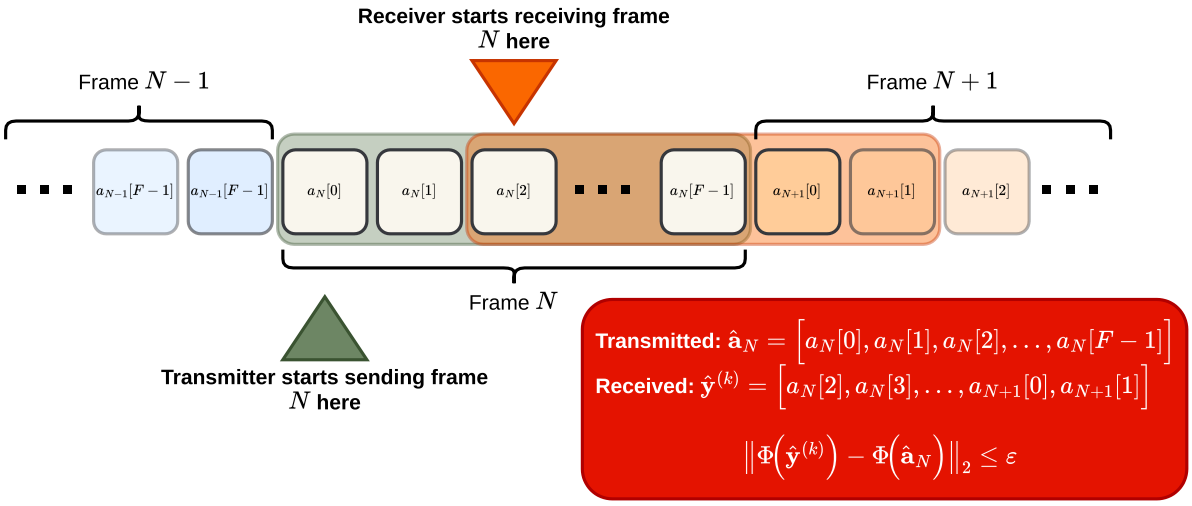
Figure 5.5: Dependence of PSNR on bandwidth (F) and $\mathrm{SNR}_{\mathrm{test}}$.



(A) The desynchronization problem when latent vectors are used for modulation and transmission over a noisy channel.



(B) Synchronization recovery using a matched filter.



(C) Phase-Invariant Decoding (PID), where explicit synchronization recovery is not required.

Figure 5.4: The desynchronization problem and the proposed solutions in [A6].

5.7 Conclusions

This chapter investigated Deep Joint Source–Channel Coding (DJSCC) for point clouds as an alternative to the classical separate coding approach analyzed in Chapter 4. It was shown that the latent representations generated by the considered autoencoder architectures possess inherent robustness to noise in the latent space, making them suitable both for quantization and for direct transmission over an AWGN channel. The results demonstrate that DJSCC approaches provide graceful degradation of reconstruction quality under deteriorating channel conditions, thereby avoiding the characteristic *cliff-effect* observed in classical digital communication schemes. Furthermore, the DPCT architecture was shown to achieve better performance than the considered reference methods in the DJSCC regime, combining higher reconstruction quality with improved bandwidth efficiency [A6]. In addition, two approaches for frame synchronization during the transmission of dynamic point clouds were analyzed: synchronization using PRC and matched filtering, and synchronization-free phase-invariant decoding (PID). The results indicate that the matched-filter approach provides practically optimal synchronization performance, while PID offers a viable alternative in scenarios with small inter-frame variations and enables decoding without explicit synchronization recovery.

At the same time, the proposed phase-invariant decoding scheme inherently introduces information loss, since it removes the phase component by relying solely on the amplitude spectrum of the DFT. This limits the representational capability of the autoencoder and reduces its capacity to model dependencies in the data, potentially leading to a lower upper bound on achievable reconstruction quality. On the other hand, practical implementation of synchronization through matched filtering requires the introduction of an additional DC component into the transmitted signal in order to enable reliable frame boundary detection. The amplitude of this component depends on the noise level and the required synchronization reliability, and must increase under more adverse channel conditions. Under a total transmit power constraint, this reduces the power allocated to the useful signal and consequently degrades reconstruction quality. This leads to a fundamental tradeoff between the two approaches: PID avoids the need for an additional DC synchronization component, but at the cost of reduced representational capability due to the loss of phase information, whereas synchronization through matched filtering preserves the complete information content but requires additional transmission resources for synchronization. Therefore, future work should further investigate this tradeoff, including whether the reduced representational capacity of PID can be compensated by more efficient resource utilization under realistic channel conditions.

III. CONCLUSIONS AND CONTRIBUTIONS OF THE THESIS

The dissertation investigates efficient coding of 3D content through learned and semantic approaches for systems for capture, transmission, and visualization. Architectures for compression and direct transmission of point clouds are proposed, based on autoencoder architectures that improve coding efficiency at low bitrates and during transmission over noisy channels. The limitations of classical entropy coding are analyzed, and a framework based on mismatched probabilistic models is introduced, enabling the use of more complex models on the encoder side. It is also shown that the proposed methods provide robustness to noise and graceful quality degradation during transmission over noisy channels. Future research directions include the development of more efficient learned architectures, the integration of probabilistic models for entropy coding, the extension toward more complex dynamic scenes, and the investigation of tradeoffs associated with different synchronization methods.

SCIENTIFIC, SCIENTIFIC-APPLIED, AND APPLIED CONTRIBUTIONS

Scientific Contributions:

1. A theoretical framework for entropy coding with mismatched probabilistic models has been developed, and an efficiency condition has been derived that relates the benefit of using a more accurate encoder-side model to the amount of required side information.

2. A Phase-Invariant Decoding (PID) method for the transmission of dynamic point clouds has been proposed. The method eliminates the need for explicit frame synchronization recovery by using a representation invariant to cyclic shifts, for which an error bound has been derived.

Scientific-Applied Contributions:

1. A systematization and analysis of 3D content coding methods have been carried out, based on which a taxonomy has been proposed according to the employed technological principles and the interpretability of intermediate representations (Fig. 1.5). A conceptual model for semantic compression based on the separation of information into global and detailed semantic context has also been formulated.

2. A four-layer operational model (Fig. 2.1) has been applied as a universal framework for the analysis of systems for capture, transmission, and visualization of 3D content, with emphasis on the distribution of computational and communication resources.

3. A broad range of point cloud processing methods has been investigated, including FoldingNet-based, graph-convolutional, self-attention-based, and 3D-convolutional architectures, resulting in the proposed GAE, TGAE, and DPCT architectures. A total of 37 models were trained and compared under different latent space dimensionalities and training conditions.

4. The compression of sparse point clouds using autoencoder architectures has been investigated, analyzing the influence of the architecture, the number of parameters, and the latent space dimensionality on the rate–distortion relationship. Comparisons were performed against G-PCC [11], Draco [90], and Context-based Residual Coding and Implicit Neural Representation-based Refinement (CRCIR) [91]. The results (Table 4.3) show that AEPCC, and especially TGAE, achieves the best rate–distortion efficiency in the low-bitrate regime.

5. The DPCT architecture has been investigated in a Deep Joint Source–Channel Coding (DJSCC) regime for point clouds, demonstrating improvements over reference methods (SEPT [57], FoldingNet [76], and G-PCC [11]) in terms of reconstruction quality and bandwidth efficiency in the presence of channel noise (Fig. 5.5). Frame synchronization approaches, including phase-invariant decoding, were developed and compared, and their behavior under different noise levels and synchronization errors was analyzed.

Applied Contributions:

1. Analysis and implementation of RGB-D image compression methods through colorization and the use of image coding schemes, including evaluation of the influence of different colorization, multiplexing, and semantic-oriented processing strategies on the rate–distortion characteristics (the software implementation can be found in [B1]).

2. A software implementation of a system for coding the geometric structure of sparse point clouds (AEPCC) has been developed, including implementations of different types of autoencoder architectures, positional encoding mechanisms, data preparation and augmentation procedures, and a complete training, validation, and testing pipeline. The implementation of the autoencoders from [A5] is available in [B2], while the implementation of the autoencoder from [A6] is available in [B3].

3. A standalone software implementation of a separate coding scheme for the geometric structure within AEPCC has been developed, including component-wise scalar quantization of latent vectors, dequantization, packetization, serialization using Concise Binary Object Representation (CBOR), and secondary coding using DEFLATE. The software implementation is available in [B2].

ABBREVIATIONS

ABBREVIATION	DEFINITION	ABBREVIATION	DEFINITION
3D	three-dimensional	DJSCC	Deep Joint Source-Channel Coding
V-PCC	Video-based Point Cloud Compression	R-D	Rate-Distortion
G-PCC	Geometry-based Point Cloud Compression	BD-Rate	Bjøntegaard Delta Rate
JSCC	Joint Source-Channel Coding	BD-PSNR	Bjøntegaard Delta PSNR
HTC	Holographic Type Communication	PSNR	Peak Signal-to-Noise Ratio
HMD	Head-Mounted Display	XR	Extended Reality
RGB-D	Red-Green-Blue-Depth	AWGN	Additive White Gaussian Noise
kNN	k-Nearest Neighbours	DTAT	Discrete Time Analog Transmission
AEPCC	Autoencoder-based Point Cloud Coding	RRC	Root-Raised-Cosine
GAE	Graph Autoencoder	ISI	Inter-Symbol Interference
TGAE	Transformer Graph Autoencoder	PRC	Pseudorandom Code
ADAM	Adaptive Moment Estimation	DFT	Discrete Fourier Transform
DPCT	Dynamic Point Cloud Transmission	PID	Phase-Invariant Decoding
SEPT	Semantic Point Cloud Transmission	CRCIR	Context-based Residual Coding and Implicit neural representation based Refinement
SNR	Signal-to-Noise Ratio	CBOR	Concise Binary Object Representation

LIST OF PUBLICATIONS RELATED TO DISSERTATION WORK

[A1] I. Bozhilov, R. Petkova, K. Tonchev, and A. Manolova, “A Systematic Survey Into Compression Algorithms for Three-Dimensional Content,” IEEE Access, vol. 12, pp. 141604–141624, 2024. DOI: 10.1109/ACCESS.2024.3469549.

[A2] I. Bozhilov, “Semantic Compression for 3D Content: A Unified Conceptual Framework and Survey of Advanced Solutions,” in 2026 Joint International Conference on Digital Arts, Media and Technology with ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunication Engineering (ECTI DAMT & NCON), 2026, pp. 710–715. DOI: 10.1109/ECTIDAMTNCON67592.2026.11459979.

[A3] I. Bozhilov, R. Petkova, K. Tonchev, A. Manolova, and V. Poulkov, “HOLOTWIN: A Modular and Interoperable Approach to Holographic Telepresence System Development,” Sensors, vol. 23, no. 21, 2023, ISSN: 1424-8220. DOI: 10.3390/s23218692. Available: https://www.mdpi.com/1424-8220/23/21/8692
.

[A4] I. B. Bozhilov, R. R. Petkova, K. T. Tonchev, and A. H. Manolova, “Exploring Semantic-Aware Compression of RGBD Images Using Conventional Codecs,” in 2025 60th International Scientific Conference on Information, Communication and Energy Systems and Technologies (ICEST), IEEE, 2025, pp. 1–4.

[A5] I. Bozhilov, R. Petkova, K. Tonchev, A. Manolova, V. Poulkov, and H. Vincent Poor, “Autoencoder Architectures for Low-Rate Sparse Point Cloud Geometry Coding,” IEEE Access, vol. 13, pp. 214122–214140, 2025. DOI: 10.1109/ACCESS.2025.3646031.

[A6] I. Bozhilov, R. Petkova, K. Tonchev, and A. Manolova, “Learning-Based Dynamic Point Cloud Transmission,” in 2025 28th International Symposium on Wireless Personal Multimedia Communications (WPMC), 2025, pp. 1–6. DOI: 10.1109/WPMC67460.2025.11351024.

SOFTWARE IMPLEMENTATIONS

[B1] I. B. Bozhilov, R. R. Petkova, K. T. Tonchev, and A. H. Manolova, Software Implementation: Exploring Semantic-Aware Compression of RGBD Images Using Conventional Codecs, Available: https://github.com/Teleinfrastructure-Research-Lab/rgbd-fusion, Accessed: 2026-04-07, 2025.

[B2] I. Bozhilov, R. Petkova, K. Tonchev, A. Manolova, V. Poulkov, and H. Vincent Poor, AEPCC: Software Implementation of the System, Available: https://github.com/Teleinfrastructure-Research-Lab/aepcc, Accessed: 2026-04-07, 2026.

[B3] I. Bozhilov, R. Petkova, K. Tonchev, and A. Manolova, DPCT: Software Implementation of the System, Available: https://github.com/Teleinfrastructure-Research-Lab/lb-dpct, Accessed: 2026-04-21, 2026.

BIBLIOGRAPHY

[4] R. Petkova, I. Bozhilov, A. Manolova, K. Tonchev, and V. Poulkov, “On the Way to Holographic-Type Communications: Perspectives and Enabling Technologies,” IEEE Access, vol. 12, pp. 59236–59259, 2024. DOI: 10.1109/ACCESS.2024.3393124.

[44] A. Vaswani et al., “Attention Is All You Need,” Advances in Neural Information Processing Systems, vol. 30, 2017.

[55] J. Gao et al., “Finite-Blocklength Information Theory,” Fundamental Research, 2026.

[57] C. Bian, Y. Shao, and D. Gündüz, “Wireless Point Cloud Transmission,” in Proceedings of the 2024 IEEE 25th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC), 2024, pp. 851–855. DOI: 10.1109/SPAWC60668.2024.10694621.

[67] R. Petkova, V. Poulkov, A. Manolova, and K. Tonchev, “Challenges in Implementing Low-Latency Holographic-Type Communication Systems,” Sensors, vol. 22, no. 24, p. 9617, 2022.
[68] J. Kaplan et al., “Scaling Laws for Neural Language Models,” arXiv preprint arXiv:2001.08361, 2020.

[76] Y. Yang, C. Feng, Y. Shen, and D. Tian, “FoldingNet: Point Cloud Auto-Encoder via Deep Grid Deformation,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 206–215.

[79] I. Bozhilov, K. Tonchev, A. Manolova, and R. Petkova, “3D Human Body Models Compression and Decompression Algorithm Based on Graph Convolutional Networks for Holographic Communication,” in Proceedings of the 2022 25th International Symposium on Wireless Personal Multimedia Communications (WPMC), 2022, pp. 532–537. DOI: 10.1109/WPMC55625.2022.10014791.

[92] M. Gastpar, B. Rimoldi, and M. Vetterli, “To Code, or Not to Code: Lossy Source-Channel Communication Revisited,” IEEE Transactions on Information Theory, vol. 49, no. 5, pp. 1147–1158, 2003. DOI: 10.1109/TIT.2003.810631.

[95] Y. Shao and D. Gunduz, “Semantic Communications With Discrete-Time Analog Transmission: A PAPR Perspective,” IEEE Wireless Communications Letters, vol. 12, no. 3, pp. 510–514, 2023. DOI: 10.1109/LWC.2022.3232946.

[96] M. Liu, W. Chen, J. Xu, and B. Ai, “Real-Time Implementation and Evaluation of SDR-Based Deep Joint Source-Channel Coding,” in 2022 IEEE 96th Vehicular Technology Conference (VTC2022-Fall), 2022, pp. 1–5. DOI: 10.1109/VTC2022-Fall57202.2022.10012971.

[97] G. Fernandes, H. Fontes, and R. Campos, Semantic Communications: The New Paradigm Behind Beyond 5G Technologies, arXiv:2406.00754, 2024.

TECHNICAL UNIVERSITY OF SOFIA

FACULTY OF TELECOMMUNICATIONS

DEPARTMENT "RADIOCOMMUNICATIONS AND VIDEOTECHNOLOGIES"

Ivaylo Bozhidarov Bozhilov, MSc

ENCODING AND VISUALIZATION OF 3D OBJECTS USING DEEP LEARNING ARCHITECTURES

ABSTRACT of PhD THESIS

This thesis investigates the encoding and visualization of 3D objects using deep learning architectures, with a focus on efficient compression, transmission, and reconstruction of sparse point clouds. It analyzes classical, learning-based, and semantic approaches for 3D content coding and introduces an operational model for systems for acquisition, transmission, and visualization of three-dimensional content.

The work proposes and evaluates autoencoder-based architectures for point cloud geometry coding in both separate source coding and deep joint source-channel coding scenarios. The results demonstrate efficient low-rate compression, robustness to channel noise, and graceful degradation of reconstruction quality. The dissertation also addresses synchronization in dynamic point cloud transmission and proposes methods based on pseudo-random coding and phase-invariant decoding.