Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук
(повна назва)

Кафедра Штучного інтелекту
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти другий (магістерський)

Дослідження Point-based Neural Radiance Fields для реконструкції великих 3Д сцен з невідомими позами камер за допомогою нейронного bundle adjustment
(тема)

Виконав:
студент 2 курсу, групи СШІм-21-1
Шашко О. В.
(прізвище, ініціали)

Спеціальність 122 Комп'ютерні науки
(код і повна назва спеціальності)

Тип програми освітньо-наукова
(освітньо-професійна або освітньо-наукова)

Освітня програма Системи штучного інтелекту
(повна назва спеціалізації)

Керівник проф. Семенець В. В.
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри _____ В.О. Філатов
(підпис) (прізвище, ініціали)

2023 р.

Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук
(повна назва)

Кафедра Штучного інтелекту
(повна назва)

Рівень вищої освіти другий (магістерський)

Спеціальність 122 Комп'ютерні науки
(код і повна назва)

Тип програми освітньо-наукова
(освітньо-професійна або освітньо-наукова)

Освітня програма Системи штучного інтелекту (СШІ)
(повна назва)

ЗАТВЕРДЖУЮ:
Зав. кафедри _____
(підпис)
«_____» _____ 20 ___ р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові Шашко Олексію В'ячеславовичу
(прізвище, ім'я, по батькові)

1. Тема роботи Дослідження Point-based Neural Radiance Fields (NeRF) для реконструкції великих 3Д сцен з невідомими позами камер за допомогою нейронного bundle adjustment

затверджена наказом університету від 31 березня 20 23 р. № 306Ст

2. Термін подання студентом роботи до екзаменаційної комісії 16 травня 20 23 р.

3. Вихідні дані до роботи Науково-технічні публікації, дані Інтернет-джерел та відомих наукових проектів щодо реконструкції зображень та 3Д сцен за допомогою NeRF, сучасні архітектури Neural Radiance Fields, тестувальні набори даних та функції розрахунку метрик якості.

4. Перелік питань, що потрібно опрацювати в роботі
1 Аналіз пердметної області та постановка задачі
2 Обґрунтування та опис моделі
3 Опис практичних експериментів
4 Впровадження результатів та перспективи розвитку

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (п.5 включається до завдання за рішенням випускової кафедри) Рисунок 1.1 – Огляд роботи мережі NeRF, Рисунок 1.2 – Архітектура PixelNeRF, Рисунок 1.3 – Архітектура RegNeRF, Рисунок 1.4 – Схема навчання GNeRF, Рисунок 1.5 – Схема навчання NeRF--, Рисунок 1.6 – Архітектура PointNeRF, Рисунок 1.7 – Схематичне зображення роботи методу коригування променів, Рисунок 2.1 – Приклад неламбертовських ефектів відтворених за допомогою моделі NeRF, Рисунок 2.2 – Схема БШП в моделі NeRF, Рисунок 2.3 – Візуалізація рузільтатів відновлення 3Д сцени, Рисунок 2.4 – Схема прогнозування випроміненого кольору і об'ємної щільності в моделі PNeRF, Рисунок 2.5 – Результати процесу «обрізки» нейронної хмарини точок, Рисунок 2.6 – Результати процесу «нарощування» нейронної хмарини точок, Рисунок 2.7 – Схема прогнозування випроміненого кольору і об'ємної щільності в модифікованій моделі PNeRF, Рисунок 3.1 – Приклад сцен набору даних ScanNet, Рисунок 3.2 – Приклад зі сцени 0005_00 набору даних ScanNet, Рисунок 3.3 – Приклад сцени 0010_00 набору даних ScanNet.

6. Консультанти розділів роботи (п.6 включається до завдання за наявності консультантів згідно з наказом, зазначеним у п.1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Отримання завдання	03.04.2023	Виконано
2	Аналіз предметної області	04.04.2023-09.04.2023	Виконано
3	Постановка задачі	10.04.2023-12.04.2023	Виконано
4	Вибір моделей для дослідження	13.04.2023-15.04.2023	Виконано
5	Проектування механізмів модифікації моделі	16.04.2023-18.04.2023	Виконано
6	Програмна реалізація моделей та алгоритмів	19.04.2023-21.04.2023	Виконано
7	Планування та підготовка експериментів	21.04.2023-22.04.2023	Виконано
8	Проведення експериментів	22.04.2023-29.04.2023	Виконано
9	Підготовка пояснювальної записки	27.04.2023-09.05.2023	Виконано
10	Надання записки на перевірку керівнику	10.05.2023-15.05.2023	Виконано
11	Захист кваліфікаційної роботи	16.05.2023	Виконано

Дата видачі завдання 3 квітня 2023 р.

Студент _____
(підпис)

Керівник роботи _____ проф. Семенець В. В.
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка: 81 с., 32 рис., 3 табл., 2 дод., 36 джерело.

БАГАТОШАРОВІ НЕЙРОННІ МЕРЕЖІ, ГЛИБИННЕ НАВЧАННЯ, КОРЕГУВАННЯ ПОЗИ КАМЕРИ, НЕЙРОННА ХМАРИНА ТОЧОК, ПОТОЧКОВИЙ NERF, ШТУЧНА НЕЙРОННА МЕРЕЖА

Об'єкт дослідження – архітектура та навчання поточкових Neural Radiance Fields.

Предмет дослідження – методи моделювання поточкових Neural Radiance Fields та алгоритми їх навчання.

Мета роботи – покращення якості генерації 3Д структур при використанні неякісних (або повної відсутності) поз камер у поточкових Neural Radiance Fields.

Методи дослідження – аналіз існуючих підходів для генерації 3Д структур за допомогою набору зображень, розробка архітектури та алгоритму навчання поточкових Neural Radiance Fields та нейронних мереж в їх складових, програмна реалізація та проведення експерименту, обробка та аналіз отриманих результатів.

Під час виконання кваліфікаційної роботи проведено аналіз літературних джерел в області Neural Radiance Fields та алгоритмів їх навчання, наукових публікацій щодо розробки та проектування такого виду систем нейронних мереж для генерації 3Д об’єктів з набору зображень. Було виділено основні недоліки існуючих підходів та запропоновано нову архітектуру – поточковий Neural Radiance Fields та алгоритм його навчання, який в процесі корегує (або визначає) пози камер відповідних зображень. Реалізовано програмний модуль моделі зі всіма необхідними алгоритмами та протестовано на відповідних відкритих даних. На основі результатів був проведений порівняльний аналіз якості запропонованого підходу.

ABSTRACT

Explanatory note: 81 p., 32 fig., 3 tabl., 3 ann., 36 sources.

BUNDLE ADJUSTMENT, MULTILAYER PERCEPTRON, NEURAL POINT CLOUD, NEURAL RADIANCE FIELDS, NEURAL (DENSE) BUNDLE ADJUSTMENT, POINT-BASED NEURAL RADIANCE FIELDS

The object of research is the architecture and training of Point-based Neural Radiance Fields.

The subject of research is methods of modeling streaming Point-based Neural Radiance Fields and algorithms for their training.

Purpose – to improve the quality of 3D structure generation when using imperfect camera poses (or without poseses at all) in Point-based Neural Radiance Fields.

Research methods – analysis of existing approaches for generating 3D structures using a set of images, development of architecture and algorithm for training Point-based Neural Radiance Fields and neural networks as their components, software implementation and experiment, processing and analysis of the results.

During the qualification work, the author analyzed literature sources in the field of Neural Radiance Fields and algorithms for their training, scientific publications on the development and design of this type of neural network systems for generating 3D objects from a set of images. The main shortcomings of existing approaches were identified and a new architecture was proposed – Point-based Neural Radiance Fields and its training algorithm, which in the process corrects (or determines) the camera poses of the corresponding images. The program module of the model with all the necessary algorithms was implemented and tested on relevant open data. Based on the results, a comparative analysis of the quality of the proposed approach was carried out.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

NeRF – Neural Radience Fields;

BA – Bundle Adjustment – корегування променів;

BARF – Bundle Adjusting Neural Radience Fields;

GAN – Generative Adversarial Network – генератовно змагальна мережа;

LLFF – Local Light Field Fusion – локальний синтез світлового поля;

LPIPS – Learned Perceptual Image Patch Similarity – Вивчена перцептивна схожість патчів зображень;

MLP або БШП – Multilayer Perceptron – багатошаровий персептрон;

MSE – Mean Square Error – середньоквадратична похибка;

MVS – Multi-view stereo – багатовидова стереосистема;

NBA – Neural (Dense) Bundle Adjustment – нейронне корегування променів;

PNeRF або PointNeRF – Point-based Neural Radience Fields – поточковий Neural Radience Fields;

PSNR – Peak Signal-to-Noise Ratio – пікове співвідношення сигналу до шуму;

ReLU – Rectified Linear Unit – зрізаний лінійний вузол;

SSIM – Structural Similarity Index – індекс структурної подібності.

ВСТУП

В останній час інформаційні технології, а особливо системи штучного інтелекту, з’являються у все більшій кількості сучасних процесів. Тому не дивно, що зараз починає зростати попит на системи ШІ для задач генерування та оперування 3Д об’єктів. Такі моделі та технології можуть значно полегшити роботу дизайнерів, архітекторів, ігрових дизайнерів, тощо.

Так як 3Д сцена або 3Д об’єкт – це доволі складна структура, тому для створення та маніпулювання ними треба або дороге обладнання, або спеціаліст і багато часу. Для спрощення цієї ситуації зараз активно розробляються методи комп’ютерного зору в області просторового розуміння та орієнтації. Одним із таких рішень є Neural radiance fields.

Neural radiance fields (NeRF) набули популярності в останні роки як потужний інструмент для рендерингу реалістичних 3Д сцен з набору 2Д зображень. NeRF – це метод, заснований на глибокому навчанні, який моделює об'ємну щільність і колір сцени шляхом навчання нейронної мережі на наборі вхідних зображень, знятих з різних точок зору на об’єкт. Це дозволяє якісно відтворювати нові види сцени, які не були захоплені оригінальними зображеннями.

Однією з проблем використання NeRF є точна оцінка пози камери та її внутрішніх параметрів, які необхідні для точної 3Д реконструкції та рендерингу. В подальшому дослідженні буде описано використання методу коригування променів (bundle adjustment), класичного методу оптимізації в комп'ютерному зорі, для уточнення параметрів камери і підвищення точності NeRF реконструкцій.

Зокрема, буде детально розглянуто самокалібрування, яке передбачає оцінку внутрішніх параметрів камери (таких як фокусна відстань і головна точка), використовуючи лише відповідності зображень, не вимагаючи попередніх знань про камеру або ціль калібрування. Також буде

проаналізовано та досліджено можливість використання методу коригування променів для спільної оптимізації положення камери та її внутрішніх параметрів, що призводить до більш точних реконструкцій NeRF зображень.

Таким чином результатом дослідження буде демонстрація ефективності поєднання методів, заснованих на глибокому навчанні, таких як NeRF, з класичними методами комп'ютерного зору, такими як коригування променів, і підкреслити потенціал для подальших досягнень у цій галузі.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ

1.1 Аналіз існуючих методів представлення 3Д геометрії

Перспективним напрямком комп'ютерного зору останнім часом є кодування об'єктів і сцен у вагах багатошарового персептрону, які безпосередньо відображають тривимірне просторове розташування в неявне представлення форми, наприклад, орієнтовану відстань [3] в цьому розташуванні. Однак ці методи не можуть відтворювати реалістичні сцени зі складною геометрією з такою ж точністю, як методи, що представляють сцени за допомогою дискретних зображень, таких як трикутні або воксельні сітки.

Подібний підхід використання багатошарового персептрону для відображення низьковимірних координат у кольори також використовувався для представлення інших графічних функцій, таких як зображення [27], текстуровані матеріали [11] та значення непрямої освітленості [9].

Деякі роботи досліджували неявне представлення неперервних 3Д форм у вигляді наборів рівнів шляхом оптимізації глибинних мереж, які відображають координати xyz у орієнтовані функції відстані [17] або поля заповнюваності [16]. Однак ці моделі обмежені вимогою наявності реальної 3Д геометрії, яку зазвичай отримують з наборів даних синтетичних 3Д форм, таких як ShapeNet. Подальші роботи пом'якшили це обмеження, сформулювавши диференційовані функції рендерингу, які дозволяють оптимізувати нейронні неявні представлення форми, використовуючи лише 2Д зображення. Деякі з досліджень [7] представляють поверхні як 3Д поля заповнення і використовують чисельний метод для знаходження перетину поверхні для кожного променя, а потім обчислюють точну похідну, використовуючи неявне диференціювання. Кожне місце перетину променів подається як вхідні дані для нейронного поля 3Д текстури, яке прогнозує дифузний колір для цієї точки. Інші дослідження [5] використовують менш

пряме нейронне 3Д представлення, яке просто виводить вектор ознак і колір RGB для кожної неперервної 3Д координати, і пропонують диференційовану функцію рендерингу, що складається з рекурентної нейронної мережі, яка рухається вздовж кожного променя, щоб визначити, де знаходиться поверхня.

Хоча ці методи потенційно можуть відображати складну геометрію з високою роздільною здатністю, досі вони обмежувалися простими формами з низькою геометричною складністю, що призводило до згладжених результатів рендерингу. На відміну від них, Neural Radiance Fields показує, що альтернативна стратегія оптимізації мереж для кодування 5D полів випромінювання (3D об'єми з 2D залежним від виду виглядом) може представляти геометрію з високою роздільною здатністю і генерувати фотореалістичні нові види складних сцен.

За наявності щільної вибірки видів нові фотореалістичні види можна реконструювати за допомогою простих методів інтерполяції вибірки поля [29]. Для синтезу нових видів з більш розрідженою вибіркою спільноти комп'ютерного зору та графіки досягли значного прогресу, прогнозуючи традиційні геометричні та текстурні представлення на основі спостережуваних зображень. Один з популярних підходів використовує меш представлення сцен з дифузним [31] або залежним від виду [30] виглядом.

Диференційовані растеризатори [2] або трасувальники шляхів [6] можуть безпосередньо оптимізувати меш представлення для відтворення набору вхідних зображень за допомогою градієнтного спуску. Однак, градієнтна оптимізація мешу на основі репроекції зображень часто є складною, ймовірно, через локальні мінімуми або погану обумовленість ландшафту функції втрат. Крім того, ця стратегія вимагає шаблонного мешу з фіксованою топологією, яка повинна бути надана в якості ініціалізації перед оптимізацією [6], що, як правило, недоступно для реальних сцен.

Інший клас методів використовує об'ємні уявлення для вирішення завдання високоякісного фотореалістичного синтезу зображення з набору

вхідних RGB зображень. Об'ємні підходи здатні реалістично відображати складні форми і матеріали, добре підходять для градієнтної оптимізації і, як правило, створюють менше візуально відволікаючих артефактів, ніж методи на основі мешу.

Ранні об'ємні підходи використовували наявні зображення для безпосереднього розфарбовування воксельних сіток [15]. Нещодавно кілька методів [4], [26] почали використовувати великі набори даних з декількох сцен для навчання глибинних мереж, які прогнозують вибіркове об'ємне представлення з набору вхідних зображень, а потім використовують або альфа-композицію [23], або навчену композицію вздовж променів для візуалізації нових видів під час генерації нових зображень.

Інші роботи оптимізували комбінацію згорткових мереж (CNN) і вибіркових воксельних сіток для кожної конкретної сцени, так що CNN може компенсувати артефакти дискретизації воксельних сіток низької роздільної здатності [5] або дозволити прогнозованій воксельній сітці змінюватися в залежності від часу [18]. Хоча ці об'ємні методи досягли вражаючих результатів у синтезі нових видів, їхня здатність масштабуватися до зображень з вищою роздільною здатністю принципово обмежена низькою часовою та просторовою складністю через дискретизацію – рендеринг зображень з вищою роздільною здатністю вимагає більш щільною вибіркою 3Д простору.

Neural Radiance Fields обходить цю проблему шляхом кодування неперервного об'єму в межах параметрів глибокої повнозв'язної нейронної мережі, що не тільки забезпечує значно вищу якість рендерингу, ніж попередні об'ємні підходи, але й вимагає лише частку вартості зберігання вибіркових об'ємних зображень.

1.2 Огляд моделей Neural Radiance Fields

В базовому варіанті моделі NeRF (Neural Radiance Fields) [20] неперервна сцена представлена як 5-ти вимірна векторна функція, входом якої є 3Д локація x = (x, y, z) і 2Д напрямок огляду (θ, φ), а виходом – випромінений колір c = (r, g, b) і щільність об'єму σ. На практиці напрямок представлений як 3Д декартовий одиничний вектор d.

Це неперервне 5-ти вимірне представлення сцени апроксимується за допомогою багатошарового персептрону, ваги якого оптимізуються таким чином, щоб зіставити кожну вхідну 5D координату з відповідною щільністю об'єму та спрямованому випроміненному кольору (рис. 1.1).

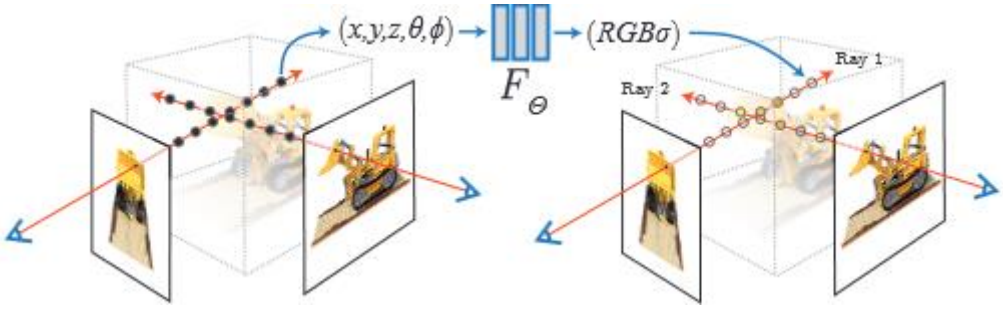

Рисунок 1.1 – Огляд роботи мережі NeRF

Іншими словами, кожна 5-ти вимірна координата повинна відповідати кольору та щільності об’єкта в точці перетину проміня і об’єкта.

Можна виділити такі основні недоліками цієї архітектури:

– для навчання мережі треба дуже багато фотографій 3Д об’єкта з різних ракурсів;

– дуже великий час тренування;

– обов’язково необхідно точні пози камери на кожне зображення із тренерувального набору даних;

– не може працювати з великими сценами (такими як 3Д

представлення кімнати, тощо).

Далі розглянемо модифікації цього підходу для вирішення зазначених проблем.

1.2.1 Архітектури NeRF адаптовані до невеликих тренувальних наборів даних

Однією з основних досягнень в цьому напрямку є архітектура PixelNeRF (рис. 1.2) [21]. Основним досягненням цієї роботи є те, що автори знайшли спосіб претренеровувати мережу NeRF для того, щоб скоротити час та розмір навчальної вибірки.

Цей ефект досягається шляхом впровадження згорткової підмережі в архітектуру, яка формує піксельну сітку ознак вхідного зображення. Ці ознаки і використовуються в багатошаровому персептроні (БШП) NeRF для визначення кольору і щільності об’єму точок на промені.

За рахунок цього БШП навчається не на абсолютних значеннях положення тривимірної точки, а на векторі ознак цієї точки. Таким чином попереднє навчання згорткової підмережі і БШП робить можливим генерацію нових видів основуючись лише на одному або декількох зображень 3Д сцени.

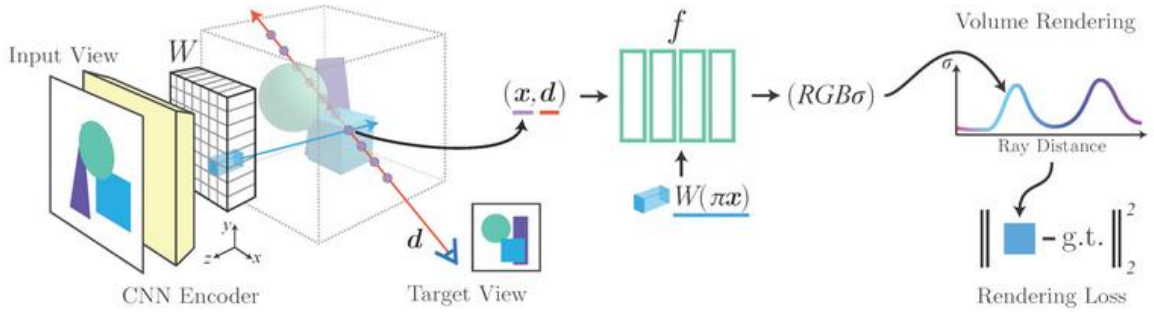

Рисунок 1.2 – Архітектура PixelNeRF

Явним недоліком є те що процедура визначення відповідного вектора ознак для конкретної точки має високу обчислювальну складність. А за умови, якщо в тренувальному наборі більше одного зображення, ми додаємо проблему зберігання створених сіток ознак.

Інша важлива робота в цьому напрямку описує методи регуляризації мережі NeRF для її якісної роботи в розрідженому просторі навчальної вибірки. Як результат цього дослідження було представлено архітектуру RegNeRF (рис. 1.3) [24].

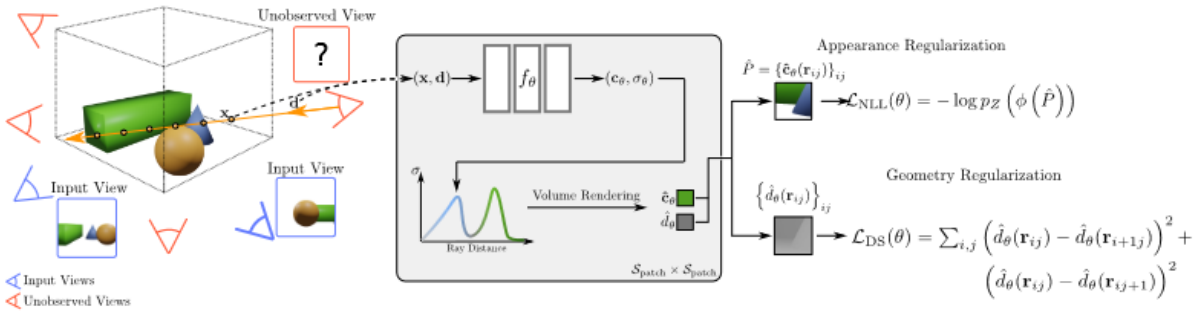

Рисунок 1.3 – Архітектура RegNeRF

Особливістю цього підходу є те, що він використовує для навчання штучно згенеровані види сцени. Ці нові види і лягають в основу регуляризації: по ним генерується певний набір RGB кольорів та щільностей, які потім рендерються у вихідні кольори та карти глибин.

Першим елементом функції втрат, після фотометричної функції схожості, є геометрична регуляризація, основна ідея якої – в основному поверхні плоскі та не мають дуже різких перепадів. Тому, ця функція перевіряє середньоквадратичне відхилення від сусідніх прогнозів глибини.

За допомогою попередньо натренованої моделі нормалізованого потоку RealNVP [8] оцінюється логарифмічна правдоподібності вихідних кольорів. Задача оптимізації – максимізувати це значення. Таким чином ми позбуваємося великої деградації в генерації кольору нового виду.

На відміну від інших варіантів архітектури, ця дає можливість навчатися на невеликій і розрідженій навчальній вибірці без попереднього навчання. Але недоліками такої архітектури є необхідність попередньо навченої моделі оцінки правдоподібності. Як результат ми отримуємо відносно кращі результати, але не ідеальні. Також архітектура не можу працювати з великими 3Д сценами.

1.2.2 Архітектури NeRF адаптовані до неточних або невідомих поз камери

Першими спробами вирішити проблему з неточними або невідомими позами камер стала архітектура GNeRF [10] – генеративно змагальна версія NeRF (рис. 1.4).

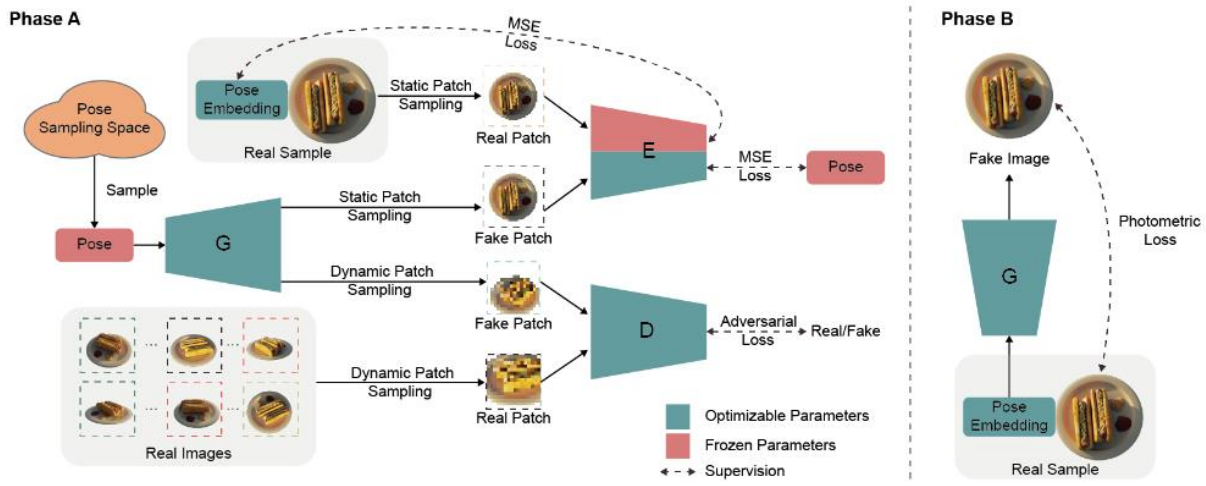

Рисунок 1.4 – Схема навчання GNeRF

Особливістю цієї архітектури є її процес навчання. Вона використовує генеративно змагальний алгоритм навчання: спочатку довільно вибирається поза камери, далі на основі цієї пози NeRF генерує зображення, яке подається на дискримінатор (мережу, завдання якої визначити, яка картинка реальна, а яка згенерована), також цей дискримінатор отримує реальні

картинки з тренувального набору даних для коректного навчання. Ця зв’язка дозволяє моделі NeRF зрозуміти, які існують зображення (а отже і пози) існують в тренувальних даних.

Також з моделі NeRF зображення ідуть на іншу модель дискримінатор, завдання якої на основі зображення відтворити позу камери. Після тренування цей дискримінатор використовується для визначення поз камери навчального набору зображень.

Після цього починається 2 фаза тренування, в якій стандартним чином оптимізується за допомогою фотометричної функції втрат NeRF мережа використовуючи визначені для тренувального набору даних пози камери.

Головним недоліком цього підходу очевидно є його складність навчання, реалізації і всі проблеми та недоліки, які виникають під час роботи з генеративно змагальними мережами.

Схожою ідеєю вирішення поставленої проблеми є NeRF-- (рис. 1.5). Основною особливістю цього підходу є те, що автори, зробивши ряд припущень, оптимізують параметри камери та її позу разом з мережею NeRF використовуючи лише фотометричну функцію втрат [19].

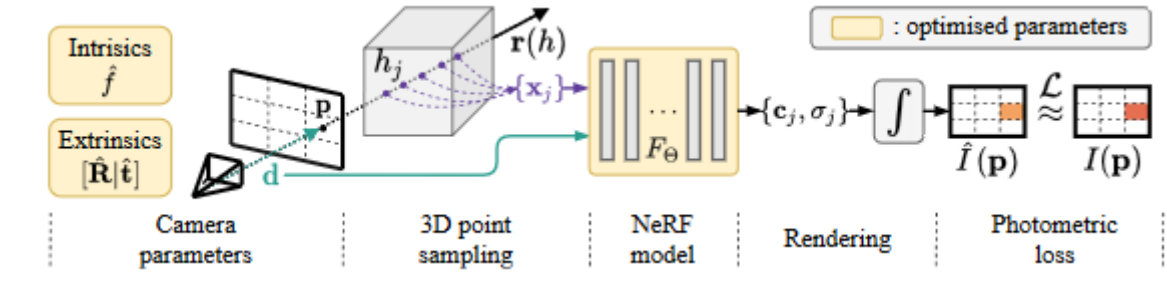

Рисунок 1.5 – Схема навчання NeRF--

Основна перевага цієї архітектури – це її простота, яка можлива лише з тими припущеннями, які були зроблені. Але вони ж є і недоліками.

Всього сформульовано 2 припущення: перший – всі зображення повинні бути зроблені відносно однієї 3Д площини (тобто з одного боку

об’єкта), що робить неможливим повне і акуратне відтворення 3Д об’єкта та роботу з великими сценами. Друге – всі зображення зроблені камерою із одними і тими ж внутрішніми параметрами.

Іншим підходом до вирішення цієї проблеми є комбінація алгоритму корегування променів і класичного NeRF. Така архітектура була створена та має назву Bundle-adjusting NeRF (BARF) [1].

Особливостями цього підходу є те, що він уточнює, у випадку якщо є, позу або відтворює, у випадку якщо нема, її за допомогою алгоритму корегування променів. А мережу NeRF адаптує бути несприятливою до невеликих коливань пози камери за допомогою регулярізації функції втрат. Паралельна оптимізація дає можливість корегувати пози камер та паралельно вивчати більш детальне представлення 3Д сцени.

Для того щоб мережа NeRF не перенавчилася на неточних позах (до того як вони виправляться), використовують нелінійне зважування позиційного кодування вхідних даних. Тобто в процесі тренування буде поступово плавно активуватися позиційне кодування вхідних даних, тим самим навчаючи мережу спочатку на даних закодованих у низьких частотах, що відповідає абстрактним формам та грубим контурам, поступово додаючи все більше і більше деталей.

В той же час L2 регуляризація буде допомагати функції втрат менше реагувати на позу камери, що робить мережу більш стабільною до некоректних вхідних даних.

Недоліком такого підходу є те, що в основі архітектури лежить класичний БШП NeRF, який накладає обмеження на розмір 3Д сцени.

1.2.3 Архітектури NeRF адаптовані для роботи з великими сценами

Для вирішення цієї проблеми, треба відійти від класичного представлення NeRF, а саме від кодування 3Д сцени у ваги багатошарового персептрону. Дуже гарним рішенням є представлення конвертування

2Д зображень в зручну для роботи з 3Д структуру зі збереженням двовимірних ознак. Такий підхід реалізували в архітектурі Point-based Neural Radiance Fields (PointNeRF) (рис. 1.6) [22].

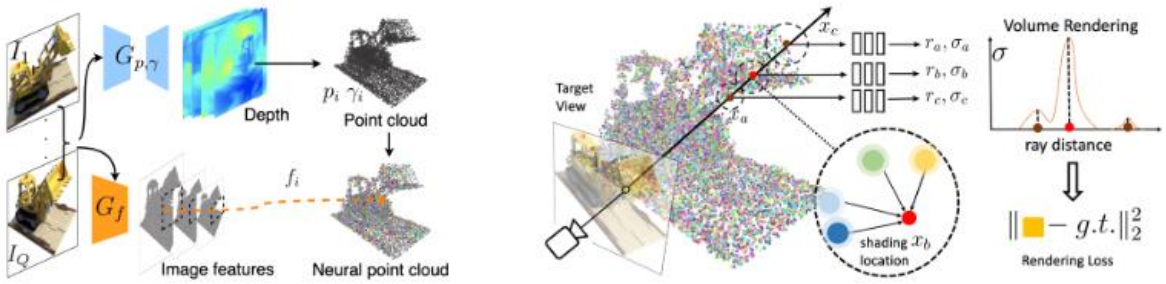

Рисунок 1.6 – Архітектура PointNeRF

В якості структури збереження даних було використано хмарину точок, де кожна точка являє собою вектор ознак відповідного пікселя певного зображення з тренувального набору даних.

Щоб сформувати таку структуру використовується 2 мережі: Multiview Stereo (MVS) [12], для прогнозування карт глибин і формування хмарини точок, і звичайний попередньо навчений згортковий енкодер, для формування карт ознак вхідних зображень. Отримана структура називається – нейронна хмарина точок.

Для побудови нових видів за допомогою цієї структури, як і у звичайному NeRF, посилається промінь в нейронну хмарину точок. Де для визначення кольору і щільності та для уникнення ситуації трасування в пустий простір, в певному радіусі обираються К найближчих точок та по кожній з них прогнозується вектор ознак відносно променю на окремому БШП. Потім ці К кандидатів зважуються відносно відстані до променю та подаються на мережу NeRF, яка і визначає колір в обраній точці на промені.

Таким чином вдається не тільки представити 3Д сцену бідь-якого розміру та складності, а й досягти гарної якості генерації нових зображень,

уникаючи проблем розрідженості хмарини точок.

Недоліком цього методу є сильна залежність від поз камери та наявність помилки мережі MVS для формування нейронної хмарини точок. Але проблему з помилками на MVS можна уникнути використовуючи карти глибин отримані під час формування тренувального набору (наприклад з LiDAR або стереопарою). Але цей підхід вимагає дорогого обладнання і спеціалістів.

1.3 Метод корегування променів

Метод корегування променів [14] – це метод оптимізації, який використовується в комп'ютерному зорі для уточнення внутрішніх параметрів камери, її пози в просторі та 3Д структури сцени на основі набору зображень 2Д зображень та 3Д опорних точок (рис. 1.7).

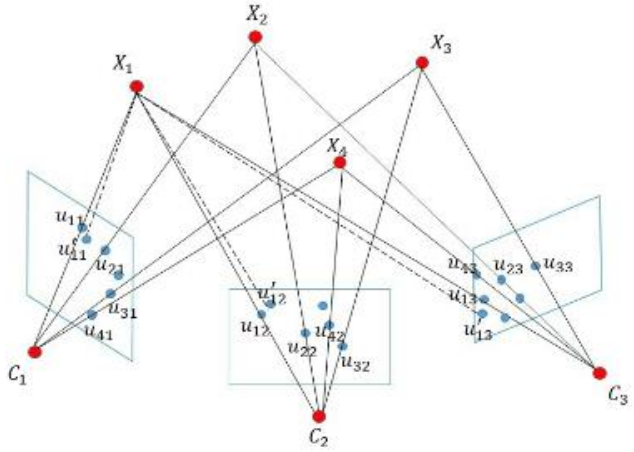

Рисунок 1.7 – Схематичне зображення роботи методу коригування променів

Цей використовується для підвищення точності 3Д реконструкції шляхом мінімізації помилки відтворення між спостережуваними точками зображення і спроектованими 3Д точками. Методика передбачає одночасне

налаштування параметрів камери та 3Д структури для мінімізації похибки між спостережуваними та прогнозованими точками зображення.

Метод корегування пакету є ітеративним процесом і зводиться до нелінійної задачі найменших квадратів. Класичними методами для вирішення подібних задач є Ґаусс-Ньютона, але він є обчислювально нестабільним, або Левенберґа-Марквардта. Останній же широко використовується для вирішення невеликих (в плані кількості змінних) задач корегування променів.

Але при зростанні кількості поз камер та кількості опорних точок, можна визначити, що матриця Якобі, яка лежить в основі метода Левенберґа-Марквардта, має розріджену структуру. Так як параметри камер та їх пози незалежні одне від одного.

Тому для обчислювальної оптимізації методу коригування променів та формування зменшеної системи камер використовують доповнення Шура та розклад Холецького з подальшим розв’язанням вже лінійної системи рівнянь.

Подальший розвиток цього методу приводить до його інтегрування в нейронні мережі [28], [25], де він ітераційно оптимізується за допомогою механізму зворотного розповсюдження помилки. Це дає можливість використовувати його як частину більш складної мережі для уточнення параметрів та поз камери навчаючи одну наскрізну модель.

1.4 Постановка задачі

Виходячи з усього вище зазначеного, можна сказати, що Neural Radiance Fields – один з найбільш актуальних, точних і простих методів представлення та реконструкції 3Д сцен з 2Д зображень. Маючи ряд недоліків він породив цілий напрям досліджень, які модифікують його таким чином, щоб виправити його недоліки зі збереженням високої якості.

Найбільш актуальним використанням цих систем в області науки про

дані та комп’ютерного зору – це генерація майже нескінченних наборів даних з обмеженого, а як виявилося, і відносно невеликого набору 2Д зображень, що може значно полегшити етап збору та підготовки даних в сферах розпізнавання зображень, відео, просторового розуміння, тощо.

В інших сферах, ця архітектура спрощує процес створення або відтворення 3Д сцен та об’єктів в області дизайну, архітектури, ігрової індустрії, тощо. Це може звести процес створення 3Д моделі будь-якого об’єкта до фотографування цього об’єкта з різних ракурсів та подачу цього набору зображень в систему.

Основною проблемою, яка залишається на поточний момент часу, є неможливість існуючих архітектур та модифікацій якісно відтворювати великі 3Д сцени не використовуючи при цьому дороге та спеціальне обладнання.

Оскільки мета цієї роботи – покращення якості генерації 3Д структур при використанні неякісних (або повної відсутності) поз камер у Neural Radiance Fields, було прийнято рішення за основу взяти поточкові NeRF (Point-NeRF), так як вони гарно адаптовані для роботи з великими 3Д сценами.

Для досягнення поставленої мети необхідно опрацювати наступні питання:

- сформувати та описати архітектуру системи точкової NeRF для сформульованої задачі та механізм її навчання;
- обґрунтувати обрані методи та механізми;
- програмно реалізувати та навчити отриману модель;
- протестувати отриману модель та проаналізувати результати.

Таким чином, основним завданням магістерської кваліфікаційної роботи є дослідження, реалізація та доведення працездатності системи точкової NeRF для реконструкції великих 3Д сцен (таких як кімната) з неточними (або невідомими) позами камер.

2 ОБҐРУНТУВАННЯ ТА ОПИС МОДЕЛІ

2.1 Обґрунтування вибору моделі

Як зазначалося раніше через перспективність і доволі широкі можливості моделей Neural Radiance Fields (далі NeRF) вони зараз являються одним із основних напрямків розвитку комп’ютерного зору та штучного просторового сприйняття.

На відміну від класичної хмарини точок, NeRF може відтворювати значно більше деталей об’єкту та складнішу текстуру. Так, наприклад, хмарина точок не здатна ні в якому вигляді працювати зі сценами, які містять оклюзії або віддзеркалення. Основною перевагою хмарин точок є їхня здатність зберігати необмежену, в певній мірі, кількість інформації, що потенційно дає можливість оперувати більшими 3Д об’єктами або сценами об’єктів.

Моделі на основі мешу містять більше деталей ніж хмарини точок, але потребують значно більше даний, щоб сформувати високоякісний меш, який в той же час робить операції над ним дуже обчислювально затратними. У цьому плані NeRF потребує значно менше машинного часу та даних для досягнення такого ж рівня якості.

Моделі на основі вокселів мають меншу обчислювальну складність. Однак роздільна здатність вихідного зображення напряму обумовлена дискретною природою вокселя. А значить ці моделі втрачають можливість відтворювати високо деталізовані сцени та об’єкти, на відміну від NeRF моделей, які здатні моделювати неперервні функції, що дає можливість покривати значно більший спектр деталізації об’єкту.

Багатовидові стереосистеми (multi-view stereo, далі MVS) дуже розповсюджена технологія для 3Д реконструкції, в основі якої лежить комбінування просторово або часово залежних зображень сцени або об’єкту.

Але, як і у випадку з моделями на основі мешу, це сімейство систем потребує великої кількості вхідних зображень для формування деталізованого виходу.

В порівнянні з генеративно змагальними мережами (далі GAN) NeRF моделі мають значно дешевший та простіший механізм навчання. Також в порівнянні з генеративними моделями NeRF не має на меті узагальнити всі можливі об’єкти та сцени для подальшої їх реконструкції, що робить можливим відтворення більш деталізовані сцени зі значно меншими витратами часу та потребами в даних.

Тому виходячи з усього вище описаного, було прийнято рішення обрати саме системи NeRF як основу дослідження вирішення поставленої проблеми.

2.2 Опис моделі Neural Radiance Fields

Модель Neural Radiance Fields представляє неперервну сцену у вигляді 5-тивимірної функції, на вхід якої подається 3Д положення точки $\mathrm{x} = (x, y, z)$ і 2Д напрямок огляду (θ, ϕ) з цієї точки, а виходом моделі є випромінюваний колір $c = (r, g, b)$ та щільність простору σ в даній точці. Але на практиці для визначення напрямку огляду використовують 3Д декартовий одиничний вектор d. Апроксимується ця неперервна 5Д сцена за допомогою нейронної мережі, яка складається з багатошарового персептрону (далі БШП або MLP) $F_\theta: (\mathrm{x}, d) \rightarrow (c, \sigma)$, і оптимізує ваги θ так, щоб кожна 5-тивимірна координата відображала відповідні їй випромінюваний колір та об’ємну щільність.

Для досягнення узгодження між багатьма видами NeRF обмежує мережу таким чином, щоб прогнозувати щільність простору σ як функцію лише від 3Д положення точки x, в той час як RGB колір c передбачається як функція від і від положення точки, і від напрямку огляду. Для досягнення цього ефекту, БШП F_θ спочатку обробляє вхідні 3Д координати за

допомогою 8 повнозв'язних шарів, з використанням ReLU в якості функції активації, та повертає щільність простору σ та вектор ознак (в оригінальній реалізації [20] 256-мірний вектор). Отриманий вектор об'єднується з напрямком огляду камери і далі подається на додатковий повнозв'язний шар (також з ReLU в якості активації), щоб на виході отримати RGB колір залежний від напрямку погляду камери (рис. 2.1).

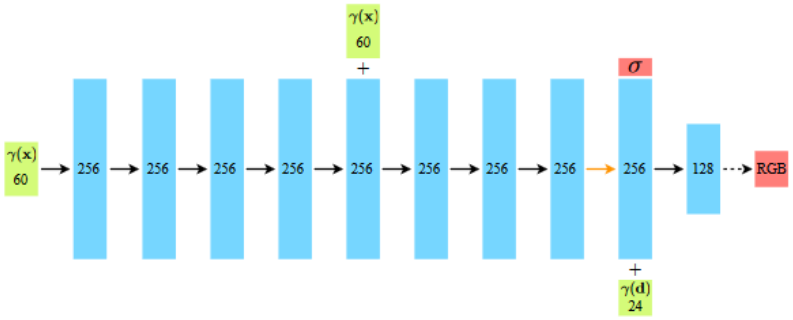

Рисунок 2.1 – Схема БШП в моделі NeRF

Цей механізм дозволяє відтворювати неламбертівські ефекти (такі як дзеркальні відбиття від недзеркальних поверхонь). Такі ефекти продемонстровані на рисунку 2.2. Так під певним кутом на борту корабля з'являється дзеркальне відбиття, розмір, положення та форма якого напряму залежать від напрямку променю від камери до поверхні.

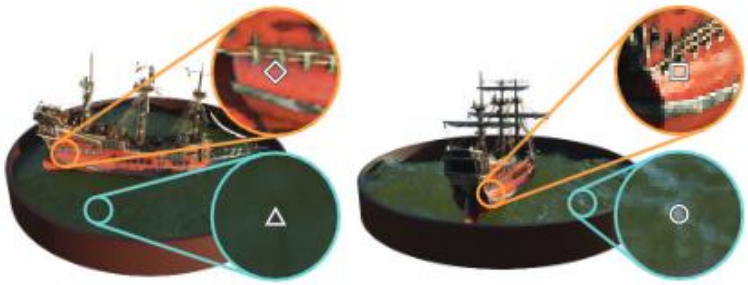

Рисунок 2.2 – Приклад неламбертовських ефектів відтворених за допомогою моделі NeRF

2.2.1 Об’ємний рендеринг в полях випромінювання (Radiance Fields)

Як зазначалося вище, NeRF представляє сцену як об’ємну щільність та спрямоване випромінення в будь-якій точці простору. Для рендеру кольору для будь-якого променю, що проходить через сцену, використовують принципи з класичного об’ємного рендерингу [32].

Об’ємну щільність σ(x) можна інтерпретувати як диференціальну ймовірність того, що промінь закінчиться на нескінченно малій частинці в точці x. Очікуваний колір $C(r)$ променю, що розпочинається в точці положення камери, $r(t) = o + td$ можна визначити за допомогою рівняння (2.1).

$$C(r) = \int_{t_n}^{t_f} T(t)\, \sigma\big(r(t)\big)\, c(r(t), d)\, dt, \tag{2.1}$$

де $C(r)$ – очікуваний колір;

$r(t)$ – точка на промені r на відстані t від його початку (тобто від положення камери) та визначається рівнянням (2.2);

t_n, t_f – відповідно найближча та найдальша точки променю (його межі);

$T(t)$ – акумулюючий коефіцієнт пропускання променю;

$\sigma\big(r(t)\big)$ – об’ємна щільність простору у точці t на промені r;

$c(r(t), d)$ – випромінений колір у точці t на промені r з напрямком огляду d.

Точка на промені вираховується за допомогою простого рівняння 2.2.

$$r(t) = o + td, \tag{2.2}$$

де $r(t)$ – точка на промені r на відстані t;

d – одиничний вектор напряму погляду камери;

o – точка початку променю (положення камери в просторі).

Функція $T(t)$ визначає акумулятивний коефіцієнт пропускання променю в просторі від t_n до t. Іншими словами, він показує ймовірність того, що промінь пройде шлях від t_n до t не зачепивши жодного об’єкту на своєму шляху. Вона обчислюється наступним чином:

$$T(t) = \exp\left(-\int_{t_n}^{t} \sigma\big(r(s)\big)\, ds\right). \tag{2.3}$$

Таким чином відтворення повного зображення потребує обчислення інтегралу $C(r)$ для кожного променю віртуальної камери, пропущеного через кожен піксель зображення.

Для чисельного наближення цього інтегралу використовується квадратура. Детермінована квадратура, яка зазвичай використовується для рендерингу дискретизованих воксельних сіток, ефективно обмежує роздільну здатність кінцевого представлення 3Д об’єкту, оскільки MLP викликається лише у фіксованому дискретному наборі локацій. Щоб вдосконалити цей механізм, використовується підхід стратифікованої вибірки, де розбивається проміжок $[t_n, t_f]$ на N рівномірно розташованих шматочків, а потім витягуємо одну точку рівномірно навмання з кожного з таких шматочків:

$$t_i \sim \mathcal{U}\left[t_n + \frac{i-1}{N}\big(t_f - t_n\big), t_n + \frac{i}{N}\big(t_f - t_n\big)\right]. \tag{2.4}$$

Хоча в такій схемі і використовується дискретний набір точок для оцінки інтеграла, стратифікована вибірка дозволяє нам представити безперервну функцію сцени, тому що ця вибірка призводить до того, що БШП викликається кожного разу на різних, а головне, безперервних точках протягом усієї оптимізації.

Використовуючи отриманий набір вибраних даних і квадратурне правило можна адаптувати рівняння 2.1 для чисельного наближення значення $C(r)$ наступним чином:

$$\hat{C}(r) = \sum_{i=1}^{N} T_i(1 - \exp(-\sigma_i \delta_i))c_i, \tag{2.5}$$

де $\hat{C}(r)$ – чисельне наближення значення $C(r)$;

N – кількість вибраних точок на промені;

σ_i, c_i – прогнозовані значення об'ємної щільності та випроміненого кольору в i-ій точці;

$\delta_i = t_{i+1} - t_i$ – відстань між сусідніми точками.

Відповідно адаптовано і функцію розрахунку акумулятивний коефіцієнт пропускання променю в просторі T_i для i-ої точки:

$$T_i = \exp\left(-\sum_{j=1}^{i-1} \sigma_j \delta_j\right), \tag{2.6}$$

Таким чином, чисельно наближується значення кольору для певного пікселю з дискретного набору точок і зберігається можливість диференціювання функції рендерингу, що критично важливо для оптимізації БШП NeRF. Функція $\hat{C}(r)$ для набору (σ_i, c_i) тривіально диференціюється та зводиться до традиційного альфа-компонування з альфа-значеннями $\alpha_i = 1 - \exp(-\sigma_i \delta_i)$.

2.2.2 Позиційне кодування вхідних даних

Спираючись на той факт, що глибокі штучні нейронні мережі більш схильні до навчання функцій нижчої частоти [33]. В зазначеній роботі також показано, що відображення вхідних даних в простір більшої розмірності використовуючи високочастотні функції дозволяє краще вивчити дані, які містять високочастотні коливання.

Адаптував описані вище дослідження до задачі 3Д реконструкції сцени, переформулюємо математичне визначення F_θ як композицію двох функцій $F_\theta = F'_\theta \circ \gamma$, перший з яких навчається, а другий – ні. Ця зміна значно покращило результати (рис. 2.3).

У новій нотації функція γ – це відображення даних з $\mathbb{R}$ в простір більшої розмірності $\mathbb{R}^{2L}$, а функція F'_θ – звичайний багатошаровий персептрон. Формально позиційне кодування формулюється так:

$$\gamma(p) = (\sin(2^0\pi p), \cos(2^0\pi p), \dots, \sin(2^{L-1}\pi p), \cos(2^{L-1}\pi p)). \qquad (2.7)$$

Рисунок 2.3 – Візуалізація рузільтатів відновлення 3Д сцени. Ліве зображення – справжній вигляд об’єкту, середнє – результат NeRF з позиційним кодуванням, праве – NeRF без позиційного кодування.

Функція позиційного кодуівання γ використовується окремо для

кожної з трьох значень координат x, які в свою чергу нормалізовані в діапазон $[-1,1]$, та окремо до трьох компонент декартового вектору напрямку погляду d, які за своєю природою знаходяться в діапазоні $[-1,1]$.

В оригінальній статті NeRF використовують значення L=10 для $\gamma(\mathrm{x})$ і L=4 для $\gamma(d)$.

2.2.3 Ієрархічний відбір об’єму

Описаний вище варіант вибору точок на промені є доволі неефективним: вільний простір та закриті об’єктом області, які не впливають на зображення, все одно багаторазово відбираються в процесі рендерингу. Натомість було запропоновано ієрархічну модель відбору точок.

Замість того, щоб використовувати одну нейрону мережу, NeRF під час навчання одночасно оптимізує два БШП: одну «грубу» і одну «фінальну». Спочатку ми відбираємо N_c точок методом стратифікованої вибірки, описану рівнянням (2.4), і подаємо на вхід «грубої» мережі. На основі отриманих результатів з мережі, створюється більш обґрунтована вибірка точок вздовж кожного променю, де місця відбору зміщені більше до відповідних частин об’єму.

Щоб реалізувати такий механізм, треба переписати рівняння (2.5) у вигляді зваженої суми точок з «грубої» мережі $\hat{C}_c(r)$:

$$\hat{C}_c(r) = \sum_{i=1}^{N_c} w_i\, c_i, \qquad w_i = T_i(1 - \exp(-\sigma_i \delta_i)), \tag{2.8}$$

де N_c – кількість точок в наборі для «грубої мережі».

Нормалізував ваги $\widehat{w}_i = w_i / \sum_{i=1}^{N_c} w_i$ дає кусково-постійну функцію щільності ймовірності вздовж променю. Далі вибирається другий набір даних з N_f точок з отриманого розподілу методом оберненого перетворення.

Отримані дані об’єднується з набором для «грубої» мережі і поганяється через «фінальну» мережу і обчислюється фінальний рендеринг кольору для променю $\hat{C}_f(r)$ на базі двох сетів даних $N_f + N_c$.

Ця процедура розподіляє більше зразків у регіонах, які, як очікується, містять видимий об’єкт. Вона вирішує ту саму задачу, що й вибірка по значущості, але використовує вибіркові значення як нерівномірну дискретизацію всієї області інтегрування, а не розглядає кожну вибірку як незалежну ймовірнісну оцінку всього інтеграла.

2.2.4 Процес оптимізації

Для кожної конкретної 3Д сцени чи об’єкту оптимізується окрема мережа. Для цього процесу потребується RGB зображення об’єкту чи сцени, відповідні їм точні пози камери, внутрішні параметри камери (такі як фокальна відстань та принципові точки) та межі 3Д сцени.

На кожній ітерації оптимізації довільно обирається пакет променів з набору всіх пікселів з усіх зображень. Далі методом описаним в попередньому підрозділі обираються точки на променях для «грубої» та «фінальної» мережі. Далі прогнозуються випромінений колір та об’ємна щільність для кожної точки на промені та рендериться колір для кожного променю. Функція втрат обчислюється як середньоквадратичне відхилення між відрендереним та реальним пікселя:

$$\mathcal{L} = \sum_{r \in \mathcal{R}} \left[\left\| \hat{C}_c(r) - C(r) \right\|_2^2 + \left\| \hat{C}_f(r) - C(r) \right\|_2^2 \right], \tag{2.9}$$

де $\mathcal{R}$ – це сет променів в пакеті;

$\hat{C}_c(r)$ – прогнозований колір «грубою» мережею;

$\hat{C}_f(r)$ – прогнозований колір «фінальною» мережею;

$C(r)$ – реальний колір пікселю.

Варто зауважити, що незважаючи на те, що фінальний колір визначається рендерингом з «фінальної» мережі $\hat{C}_f(r)$, все одно мінімізуються втрати $\hat{C}_c(r)$ так, щоб розподіл ваг від «грубої» мережі можна було використовувати для розподілу зразків в «фінальній» мережі.

2.3 Опис моделі Bundle Adjusting Neural Radiance Fields

Як було описано в підрозділі 1.2.2, Bundle Adjusting Neural Radiance Fields (далі BARF) знімає обмеження з архітектури NeRF на високу точність або навіть наявність поз камери для навчання. Це розширює горизонти використання цієї технології та спрощує процес збору та підготовки даних.

2.3.1 Математичний визначення цільової функції BARF

Як і у класичному NeRF для реконструкції сцени використовується БШП для прогнозування випроміненого кольору і об’ємної щільності в точці на промені випущеного з віртуальної камери, що відповідає положенню кожного окремого пікселя. Основною відмінністю від класичної архітектури є те, що в явному вигляді на мережу не подається напрям погляду камери, так як вважається, що він невідомий. Тому представлення 3Д сцени формалізується за допомогою БШП таким чином $f: \mathbb{R}^3 \to \mathbb{R}^4$, де вихід моделі це випромінений колір $c \in \mathbb{R}^3$ і об’ємна щільність $\sigma \in \mathbb{R}$, а на вхід подається 3Д координата точки $\mathrm{x} \in \mathbb{R}^3$. Таким чином, якщо вихід моделі записати як $y = [c, \sigma\,]^T$, представлення сцени виглядає наступним чином:

$$y = f(\mathrm{x}, \theta), \tag{2.10}$$

де θ – це параметри мережі, які тренуються.

Щоб отримати 3Д координати x точки на промені, або в загалом отримати промінь, потрібно перейти від простору камери до світових координат. Для цього визначимо функцію (2.10) відносно простору камери. Нехай $u \in \mathbb{R}^2$ координати пікселя на зображені, а $\bar{u} = [u; 1]^T \in \mathbb{R}^3$ однорідні координати цього пікселя. Звідси можна визначити 3Д точку на промені на глибині z_i від його початку як $t_i = z_i \bar{u}$.

Маючи визначення 3Д точки, можна адаптувати рівняння (2.1) під координати простору камери наступним чином:

$$\hat{\mathrm{C}}(u) = \int_{z_n}^{z_f} T(u, z)\, \sigma(z\bar{u})\, c(z\bar{u})\, dz, \tag{2.11}$$

де $\hat{\mathrm{C}}(u)$ – RGB колір пікселю u;

z_n, z_f – відповідно найменша та найбільша глибина на промені;

$T(u, z)$ – акумулюючий коефіцієнт пропускання променю для пікселя u, який розраховується наступним чином:

$$T(u, z) = exp\left(-\int_{z_n}^{z} \sigma(s\bar{u}) ds\right). \tag{2.12}$$

Як було вже описано в підрозділі 2.2.2, процес рендерингу на практиці обчислюється за допомогою квадратури на дискретному наборі точок N з глибинами $\{z_1, \dots, z_N\}$ вибраними на промені. Звідси можна констатувати факт, що мережа f викликається N разів, а її виходи можна визначити як сет такого виду $\{y_1, \dots, y_N\}$, які потім будуть скомпоновані механізмом рендерингу (описаним в рівняннях (2.11) і (2.12)). Для простоти запису механізм рендерингу можна сформулювати як детерміновано диференційовану функцію наступного виду $g: \mathbb{R}^{4N} \to \mathbb{R}^3$ та записати $\hat{\mathrm{C}}(u)$ як $\hat{\mathrm{C}}(u) = g(y_1, \dots, y_N)$.

Для остаточного переходу з простору камери в світовий простір необхідно додати в відповідну параметризовану функцію перетворення. Для правильного відображення точки з простору камери в світовий використовують позу камери зі шістьма ступенями свободи (6-DoF) $p \in \mathbb{R}^6$ та Евклідові перетворення (ізометрію) вигляду $\mathcal{W}$: $\mathbb{R}^3 \to \mathbb{R}^3$. Отже функція рендерингу для кольору точки в світовій системі координат матиме вигляд:

$$\hat{\mathrm{C}}(u,p) = g\big(f(\mathcal{W}(z_1\bar{u},p),\theta),\dots,f(\mathcal{W}(z_N\bar{u},p),\theta)\big). \tag{2.13}$$

Якщо позначити кількість зображень для тренування моделі BARF за M і представити зображення як функцію від координат пікселя, яка повертає його значення (інтенсивність по кожному каналу), то даними для задачі оптимізації цієї моделі є сет всіх зображень $\{C_i\}_{i=1}^{M}$ та сет поз камери цих зображень $\{p_i\}_{i=1}^{M}$. А сама задача оптимізації приймає наступний вигляд:

$$\min_{p_1,\dots p_M,\theta} \sum_{i=1}^{M}\sum_{u} \left\|\hat{\mathrm{C}}(u,p_i,\theta) - C_i(u)\right\|_2^2, \tag{2.14}$$

де $\hat{\mathrm{C}}(u,p_i,\theta)$ – відрендерений колір пікселя прогнозований мережею з параметрами θ.

Отже, як можна бачити з виду цільової функції (формула (2.14)), корегування або визначення пози камери відбувається методом градієнтної оптимізації на основі похибки всіх пікселів зображення. Подібна схема використовується в класичному алгоритмі корегуванні променів (bundle adjustment, далі BA) в області комп’ютерного зору та просторового сприйняття.

2.3.2 Позиційне кодування та його маскування

Ще однією відмінністю BARF від NeRF є модифікація позиційного кодування вхідних даних мережі. BARF представляє перетворення 3Д координат в простір більшої розмірності як функцію $\gamma: \mathbb{R}^3 \to \mathbb{R}^{3+6L}$ та визначає її наступним чином:

$$\gamma(\mathrm{x}) = [\mathrm{x}, \gamma_0(\mathrm{x}), \gamma_1(\mathrm{x}), \dots, \gamma_{L-1}(\mathrm{x})] \in \mathbb{R}^{3+6L}, \tag{2.15}$$

де $\gamma_k(\mathrm{x}) = [\cos(2^k \pi \mathrm{x}), \sin(2^k \pi \mathrm{x})]$ – кодування вхідного значення k-ою частотою.

Але ключовою ідеєю всього Bundle Adjusting Neural Radiance Fields є використання нечіткого маскування закодованих вхідних даних. Цей механізм дає можливість відключати певні частоти в певний момент часу тренування для того, щоб дати можливість оптимізатору краще знайти позу камери.

В оригінальній статті [1], автори називають цю стратегію «реєстрація від грубого до точного». Основна її ідея в тому, що ми на початку навчання за допомогою нечіткого маскування пропускаємо в мережу лише низькочастотні сигнали, що дає можливість паралельно навчатися мережі дуже грубим деталям сцени (образи, розподіл кольорів, тощо) і в той же час оптимізувати (знаходити) більш точні пози камери. В процесі навчання додаються все більше частот і таким чином під кінець оптимізації мережа приймає весь діапазон позиційного кодування. Ця стратегія змушує мережу вивчати спочатку лише грубі образи 3Д сцени, поступово додаючи все більше і більше деталей для реєстрації.

В основному це досягається за допомогою зважування кожної частотної компоненти позиційного кодування:

$$\gamma_k(\mathrm{x}, \alpha) = w_k(\alpha) * [\cos(2^k \pi \mathrm{x}), \sin(2^k \pi \mathrm{x})], \quad (2.16)$$

де α – параметр пропорційний до прогресу оптимізації і $\alpha \in [0, L]$;

w_k – залежний від α нечіткий ваговий коефіцієнт частоти, що рахується за наступною формулою:

$$w_k(\alpha) = \begin{cases} 0 & \text{при } \alpha < k \\ \dfrac{1 - \cos\big((\alpha - k)\pi\big)}{2} & \text{при } 0 \le \alpha - k < 1 \\ 1 & \text{при } \alpha - k \ge 1 \end{cases} \quad (2.17)$$

При $\alpha = 0$ на мережу подається не закодована 3Д координати точки. Це дозволяє більш якісніше оптимізувати цільову функцію за рахунок того, що початкове наближення відбувається на простих (згладжених) даних низької частоти.

Таким чином модель BARF дозволяє не лише реконструювати 3Д об’єкт по сету 2Д зображень, а й уточнювати, або навіть знаходити, пози камер паралельно з навчанням БШП.

2.4 Опис моделі поточкових Neural Radiance Fields

Основною проблемою моделей описаних в підрозділах 2.2 і 2.3 є їхня неможливість відтворювати великі 3Д сцени або 3Д об’єкти зі збереженням високої деталізації. Це обумовлено тим, що NeRF (і відповідно BARF) кодують 3Д простір в середині ваг персептрону і, очевидно, через обмежену кількість ваг моделі (а БШП використовується не глибокий) кількість потенційно закодованої інформації обмежений та невеликий.

Основною ідеєю поточкових NeRF (далі PNeRF) є відокремлення місця зберігання даних від механізму їх відтворення (реконструкції). Таким чином, маючи окремо сховище попередньо оброблених даних в певній

структурі і інструменту реконструкції зображення з цього сховища, нівелюється обмеження на розміри 3Д сцен. Також перевагою такого підходу є те, що інструмент реконструкцій, роль якого виконує ансамбль персептронів, можна попередньо навчити на наборі різних 3Д об’єктів та сцен і донавчати конкретній цільовій сцені.

2.4.1 Нейронна хмарина точок (Neural Point Cloud)

Нейронна хмарина точок – це хмарина точок, в класичному її понятті, але кожна її частина являє собою вектор ознак підготовлений 2Д згортковим енкодером.

Для створення нейронної хмарини точок спочатку треба сформувати з поданого набору зображень звичайну хмарину точок, і кожному її елементу зіставити вектор ознак.

Розглянемо 2 варіанти формування звичайної хмарини точок: генерація її за допомогою 3Д згорткових мереж багатовидової стереосистеми (MVSNet подібні мережі) та зняті з пристрою значення глибини (RGB-D зображення).

Нехай ми маємо набір зображень якогось 3Д об’єкта чи сцени $\{C_i\}_{i=1}^{M}$ з параметрами камери $\{\Phi_i\}_{i=1}^{M}$. MVSNet на основі цих даних будує плоску об’єму вартість поданих зображень шляхом викривлення ознак 2Д зображень опираючись на сусідні картинки з набору, тим самим симулюючи стереопари, а потім на основі отриманих даних регресією прогнозує об’єм ймовірності глибини використовуючи глибокі 3Д згорткові мережі. Фінальні карти глибин обчислюються шляхом лінійного комбінування значень глибина в кожній площині, зважених на їх ймовірності. Іншими словами мережа знаходить ймовірні 3Д площини різної глибини в наборі зображень і прогнозує ймовірність кожного пікселя належати той чи іншій площині.

Таким чином роботу мережі можна описати наступним виразом:

$$\{\mathrm{x}_i, v_i\} = G_{\mathrm{x},v}(C_1, \Phi_1, \dots, C_M, \Phi_M), \tag{2.18}$$

де v_i – впевненість правильності місцезнаходження точки x_i;

$G_{\mathrm{x},v}$ – згорткова мережа генерації карт глибин.

Другий варіант це пост обробка знятої з пристрою карт глибини (мається на увазі пристрої типу LiDAR). Цей підхід обчислювально простіший, але потребує спеціального пристрою. Основна ідея, це максимально позбутися викидів та дірок в даних. Для цього використовують примітивні фільтри, основані на кластеризації, для визначення викидів та алгоритм реконструкції площин Пуассона для заповнення дірок в поверхнях. Так як для подальшої роботи системи необхідні коефіцієнти впевненості на кожну 3Д точку, для цього варіанту вони встановлюються константою ($v_i = 0.3$ з можливістю корегування цього значення, яка буде описано далі).

Другим етапом формування нейронної хмарини точок є генерація ознак із зображень. Для цього використовуються згорткові енкодери сімейства VGG. Результатом цього етапу є сет ознак для кожного зображення наступного вигляду:

$$\{f_i\} = G_f(C_j), \tag{2.19}$$

де f_i – вектор ознак;

G_f – згортковий енкодер сімейства VGG.

В архітектурі PNeRF використовується VGG з трьома блоками шарів, а вихідні ознаки формуються методом конкатенації ознак з різних рівнів таким чином щоб на виході отримати вектор довжиною 56 (8, 16 та 32 ознаки з кожного рівня енкодера).

Маючи карти глибин, вхідні RGB зображення та вектори ознак для кожного зображення легко будується нейронна хмарина точок за допомогою проекції в 3Д простір.

2.4.2 Процес поточкового рендерингу

Основною ідеєю PNeRF є орієнтація механізму рендерингу на роботу з ознаками нейронної хмарини точок. Позначимо будь-яку нейрону хмарину точок як:

$$P = \{(\mathscr{p}_i, f_i, \upsilon_i) | i = 1, \dots, N\}, \tag{2.20}$$

де P – нейрона хмарина точок;

$\mathscr{p}_i$ – 3Д координата точки у світовій системі координат;

f_i – вектор ознак i-ої точки;

υ_i – впевненість у розташуванні i-ої точки.

Як і у звичайному NeRF (описаному в підрозділі 2.2.1) ми пускаємо промінь від віртуальної камери, що знаходиться в координатах пікселю зображення, яке необхідно реконструювати, в напрямку огляду камери і вибираємо на промені сет точок, на яких буде відбуватися процес рендерингу. Відмінність починається в момент, коли треба прогнозувати об’ємну щільність та випромінений колір. Для цього для 3Д точки на промені x_i PNeRF знаходить K найближчих точок-сусідів з P в середині радіусу R.

Обрані точки-сусіди спочатку оброблюються мережею F (не глибоким БШП) для того, щоб адаптувати вектор ознаки та визначити його вплив відносно точки на промені. Іншими словами, ця мережа – функція переходу вектора ознак від опису відносно впливу точки на деякий простір навколо до опису його впливу на конкретну наперед задану точку. Для кращого

узагальнюючого потенціалу цієї мережі та інваріантності до абсолютного значення положення вхідних точок, на мережу подається позиція нейронної точки відносно точки на промені. Тоді можна записати цю функцію як:

$$f_{i,\,\mathrm{x}} = F(f_i, \mathrm{x} - \mathscr{p}_i), \tag{2.21}$$

де $f_{i,\,\mathrm{x}}$ – вектор ознак f_i відносно точки x.

Для подальшого прогнозування кольору точки, отриманий набір відносних векторів ознак зважується та агрегується за наступною формулою:

$$f_{\mathrm{x}} = \sum_{i=1}^{K} \upsilon_i \frac{\mathscr{w}_i}{\sum \mathscr{w}_i} f_{i,\,\mathrm{x}}, \tag{2.22}$$

де υ_i – впевненість у розташуванні i-ої точки;

$\mathscr{w}_i$ – обернено пропорційна вага до дистанції між нейронною точкою та точкою на промені, $\mathscr{w}_i = \frac{1}{\|\mathscr{p}_i - \mathrm{x}\|}$.

Отриманий вектор ознак f_{x} подається на окрему мережу R для прогнозування випроміненого кольору в цьому місці простору. Так як результат залежить від точки погляду, на вхід в мережу подається декартовий одиничний вектор направлення променю. Отримаємо наступне рівняння:

$$c = R(f_{\mathrm{x}}, d). \tag{2.23}$$

Для обчислення випроміненого кольору використовується коефіцієнт впевненості υ_i, який в подальшому буде оптимізуватися (модифікуватися) під час навчання для надання мережі більшої гнучкості при відкиданні непотрібних або неточно розташованих точок.

Для обчислення об’ємної щільності в точці x застосовується подібна агрегація точок-сусідів. Однак в цьому випадку спочатку прогнозується об’ємна щільність для кожного відносного вектору ознак $f_{i,\,\mathrm{x}}$ за допомогою ще одної мережі T (не глибокого БШП), а вже потім зважується на обернено пропорційні до відстані коефіцієнти:

$$\sigma_i = T\left(f_{i,\,\mathrm{x}}\right). \tag{2.24}$$

$$\sigma = \sum_{i=1}^{K} \sigma_i v_i \frac{w_i}{\sum w_i}, \tag{2.25}$$

де w_i – обернено пропорційна вага до дистанції між нейронною точкою та точкою на промені, $w_i = \frac{1}{\|p_i - \mathrm{x}\|}$;

v_i – впевненість у розташуванні i-ої точки-сусіда.

Сам процес рендерингу кінцевого кольору відбувається аналогічно тому, що був описаний в підрозділі 2.2.3 згідно формули (2.8). Повний потік даних на його обробка представлені на рисунку 2.4.

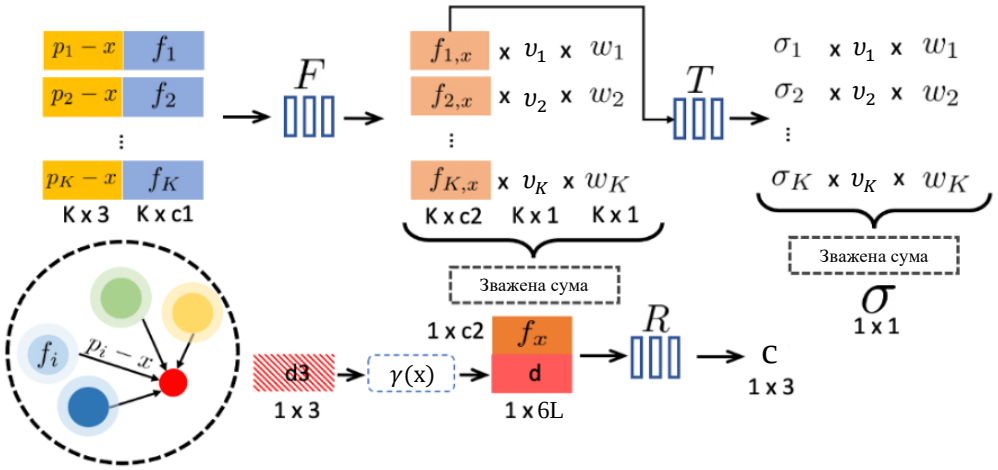

Рисунок 2.4 – Схема прогнозування випроміненого кольору і об’ємної щільності в моделі PNeRF

2.4.3 Процес оптимізації в PNeRF

Вище було описано механізм створення та зберігання даний про 3Д сцену, а також інструмент маніпулювання над цими даними (а саме генерація нових видів та реконструкція сцени). Так як увесь процес рендерингу диференційований, то в процесі навчання система може також правити дані (оптимізувати вектори ознак f_i та значення коефіцієнтів впевненості положення точки v_i). Через те, що початково згенерована нейронна хмарина точок може містити викиди і дірки, тим самим погіршуючи якість реконструкції, в PNeRF реалізовано процес «обрізки» та «нарощування» точок.

Процес «обрізки» являє собою періодичне видалення точок з коефіцієнтом $v_i < 0.1$ раз в 10 тисяч ітерацій оптимізації. Це мотивовано тим, що цей коефіцієнт показую ймовірність того, що тичка належить площині сцени (або наскільки достовірне положення цієї точки в 3Д просторі), а отже при дуже низькому значені ця точка точно позитивно не буде впливати на прогнозування.

Якщо використовується варіант побудови хмарини точок за допомогою багатовидової стереосистеми типу MVSNet, то в процесі оптимізації градієнт може правити і цю мережу також для генерації в подальшому більш зручних для рендерингу ознак і більш точних коефіцієнтів впевненості. Для цього в функцію втрат додається регуляризуючий компонент $\mathcal{L}_{\text{sparse}}$, який рахується за такою формулою:

$$\mathcal{L}_{\text{sparse}} = \frac{1}{|v|}\sum_{v_i}[\log(v_i) + \log(1 - v_i)], \tag{2.26}$$

де $|v|$ – розмір пакету тренувального сету.

Результати процесу «обрізки» та його вплив на рендеринг зображень

показано на рисунку 2.5. На верхній парі картинок показано результат реконструкції зображення до видалення точок, на нижній – після.

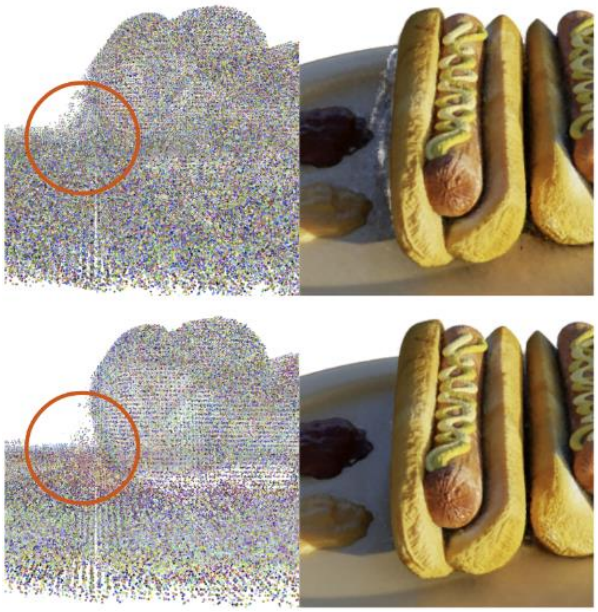

Рисунок 2.5 – Результати процесу «обрізки» нейронної хмарини точок

На відміну від процесу «обрізки», «нарощування» точок є більш складним механізмом, так як перша стратегія лише знищує непідходящі точки, тому як друга – повинна знаходити місця недостачі дочок (так звані дірки) та заповнювати їх осмисленою інформацією, яка повинна покращити результати реконструкції в майбутньому.

Для реалізації цієї стратегії, PNeRF визначає найбільш непрозору точку під час вибору точок на промені (процес описаний в розділі 2.2.3). Для цього, з рівняння (2.8) вичленимо формулу розрахунку прозорості:

$$\alpha_j = 1 - \exp\left(-\sigma_j \delta_j\right), \qquad j_g = \underset{j}{\operatorname{argmax}}\, \alpha_j, \tag{2.27}$$

де δ_j – відстань між сусідніми точками на промені;

α_j – прозорість j-ої точки.

Отримавши координати найнепрозорішої точки на промені t_{j_g} обчислюється відстань ϵ_{j_g} до найближчої нейронної точки в просторі. Маючи ці дані, алгоритм просто порівнює значення прозорості з наперед заданим мінімальним значенням ($\alpha_{j_g} > T_{\text{opacity}}$) та значення відстані з наперед заданим мінімальним значенням ($\epsilon_{j_g} > T_{\text{dist}}$). Якщо ці умови виконуються, то ми додаємо t_{j_g} до нейронної хмарини точок.

Таким чином система може в процесі навчання сама доповнювати погано згенеровану нейронну хмарину точок новими даними. Результати алгоритму «нарощування» приведені на рисунку 2.6. На верхній парі картинок показано результат реконструкції зображення до додавання точок, на нижній – після.

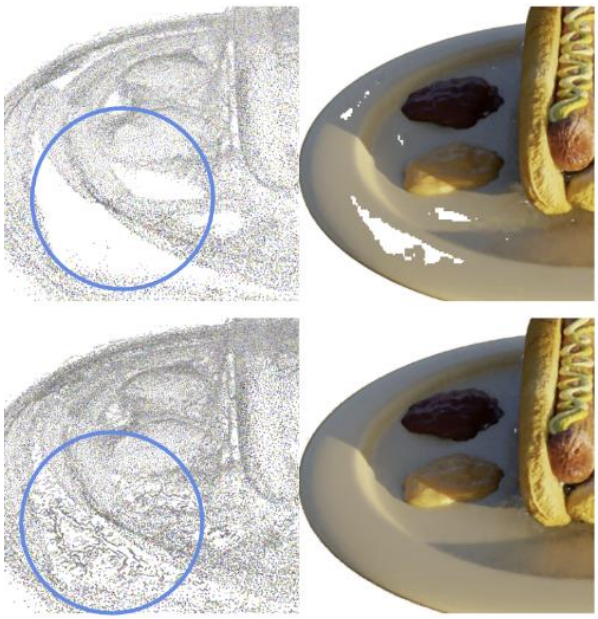

Рисунок 2.6 – Результати процесу «нарощування» нейронної хмарини точок

Для PNeRF фінальною функцією втрат слугує не тільки описана в підрозділі 2.2.4 фотометрична функція втрат (формула (2.9)), а й додаткова регуляризація для корекції коефіцієнту впевненості $\mathcal{L}_{\text{sparse}}$. Загальна функція

втрат визначається наступним чином:

$$\mathcal{L}_{\text{opt}} = \mathcal{L}_{\text{render}} + a\mathcal{L}_{\text{sparse}}, \quad (2.28)$$

де $\mathcal{L}_{\text{render}}$ – фотометрична функція втрат;

$\mathcal{L}_{\text{sparse}}$ – регуляризуючий компонент для корекції коефіцієнту впевненості;

a – балансуючий коефіцієнт (зазвичай не більше $2e^{-3}$).

А загальний процес оптимізації доповнений стратегіями покращення (уточнення) структури даних: «обрізка» та «нарощування» нейронних точок.

2.5 Опис модифікації моделі поточкової Neural Radiance Fields для невідомих поз камери

Основною ціллю цього дослідження є формулювання архітектури мережі сімейства NeRF, яка якісно працює з великими 3Д сценами і має більший ступінь свободи в плані визначення поз камери (тобто зашумлене або невідоме значення). Як основу для проведення експериментів було обрано поточкову NeRF, так як її механізми (описані в підрозділі 2.4) дозволяють реконструювати великі 3Д сцени з високою точністю.

Для реалізації стратегії реєстрації поз камери під час навчання NeRF було обрано механізми BARF (описані в підрозділі 2.3). Вони просту та ефективну реалізацію нейронного корегування променів та визначення поз камери.

Основними гіпотезами щодо архітектури і стратегії навчання, які були сформульовані під час цього дослідження, є реалізація нечіткого маскування позиційного кодування вектору напрямку погляду променю, що забезпечить реалізацію стратегії «реєстрація від грубого до точного», уточнення задачі оптимізації, використання двох незалежних оптимізаторів, для оптимізації

персептронів та поз, і стратегія «розігріву» – алгоритм налаштування менш агресивного коефіцієнта швидкості навчання на початкових ітераціях.

2.5.1 Адаптація стратегії «від грубого до точного» для PNeRF

Як зазначалося в підрозділі 2.4.2 PNeRF для прогнозування випроміненого кольору використовує БШП, на вхід якого подається зважена сума векторів відносних ознак і позиційно закодований вектор напрямку погляду віртуальної камери (або напрям променю). Так як напрям погляду d визначається за допомогою матриці Евклідових перетворень однорідної координати позиції пікселя через позу камери (формула (2.13)). Звідси можна сформулювати визначення цього вектора:

$$d_{ij} = \mathcal{W}(t_i, p_j), \qquad \hat{d}_j = \frac{d_{ij}}{|d_{ij}|}, \tag{2.29}$$

де d_{ij} – вектор від початку i-го променю на j-ій камері до точки t_i;

p_j – матриця пози j-ої камери;

$\hat{d}_{ij}$ – одиничний вектор напряму променю;

$|d_{ij}|$ – норма вектору d_{ij}.

Модифікуємо позиційне кодування для PNeRF наступним чином:

$$\gamma(\hat{d}_{ij}) = [\hat{d}_{ij}, \gamma_0(\hat{d}_{ij}), \gamma_1(\hat{d}_{ij}), \dots, \gamma_{L-1}(\hat{d}_{ij})] \in \mathbb{R}^{3+6L}, \tag{2.30}$$

де $\gamma_k(\mathrm{x}) = [\cos(2^k \pi \mathrm{x}), \sin(2^k \pi \mathrm{x})]$ – кодування вхідного значення k-ою частотою.

Додавання не закодованого значення напрямку променю обумовлене тим, що на перших ітераціях оптимізації оптимізатор буде знаходити грубе наближення пози камери опираючись лише на сирий вектор напрямку.

Також це не буде перенавантажувати оптимізацію мережі R.

Механізм маскування повністю відповідає тому, що описаний формулами (2.16) і (2.17). Єдиною відмінністю є значення коефіцієнту α, діапазон якого навмисно зменшено до $\alpha \in [1, L]$ для того, щоб швидше та точніше провести глобальну реєстрацію поз камери (грубу оцінку початкового наближення). Схема системи рендерингу представлена на рисунку 2.7.

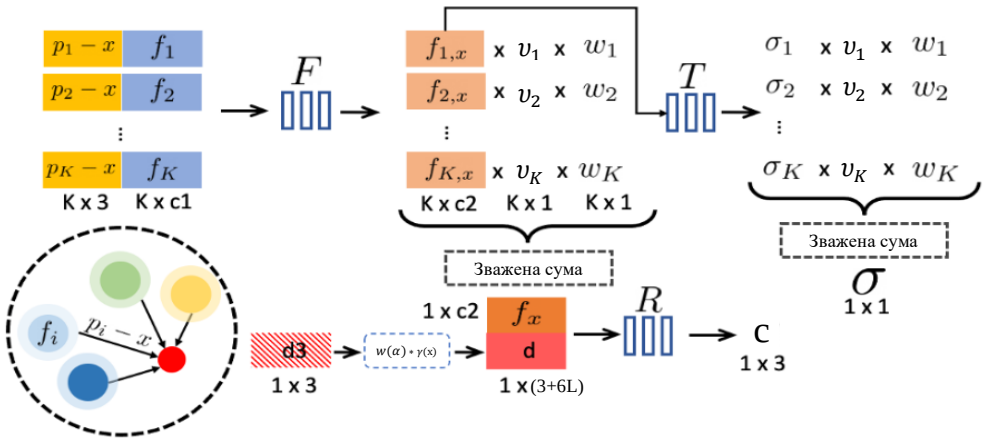

Рисунок 2.7 – Схема прогнозування випроміненого кольору і об’ємної щільності в модифікованій моделі PNeRF

2.5.2 Постановка задачі та організація процесу оптимізації

Адаптуємо задачу процесу оптимізації системи PNeRF під описану раніше проблему:

$$\min_{p_1,\dots p_M,\theta_F,\theta_T,\theta_R} \sum_{i=1}^{M} \sum_{u} \|PNeRF(u, p_i, \theta_F, \theta_T, \theta_R, P) - C_i(u)\|_2^2, \tag{2.31}$$

де $PNeRF(\cdot)$ – це функція рендерингу кольору пікселя описана в підрозділі 2.4.2;

u – координати пікселя;

p_i – поза i-ої камери;

$\theta_F, \theta_T, \theta_R$ – ваги мереж F, T і R відповідно;

P – нейронна хмарина точок;

M – кількість зображень в тренувальному наборі;

$C_i(u)$ – реальний колір пікселя u на i-ому зображенні.

Практично ми мінімізуємо фотометричну функцію втрат – квадрат різниці між реальним кольором пікселя та тим, що відрендерила система – залежно від ваг усіх персептронів та поз камер.

Для більш якісної реєстрації поз камер було прийнято рішення використовувати два окремих оптимізатори: один – для оптимізації всіх персептронів, другий – лише для нейронного корегування променів. Це дає можливість більш точно налаштувати параметри оптимізації для кожної задачі та використовувати специфічні методи для кожному оптимізатора окремо.

Для більш стабільної оптимізації поз камер в процес навчання було інтегровано стратегію «константного розігріву» [34]. Її ідея полягає в тому, що на перших ітераціях коефіцієнт швидкості навчання (μ) не змінюється і дорівнює певній константі. Кількість ітерацій «розігріву» і значення коефіцієнту є гіперпараметрами та налаштовуються користувачем.

Для подальших експериментів було обрано в якості тривалості цієї стратегії було обрано 10 тисяч ітерацій. Під час роботи розігріву жодна інша стратегія PNeRF не застосовується. Так, наприклад, алгоритми «обрізки» та «нарощування» починають виконуватися лише після 20-титисячної ітерації.

3 ОПИС ПРАКТИЧНИХ ЕКСПЕРИМЕНТІВ

3.1 Опис наборів даних

3.1.1 Набір даних ScanNet

Відповідно до поставленої задачі, основним набором даних було обрано ScanNet. Це набір відеоданих, знятих в середині приміщень, у форматі RGB-D (кольорове зображення і відповідна йому карта глибин), що містить 2.5 мільйона видів (зображень) у понад 1500 сценах. Ці дані анотовані 3Д позами камер, реконструкціями поверхонь (меш збережений у форматі PLY), та семантичної сегментації на рівні екземплярів (рис. 3.1).

Рисунок 3.1 – Приклад сцен набору даних ScanNet

Сканування охоплює широкий спектр приміщень, включаючи квартири, офіси, готелі та школи. Кожне сканування представлено у вигляді хмари точок, де кожна точка містить інформацію про RGB та глибину. Крім того, набір даних також включає семантичні мітки сегментації для кожної точки, які вказують, чи належить вона до підлоги, стелі, стіни, дверей, вікон або іншої категорії об'єктів.

Набір даних ScanNet використовувався в різних завданнях комп'ютерного зору і машинного навчання, включаючи розуміння сцени, виявлення об'єктів і 3Д реконструкцію. Набір даних також використовувався як еталон для оцінки продуктивності різних алгоритмів і моделей.

Для експериментів було обрано 2 сцени: 0005_00 (приклад сцена на рисунку 3.2) і 0010_00 (приклад сцени на рисунку 3.3). Ці сцени представляють собою офісні приміщення різних розмірів. Як можна побачити з другої картинки на кожному рисунку, ці сцени мають розмиття спричинене рухом камери.

Рисунок 3.2 – Приклад зі сцени 0005_00 набору даних ScanNet

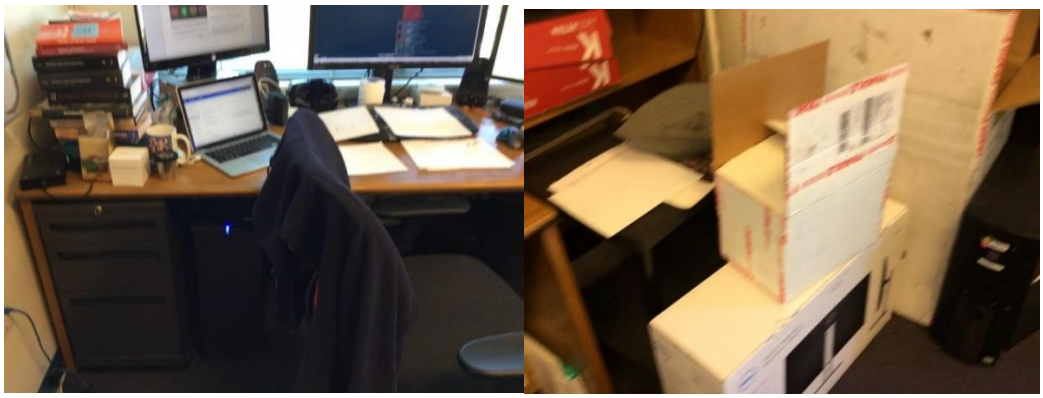

Рисунок 3.3 – Приклад сцени 0010_00 набору даних ScanNet

3.1.2 Синтетичний набір даних NeRF та LLFF

Також для перевірки коректності роботи всіх систем буде використовуватися класичний об’єкт (рис. 3.4) з синтетичного набору даних представленого в роботі [20] та декілька сцен з набору даних Local Light Field Fusion (далі LLFF).

Рисунок 3.4 – Синтетичний 3Д об’єкт (Lego бульдозер) з набору даних NeRF

Синтетичний набір даних NeRF – це сконструйовані за допомогою CAD моделей об’єкти та відрендерені за допомогою фізичного рушія

рендерингу сети зображень цих моделей з різних ракурсів (поз камери). Саме об’єкт Lego бульдозера став класичним і використовується для порівняння якості реконструкції для всіх архітектур сімейства NeRF.

LLFF – це масштабний набір даних реальних сцен, знятих за допомогою висококласного обладнання. Набір даних був створений командою дослідників з Каліфорнійського університету в Берклі та Google Research.

Набір даних LLFF включає 31 сцену з різних як приміщень так і зовнішніх середовищ, таких як парк, музей, кухня і вітальня. Кожна сцена знята з декількох ракурсів за допомогою камерної установки, яка дозволяє фіксувати інформацію про колір і глибину, а також інформацію про напрямок світлових променів.

Набір даних LLFF також включає реальні положення камер і карти глибини для кожної сцени, що дозволяє точно відтворити геометрію сцени в 3Д. Крім того, набір даних містить RGB зображення з високою роздільною здатністю і дані про світлове поле, які можна використовувати для різноманітних завдань комп'ютерного зору і машинного навчання.

Набір даних був використаний як еталон для оцінки продуктивності різних алгоритмів і моделей у завданнях синтезу зображень, екстраполяції нових зображень і 3Д реконструкцію. Приклади сцен з нього наведені на рисунку 3.5.

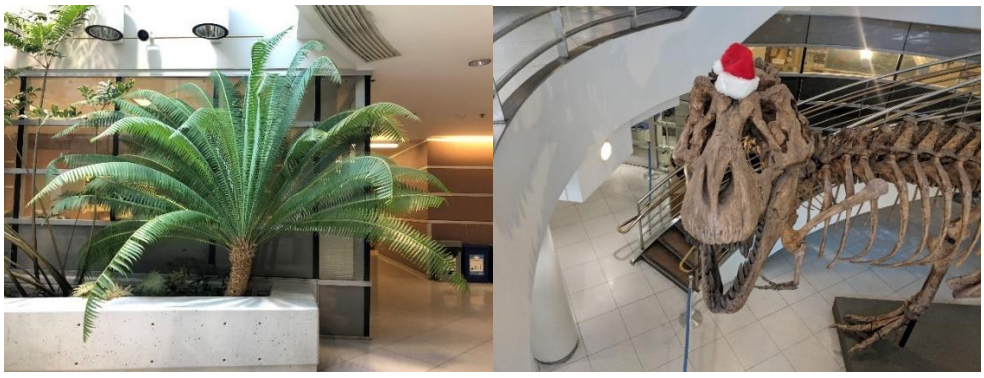

Рисунок 3.5 – Приклад сцен з набору даних LLFF

3.2 Метрики для порівняння моделей

Для порівняння реалізованих моделей буде використовуватися 2 групи метрик: метрики оцінки якості синтезу зображення (якості генерації) та метрики оцінки якості реєстрації пози камери.

3.2.1 Метрики для оцінки генерації зображень

Перша група складається з трьох метрик: Індекс структурної подібності (далі SSIM) [35], вивчена перцептивна схожість патчів зображень (далі LPIPS) [36] та пікове співвідношення сигналу до шуму (далі PSNR).

SSIM широко використовується для визначення якості зображень, а саме визначення наскільки згенероване зображення подібне до оригінального. Цей індекс коливається в діапазоні від -1 до 1, причому 1 вказує на те, що зображення одинакові.

Вона вимірює структурну подібність між двома зображеннями на основі трьох основних факторів: яскравості, контрасту та структурної інформації. В цьому контексті, контраст – це значення, яке показує різницю між світлими і темними ділянками зображення. А структурна інформація стосується розташування пікселів на зображенні і того, як вони співвідносяться один з одним.

В цьому досліджені використовується стандартне значення цього індексу, яке рахується наступним чином:

$$\mathrm{SSIM}(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}, \tag{3.1}$$

де x, y – зображення для порівняння;

μ_x, μ_y – значення яскравості відповідного зображення, $\mu_x = \frac{1}{N_x}\sum_{i=1}^{N} x_i$;

σ_x, σ_x – значення контрастності відповідного зображення, рахується за формулою середньоквадратичного відхилення всіх пікселі зображення;

σ_{xy} – коваріація двох зображень;

C_1, C_2 – константи для стабільності обчислення.

Наступна метрика, яка буде використовуватися, це LPIPS. Вона вимірює значення перцептивної схожості між зображеннями. В основі LPIPS лежить ідея, що людська зорова система сприймає зображення з точки зору локальних ділянок, а не глобальних особливостей.

Вона обчислює схожість між локальними ділянками двох зображень, порівнюючи представлення їх глибинних ознак. Ці глибинні ознаки отримуються шляхом пропускання ділянок через попередньо навчену глибоку нейронну мережу.

LPIPS має кілька переваг над іншими метриками якості зображень. По-перше, вона базується на глибоких нейронних мережах, які довели свою високу ефективність у відображенні людської інтерпретації. По-друге, має високу точність і добре корелює із людським сприйняттям, що робить його надійною метрикою для оцінки якості зображення. Нарешті, LPIPS є дуже гнучким і може бути легко адаптуватися до різних застосувань шляхом точного налаштування глибокої нейронної мережі.

Для оцінки моделей в цьому досліджені використовувалася метрика LPIPS основана на AlexNet.

Останньою метрикою цієї групи є PSNR – метрика, яка вимірює різницю між двома зображеннями з точки зору середньоквадратичної похибки (далі MSE) між значеннями пікселів двох зображень. Якщо більш детально, то PSNR обчислюється як відношення пікової потужності до MSE і розраховується за наступною формулою:

$$\mathrm{PSNR}(x, y) = 10 \log_{10} \left(\frac{R^2}{\mathrm{MSE}(x, y)} \right), \tag{3.2}$$

де x, y – зображення для порівняння;

$\mathrm{MSE}(x, y)$ – попіксельна середньоквадратична похибка;

R – пікове значення потужності сигналу.

Пікова потужність сигналу визначається як максимальне значення типу даних зображення. Тобто 1 якщо тип даних з плаваючою точкою, 255 якщо ціле 8-бітне, тощо. PSNR виражається в децибелах (дБ) і є логарифмічною мірою різниці між двома зображеннями.

Однак PSNR має певні обмеження. Воно не враховує відмінності у сприйнятті двох зображень і може неточно відображати якість зображення, як його сприймає людина. Тому ця метрика буде використовуватися в парі з SSIM та LPIPS, для більш повної оцінки якості зображення.

3.2.2 Метрики оцінки якості оптимізації поз камери

Другою групою є метрики оцінки якості реєстрації пози камери, до яких входить середня помилка повороту камери (в градусах) та середня помилка зміщення (в метрах).

Для обчислення останньої помилки рахується довжина вектору різниці між зміщеннями:

$$\Delta t_i = \| t_i - \hat{t}_i \|_2, i = \{1, \dots, M\} \tag{3.3}$$

де Δt_i – значення помилки зміщення i-ої камери;

t_i – реальне значення зміщення i-ої камери;

$\hat{t}_i$ – оптимізоване значення i-ої камери.

Для розрахунку помилки повороту пози камери використовується

більш складний алгоритм: спочатку треба вирівняти оптимізовану до реальної позу камери, а вже потім рахувати кутову відстань між отриманими значеннями.

Вирівняння відбувається за допомогою аналізу прокрустових кіл (рішення прокрустової задачі для двох точок). Цей алгоритм знаходить перетворення між координатними системами даних камер використовуючи сингулярну декомпозицію нормованого добутку позицій камер. Отримавши це перетворення, воно помножається на оптимізовану матриці повороту, тим самим переводячи систему координат оптимізованої пози до системи координат реальної.

Далі помилка вираховується за допомогою формули кутової відстані (арккосинусові відстані):

$$\Delta\theta_i = \cos^{-1}\left(\frac{\text{trace}\left(R_i \hat{R}_i'^{T}\right) - 1}{2}\right), i = \{1, \dots, M\} \tag{3.4}$$

де $\Delta\theta_i$ – значення помилка повороту i-ої камери;

$\text{trace}(\cdot)$ – слід матриці (сума значень діагоналі матриці);

R_i – реальне значення повороту i-ої камери;

$\hat{R}_i'$ – вирівняне оптимізоване значення повороту i-ої камери.

3.3 Опис експериментів

Відповідно до мети цього дослідження, всі експерименти націлені на адаптацію архітектури PNeRF до відсутності або неточності поз камери. Але для підтвердження життєздатності представленої архітектури також необхідно провести експерименти з класичними наборами даних (синтетичні 3Д об’єкти та сцени LLFF описані в підрозділі 3.1.2) для визначення спроможності цієї архітектури працювати на випадках з точними

позами камер.

Основними експериментами є тренування архітектур BARF, PNeRF та модифікований PNeRF на обраних сценах зі ScanNet.

Цілю тренування на цих 3Д сценах архітектури BARF є практичне доведення того, що класичний механізм кодування даних NeRF погано працює з великими 3Д сценами.

Тренування PNeRF повинно дати гарні результати на великих сценах з точними позами, але повністю повинно провалитися на даних без поз камери.

Експерименти з модифікованою архітектурою PNeRF повинно показати прийнятні результати на сценах ScanNet без поз камери та не гірші, від результатів оригінальної архітектури, на цих же сценах, але з точними позами камер.

Для проведення експериментів на мережі в якості невідомих поз камери подається одинична матриця розміром 4 на 4. Для навчання моделей всі вхідні зображення приводяться до розміру 480 на 640 пікселів. Всі моделі тренувалися 200 тисяч ітерацій з відповідними стратегіями описаними в розділі 2.

Для написання моделей, алгоритмів навчання та тестування, стратегій використовується мова програмування Python і відповідні фреймворки для глибокого навчання (PyTorch і TensorFlow).

4 ВПРОВАДЖЕННЯ РЕЗУЛЬТАТІВ ТА ПЕРСПЕКТИВИ РОЗВИТКУ

4.1 Аналіз результатів проведених експериментів

Для оцінки працеспроможності реалізованих алгоритмів і стратегій та оцінки їх якості було проведено ряд експериментів. Умовно їх можна поділити на три групи: перша – перевірка працеспроможності на синтетичних 3Д об’єктах та маленьких сценах з наборів даних, описаних в підрозділі 3.1.2; друга – перевірка існуючих архітектур на великих сценах ScanNet без відомих поз камер; третя – визначення якості роботи модифікованої архітектури, описаної в підрозділі 2.5.

Далі буде наведено результати цих експериментів, їх аналіз та відповідні висновки.

4.1.1 Експерименти на синтетичних та невеликих 3Д об’єктах

Як зазначалося вище, основною метою цієї групи експериментів було визначення життєздатності моделей, правильності їх реалізації та реалізації алгоритмів підготовки даних. Було перевірено моделі BARF, PNeRF та модифікований PNeRF на даних описаних в підрозділі 3.1.2 з реальними позами камер.

Перший експеримент був поставлений на 3Д об’єкті Lego бульдозера (рис. 4.1). Всі 3 моделі навчалися 200 тисяч ітерацій з відомими позами камер.

Як можна побачити на рисунку 4.2 всі 3 моделі гарно відновили форму, колір та текстуру об’єкту. Краще себе показала модель PNeRF, відновивши прозорість відповідних деталей і якісніше відобразивши тіні.

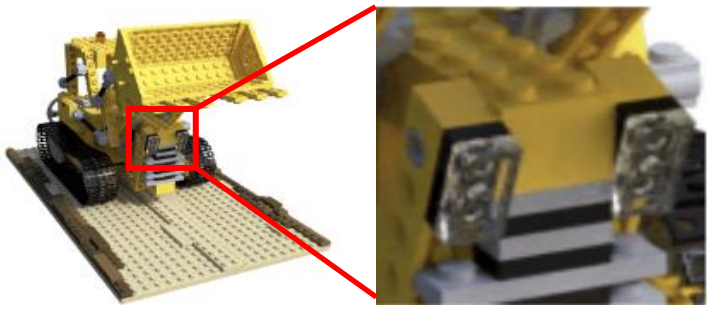

Рисунок 4.1 – 3Д об’єкт Lego бульдозера з локацією для порівняння якості

Рисунок 4.2 – Результати реконструкції зображення 3Д об’єкта моделями (зліва направо): BARF, PNeRF і модифікований PNeRF

В загалом можна сказати, що всі архітектури виконали поставлену ціль і з гарною деталізацією реконструювали цей об’єкт.

Наступний експеримент передбачав відновлення зображення невеликої 3Д сцени з набору даних LLFF. Сцена представляє собою маленьке дерево всередині приміщення (рис. 4.3).

Як можна побачити з рисунка 4.4, моделі на основі PNeRF архітектури показали кращу якість реконструкції ніж BARF. Це можу бути обумовлене тим, що сцена надто велика та різноманітна на деталі (листя дерева, смітники на задньому плані, відбиття на склі, тощо). Через це і принципові обмеження архітектури BARF (а саме жорстко зафіксована максимальна кількість інформації, яку може «запам’ятати» персептрон), результати цієї

моделі показали значну просадку в деталізації, яка виражена ефектом розмиття.

Рисунок 4.3 – Маленька 3Д сцена з набору даних LLFF з локацією для порівняння якості

Рисунок 4.4 – Результати реконструкції зображення 3Д сцени LLFF моделями (зліва направо): BARF, PNeRF і модифікований PNeRF

Та ж сама ситуація, але менш критична, повторюється і на іншій сцені з набору даних LLFF (рис. 4.5). Це доволі велика сцена, на якій основну увагу варто приділити можливості архітектур якісно відновлювати такі маленькі деталі як ребра скелету динозавра на певному віддаленні.

З результатів на рисунку 4.6 можна побачити, що архітектура BARF і стандартна PNeRF викривили кістки, на відміну від модифікованого PNeRF, який запам’ятав правильну геометрію і зміг краще відтворити ці деталі. Гірше за всіх в цьому експерименті показала себе архітектура BARF через

описані вище причини.

Рисунок 4.5 – 3Д сцена «T-Rex» з LLFF з локацією для порівняння якості

Рисунок 4.6 – Результати реконструкції зображення 3Д сцени LLFF (T-Rex) моделями (зліва направо): BARF, PNeRF і модифікований PNeRF

Нижче наведено таблицю з метриками, які повністю підтверджують наведені вище результати (таблиця 4.1).

Таблиця 4.1 – Метрики експериментів з даними NeRF та LLFF

Назва сцени	Метрики якості реконструкції зображення								
	PSNR ↑			SSIM ↑			LPIPS ↓		
	BARF	PNeRF	мод. PNeRF	BARF	PNeRF	мод. PNeRF	BARF	PNeRF	мод. PNeRF
Lego бульдозер	21.18	**26.42**	22.35	0.75	**0.91**	0.84	0.21	**0.12**	0.19
Пальма (LLFF)	23.51	**24.12**	23.87	0.7	**0.73**	0.71	0.35	**0.31**	0.34
T-Rex (LLFF)	22.94	23.04	**23.17**	0.68	0.68	**0.7**	0.26	**0.23**	0.24

В якості загального висновку щодо цієї групи експериментів, можна сказати, що всі досліджувані архітектури виконали поставлені цілі і довели правильність їх реалізації.

4.1.2 Експерименти на сценах ScanNet стандартних архітектур BARF і PNeRF

Метою цієї групи експериментів є практичне доведення твердження неспроможності архітектури, в основі якої лежить кодування 3Д сцени у ваги персептрону (якою і є BARF), реконструювати великі 3Д сцени (такі як приміщення).

Для досягнення поставленої цілі було використано 2 сцени з набору даних ScanNet, описаних в підрозділі 3.1.1. Модель BARF було навчено з позами (по факту в режимі звичайної NeRF) та без них (зі стратегією «реєстрації від грубого до точного » та іншими алгоритмами, описаними в підрозділі 2.3), а стандартну архітектуру PNeRF лише з позами камер, так як без них вона не може виконувати реконструкцію.

На рисунках 4.7 і 4.8 наведено приклади оригінальних зображень з обраних сцен.

Рисунок 4.7 – Приклад зображень зі сцени 0005_00 ScanNet

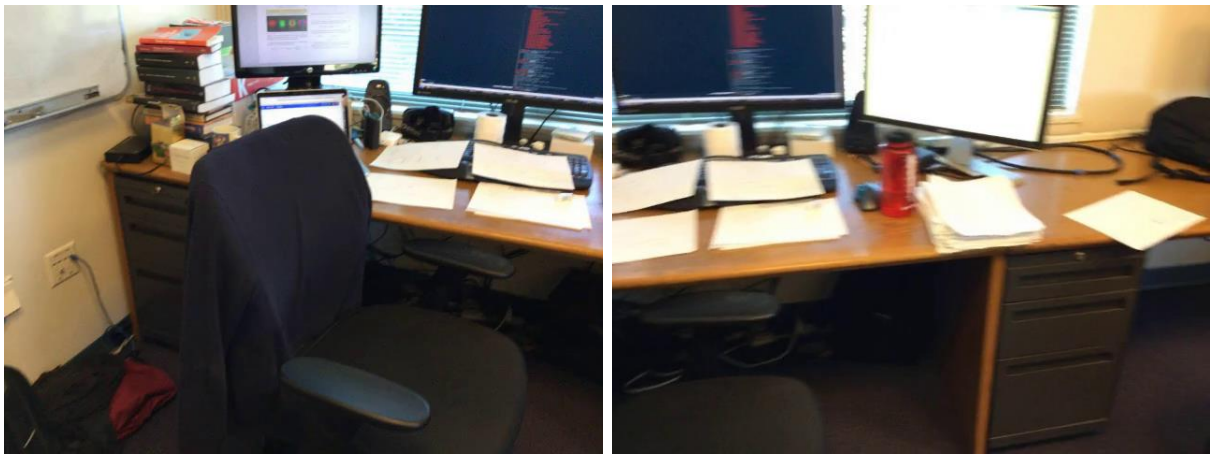

Рисунок 4.8 – Приклад зображень зі сцени 0010_00 ScanNet

На рисунку 4.9 наведено результат реконструкції зображення зі сцени 0005_00 (ліве) та сцени 0010_00 (праве) з відомими позами камер. Отримані результати вказують на те, що навіть з усіма відомими ця архітектура погано відновлює 3Д сцену. Як результат маємо сильно розмиті зображення з незначною кількістю деталей. Ще гірше результати на сцені 0010_00, де майже немає навіть контурів об’єктів. Це можна пояснити наявністю великої кількості неламбертових поверхонь (з дзеркальним відображенням) – а саме рамок моніторів.

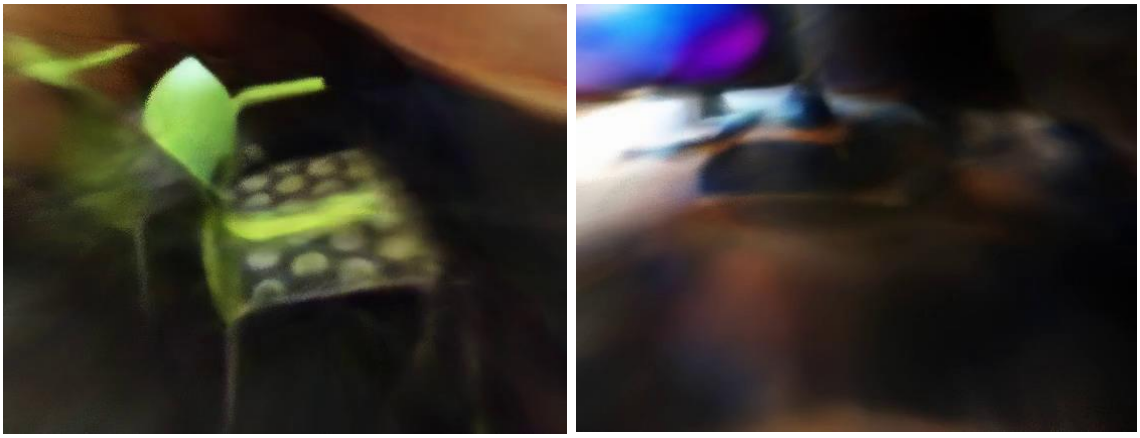

Рисунок 4.9 – Результат реконструкції 3Д сцен ScanNet архітектурою BARF з позами камер (зліва сцена 0005_00, справа – 0010_00)

Дані результати повністю підтверджують тезис стосовного того, що стандартний неглибокий БШП NeRF погано відновлює 3Д сцени типу приміщень.

Ситуація становиться ще гіршою, якщо пози камери невідомі. Як видно на рисунку 4.10, на обох сценах майже не розрізняються окремі об’єкти, все дуже змазано і можна сказати, що реконструкція повністю не виконана.

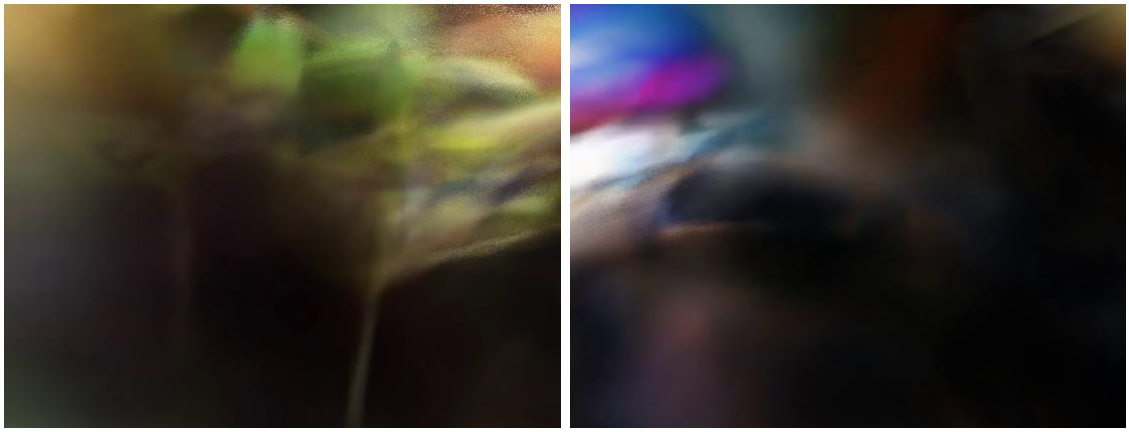

Рисунок 4.10 – Результат реконструкції 3Д сцен ScanNet архітектурою BARF без поз камер (зліва сцена 0005_00, справа – 0010_00)

Провівши аналогічний експеримент з архітектурою PNeRF з відомими позами камер отримуємо дуже якісні результати реконструкції (рис. 4.11). Результат не ідеальний – біля стільця на сцені 0005_00 є спотворення простору, на сцені 0010_00 є дуже сильна розмитість в місцях дзеркального відбиття – але в загалом ми маємо доволі якісну реконструкцію з великою кількістю деталей, правильними кольорами і подекуди навіть з правильним віддзеркаленням об’єктів (рис. 4.12).

Як було припущено PNeRF набагато краще працює з великими 3Д сценами, зберігаючи більшість деталей. Ті недоліки, які збереглися, можливо виправити більш точною підгонкою гіперпараметрів і довшим тренуванням (слід нагадати, що всі моделі тренувалися 200 тисяч ітерацій).

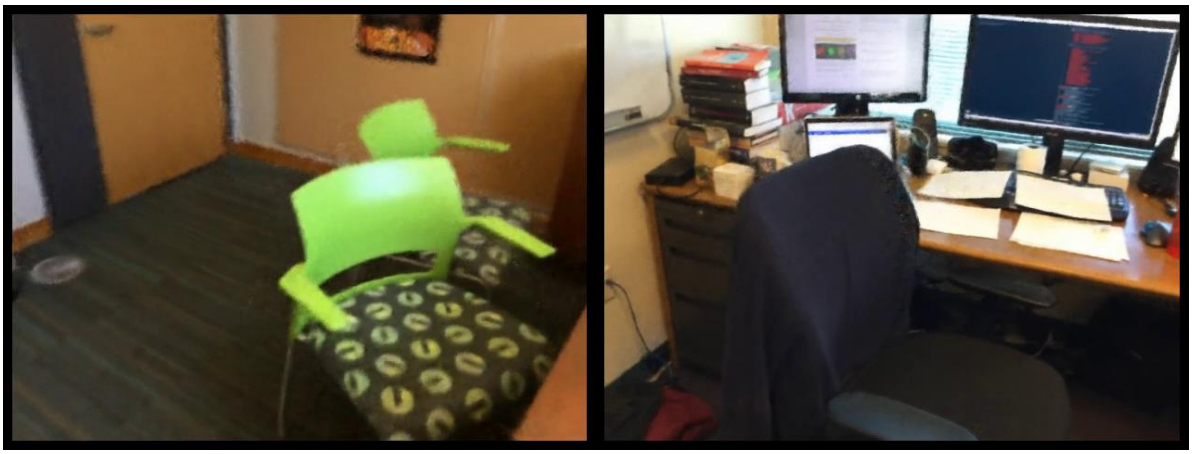

Рисунок 4.11 – Результат реконструкції 3Д сцен ScanNet архітектурою PNeRF з позами камер (зліва сцена 0005_00, справа – 0010_00)

Рисунок 4.12 – Якість реконструкції неламбертових поверхонь на сцені 0005_00 ScanNet архітектурою PNeRF

Всі наведені вище результати і ефекти підтверджують розраховані метрики, які наведені нижче в таблиці 4.2.

Таблиця 4.2 – Метрики експериментів з даними ScanNet

Назва сцени	Метрики якості реконструкції зображення								
	PSNR ↑			SSIM ↑			LPIPS ↓		
	BARF з позами	BARF без поз	PNeRF	BARF з позами	BARF без поз	PNeRF	BARF з позами	BARF без поз	PNeRF
0005_00	18.25	10.47	**29.73**	0.583	0.357	**0.84**	0.58	0.71	**0.14**
0010_00	13.53	8.61	**28.4**	0.492	0.27	**0.871**	0.61	0.75	**0.14**

Більше результатів на цих сценах наведено в додатку А. Також там наведено рисунки з реконструйованими картами глибин. Отже, як поставлені експерименти практично довели, що ідея кодувати 3Д об’єкти в ваги персептрону, дуже погано працює з великими 3Д сценами і особливо в умовах, коли невідомі пози камер.

4.1.3 Експерименти модифікованої PNeRF на ScanNet

Основними експериментами цього дослідження є експерименти з модифікованою архітектурою PNeRF на великих сценах приміщень з набору даних ScanNet. Було проведено ряд експериментів, в яких ця модель навчалася в умовах, коли пози камери невідомі. Як і у попередніх експериментах, замість поз камери було подано одиничну матрицю розміром 4x4, а тривалість навчання – 200 тисяч ітерацій.

Результати експерименту наведені на рисунку 4.13. На них можна побачити, що якість відновлення трішки гірша за оригінальну PNeRF. На сцені 0005_00 не чіткі форми дверей та картини на стіні, а на сцені 0010_00 – спинка стільця «просвічується». Але не дивлячись на такі недоліки, модель справилася з поставленою задачею, відновивши пози камер і провівши реконструкцію на рівні, кращому за BARF.

Особливу увагу варто приділити неламбертовій поверхні, яка була реконструйована на рівні не гіршому ніж звичайний PNeRF з відомими позами камер (рис. 4.14).

Рисунок 4.13 – Результат реконструкції 3Д сцен ScanNet модифікованою архітектурою PNeRF без поз камер (зліва сцена 0005_00, справа – 0010_00)

Рисунок 4.14 – Якість реконструкції неламбертових поверхонь на сцені 0005_00 ScanNet модифікованою архітектурою PNeRF

Метрики цих експериментів наведені в таблиці 4.3 і доводять, що модель не тільки непогано провела реконструкцію, а й дуже точно оптимізувала пози камер.

Більше реконструйованих зображень цією архітектурою, а також відповідні карти глибин, наведено в додатку А.

Таблиця 4.3 – Метрики експериментів з модифікованою архітектурою PNeRF на даних ScanNet

Назва сцени	Метрики якості оптимізації поз камери		Метрики якості реконструкції зображення		
	Помилка повороту, градус (°)	Помилка здвигу, см	PSNR ↑	SSIM ↑	LPIPS ↓
0005_00	0.91	1.73	27.31	0.81	0.15
0010_00	1.33	1.21	27.2	0.83	0.17

Отже, можна зробити висновок, що модифікована архітектура PNeRF дає порівняну якість реконструкції зображення в умовах відсутності поз камери. При цьому вона дуже непогано проводить нейронне корегування променів (neural bundle adjustment) від час тренування і доволі якісно оптимізує пози камер.

4.2 Перспективи розвитку

В рамках цього дослідження було проведено розробку та оцінку створеної модифікації архітектури поточкової NeRF для випадку відсутності поз камер за допомогою механізму нейронного bundle adjustment (далі BA). Одними із можливих шляхів розвитку моделі є вдосконалення якості реконструкції зображень та її швидкості методами поліпшення структури нейронної хмарини точок (заміною звичайних точок на нейронні вокселі), оптимізацією ансамбля нейронних мереж (заміною його на глибоку мережу з комплексними входами) або спробою перейти до архітектури графових нейронних мереж. Іншими шляхами вдосконалення є більш складна стратегія пошуку поз камери, яка може полягати в більш жорсткій регулярізації функції втрат та додаванню окремої мережі, яка б прогнозувала початкові наближення поз (проводила глобальну реєстрацію), на основі яких потім проводиться локальна оптимізація за допомогою нейронного BA.

ВИСНОВКИ

Метою даної магістерської кваліфікаційної роботи було дослідження методів реконструкцій зображень великих 3Д сцен за допомогою поточкових Neural Radiance Fields в умовах невідомих поз камер.

На етапі аналізу предметної області та існуючих рішень було проаналізовано, структуровано та описано основні підходи та архітектури, які існують на даних момент для задачі реконструкції зображень. Серед знайдених підходів було вирішено сконцентрувати увагу на моделях сімейства NeRF. Серед них було виділено та класифіковано, відповідно до задач, що вони вирішують, ряд архітектур та модифікацій. В якості задачі для вирішення було обрано реконструкцію великих 3Д сцен, таких як приміщення, за умови невідомих поз камери. Відповідно до поставленої задачі було знайдено та досліджено набір архітектур, які теоретично могли вирішити її. На основі проведеного аналізу предметної галузі була сформульована постановка задачі кваліфікаційної роботи.

На етапі проектування складено детальний математичний опис архітектур NeRF, BARF і поточкової NeRF. Детально розглянуто та описано основні механізми, стратегії та алгоритми, які використовують ці моделі. На цьому ж етапі було адаптовано архітектуру поточкової NeRF під поставлену задачу, методом адаптування деяких механізмів BARF під особливості PNeRF.

На етапі підготовки та планування експериментів було підібрано та описано ряд наборів даних, на яких обрані моделі мали продемонструвати як правильність роботи і життєздатність, так і можливості реконструкції великих 3Д сцен. Також на цьому етапі було визначено та описано метрики, які б числено мали охарактеризувати результати експериментів. Також було описано деталі програмної реалізації і проведено всі поставлені експерименти.

На етапі аналізу було згруповано, продемонстровано та описано отримані результати експериментів. Було проведено порівняння результатів в різних умовах та на різних наборах даних, пораховано і представлено метрики та зроблено висновки стосовно працеспроможності моделей при різних умовах. Як результат було практично доведено припущення, щодо можливості та якості реконструкції великих 3Д сцен без поз камери модифікованою архітектурою PNeRF.

Таким чином можна зробити висновок, що представлена модифікація архітектури Point-based Neural Radiance Fields достатньо якісно впоралась із поставленою задачею відновлення зображень великої 3Д сцени в умовах навчання з невідомими позами камер та задачею їх відновлення за допомогою нейронного bundle adjustment.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. BARF: bundle-adjusting neural radiance fields / C.-H. Lin та ін. *2021 IEEE/CVF international conference on computer vision (ICCV)*, м. Montreal, QC, Canada, 10-17 жовт. 2021 р. 2021. URL: https://doi.org/10.1109/iccv48922.2021.00569 (дата звернення: 20.03.2023).

2. Chen W., Gao J., Ling H. Learning to predict 3D objects with an interpolation-based differentiable renderer. *NeurIPS*, м. Vancouver. URL: https://doi.org/10.48550/arXiv.1908.01210 (дата звернення: 20.03.2023).

3. Curless B., Levoy M. A volumetric method for building complex models from range images. *The 23rd annual conference*. New York, New York, USA, 1996. URL: https://doi.org/10.1145/237170.237269 (дата звернення: 20.03.2023).

4. DeepView: view synthesis with learned gradient descent / J. Flynn та ін. *2019 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, м. Long Beach, CA, USA, 15-20 черв. 2019 р. 2019. URL: https://doi.org/10.1109/cvpr.2019.00247 (дата звернення: 20.03.2023).

5. DeepVoxels: learning persistent 3D feature embeddings / V. Sitzmann та ін. *2019 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, м. Long Beach, CA, USA, 15-20 черв. 2019 р. 2019. URL: https://doi.org/10.1109/cvpr.2019.00254 (дата звернення: 20.03.2023).

6. Differentiable Monte Carlo ray tracing through edge sampling / T.-M. Li та ін. *ACM transactions on graphics*. 2019. Т. 37, № 6. С. 1-11. URL: https://doi.org/10.1145/3272127.3275109 (дата звернення: 20.03.2023).

7. Differentiable volumetric rendering: learning implicit 3D representations without 3D supervision / M. Niemeyer та ін. *2020 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, м. Seattle, WA, USA, 13-19 черв. 2020 р. 2020. URL: https://doi.org/10.1109/cvpr42600.2020.00356 (дата звернення: 20.03.2023).

8. Dinh L., Sohl-Dickstein J., Bengio S. Density estimation using Real NVP. *5th international conference on learning representations*, м. Toulon, 24-26 квіт. 2017 р. URL: https://doi.org/10.48550/arXiv.1605.08803 (дата звернення: 20.03.2023).

9. Global illumination with radiance regression functions / P. Ren та ін. *ACM transactions on graphics*. 2013. Т. 32, № 4. С. 1-12. URL: https://doi.org/10.1145/2461912.2462009 (дата звернення: 20.03.2023).

10. GNeRF: gan-based neural radiance field without posed camera / Q. Meng та ін. *2021 IEEE/CVF international conference on computer vision (ICCV)*, м. Montreal, QC, Canada, 10-17 жовт. 2021 р. 2021. URL: https://doi.org/10.1109/iccv48922.2021.00629 (дата звернення: 20.03.2023).

11. Henzler P., Mitra N. J., Ritschel T. Learning a neural 3D texture space from 2D exemplars. *2020 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, м. Seattle, WA, USA, 13-19 черв. 2020 р. 2020. URL: https://doi.org/10.1109/cvpr42600.2020.00838 (дата звернення: 20.03.2023).

12. Hernandez Esteban C., Vogiatzis G., Cipolla R. Multiview photometric stereo. *IEEE transactions on pattern analysis and machine intelligence*. 2008. Т. 30, № 3. С. 548-554. URL: https://doi.org/10.1109/tpami.2007.70820 (дата звернення: 20.03.2023).

13. Hierarchical representations and explicit memory: learning effective navigation policies on 3D scene graphs using graph neural networks / Z. Ravichandran та ін. *2022 IEEE international conference on robotics and automation (ICRA)*, м. Philadelphia, PA, USA, 23-27 трав. 2022 р. 2022. URL: https://doi.org/10.1109/icra46639.2022.9812179 (дата звернення: 20.03.2023).

14. Kanatani K., Sugaya Y., Kanazawa Y. Bundle adjustment. *Guide to 3D vision computation*. Cham, 2016. С. 149-161. URL: https://doi.org/10.1007/978-3-319-48493-8_11 (дата звернення: 20.03.2023).

15. Kutulakos K. N., Seitz S. M. A theory of shape by space carving. *Proceedings of the seventh IEEE international conference on computer vision*,

м. Kerkyra, Greece, 20-27 верес. 1999 р. 1999. URL: https://doi.org/10.1109/iccv.1999.791235 (дата звернення: 20.03.2023).

16. Local deep implicit functions for 3D shape / K. Genova та ін. *2020 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, м. Seattle, WA, USA, 13-19 черв. 2020 р. 2020. URL: https://doi.org/10.1109/cvpr42600.2020.00491 (дата звернення: 20.03.2023).

17. Local implicit grid representations for 3D scenes / C. Jiang та ін. *2020 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, м. Seattle, WA, USA, 13-19 черв. 2020 р. 2020. URL: https://doi.org/10.1109/cvpr42600.2020.00604 (дата звернення: 20.03.2023).

18. Lombardi S., Simon T., Saragih J. Neural volumes: learning dynamic renderable volumes from images. *ACM transactions on graphics (SIGGRAPH 2019)*. 2019. Т. 29, № 4. 65. URL: https://doi.org/10.48550/arXiv.1906.07751 (дата звернення: 20.03.2023).

19. NeRF--: neural radiance fields without known camera parameters / Z. Wang та ін. *NeurIPS*, м. Vancouver, 14 лют. 2021 р. URL: https://doi.org/10.48550/arXiv.2102.07064 (дата звернення: 20.03.2023).

20. NeRF: representing scenes as neural radiance fields for view synthesis / B. Mildenhall та ін. *Computer vision - ECCV 2020*. Cham, 2020. С. 405-421. URL: https://doi.org/10.1007/978-3-030-58452-8_24 (дата звернення: 20.03.2023).

21. PixelNeRF: neural radiance fields from one or few images / A. Yu та ін. *2021 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, м. Nashville, TN, USA, 20-25 черв. 2021 р. 2021. URL: https://doi.org/10.1109/cvpr46437.2021.00455 (дата звернення: 20.03.2023).

22. Point-NeRF: point-based neural radiance fields / Q. Xu та ін. *2022 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, м. New Orleans, LA, USA, 18-24 черв. 2022 р. 2022. URL: https://doi.org/10.1109/cvpr52688.2022.00536 (дата звернення: 20.03.2023).

23. Porter T., Duff T. Compositing digital images. *ACM SIGGRAPH computer graphics*. 1984. Т. 18, № 3. С. 253-259. URL: https://doi.org/10.1145/964965.808606 (дата звернення: 20.03.2023).

24. RegNeRF: regularizing neural radiance fields for view synthesis from sparse inputs / M. Niemeyer та ін. *2022 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, м. New Orleans, LA, USA, 18-24 черв. 2022 р. 2022. URL: https://doi.org/10.1109/cvpr52688.2022.00540 (дата звернення: 20.03.2023).

25. Rosinol A., Leonard J. J., Carlone L. NeRF-SLAM: real-time dense monocular SLAM with neural radiance fields. *ACM transactions on graphics (SIGGRAPH 2019)*. 2022. URL: https://doi.org/10.48550/arXiv.2210.13641 (дата звернення: 20.03.2023).

26. Single-image tomography: 3D volumes from 2D cranial x-rays / P. Henzler та ін. *Computer graphics forum*. 2018. Т. 37, № 2. С. 377-388. URL: https://doi.org/10.1111/cgf.13369 (дата звернення: 20.03.2023).

27. Stanley K. O. Compositional pattern producing networks: a novel abstraction of development. *Genetic programming and evolvable machines*. 2007. Т. 8, № 2. С. 131-162. URL: https://doi.org/10.1007/s10710-007-9028-8 (дата звернення: 20.03.2023).

28. Tang C., Tan P. BA-Net: dense bundle adjustment network. *NeurIPS*, м. Vancouver, 25 серп. 2019 р. URL: https://doi.org/10.48550/arXiv.1806.04807 (дата звернення: 20.03.2023).

29. The lumigraph / S. J. Gortler та ін. *The 23rd annual conference*, м. Not Known. New York, New York, USA, 1996. URL: https://doi.org/10.1145/237170.237200 (дата звернення: 20.03.2023).

30. Unstructured lumigraph rendering / C. Buehler та ін. *The 28th annual conference*, м. Not Known. New York, New York, USA, 2001. URL: https://doi.org/10.1145/383259.383309 (дата звернення: 20.03.2023).

31. Waechter M., Moehrle N., Goesele M. Let there be color! Large-scale texturing of 3D reconstructions. *Computer vision - ECCV 2014*. Cham,

2014. С. 836-850. URL: https://doi.org/10.1007/978-3-319-10602-1_54 (дата звернення: 20.03.2023).

32. Kajiya J. T., Von Herzen B. P. Ray tracing volume densities. *ACM SIGGRAPH computer graphics*. 1984. Т. 18, № 3. С. 165-174. URL: https://doi.org/10.1145/964965.808594 (дата звернення: 20.04.2023).

33. Rahaman N., Baratin A., Arpit D. On the spectral bias of neural networks. *Icml* : Міжнар. конф. з машин. навчання, м. Стокгольм, 10-15 лип. 2018 р.

34. Goyal P., Dollár P., Girshick R. Accurate, large minibatch SGD: training imagenet in 1 hour. *NeurIPS*. 2017. URL: https://arxiv.org/abs/1706.02677 (дата звернення: 20.04.2023).

35. Image quality assessment: from error visibility to structural similarity / Z. Wang та ін. *IEEE transactions on image processing*. 2004. Т. 13, № 4. С. 600-612. URL: https://doi.org/10.1109/tip.2003.819861 (дата звернення: 20.04.2023).

36. The unreasonable effectiveness of deep features as a perceptual metric / R. Zhang та ін. *2018 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, м. Salt Lake City, UT, 18-23 черв. 2018 р. 2018. URL: https://doi.org/10.1109/cvpr.2018.00068 (дата звернення: 20.04.2023).

ДОДАТОК А

Результати експериментів на наборі даних ScanNet

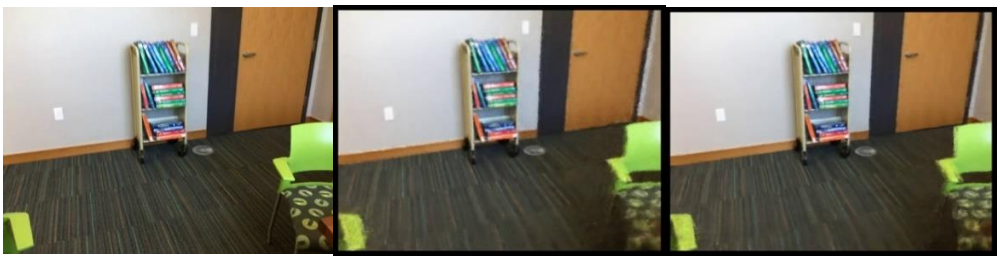

Рисунок А.1 – Результати реконструкції зображення на 1 кадрі сцени 0005_00 ScanNet моделей (зліва направо): реальне зображення, PNeRF, модифікований PNeRF

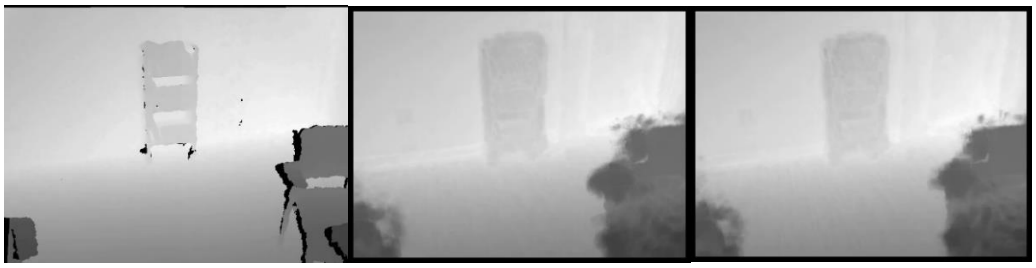

Рисунок А.2 – Результати реконструкції карт глибин на 1 кадрі сцени 0005_00 ScanNet моделей (зліва направо): реальне зображення, PNeRF, модифікований PNeRF

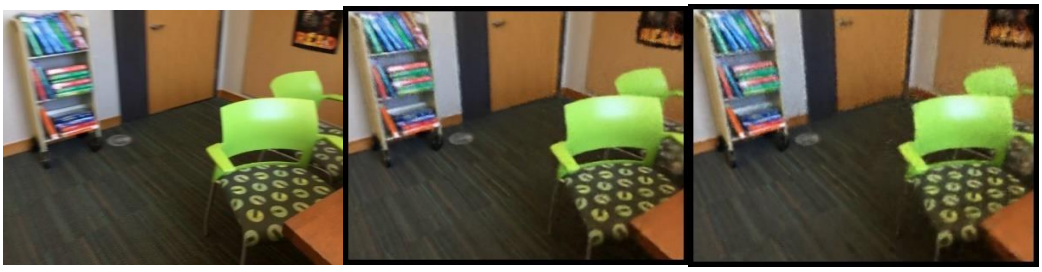

Рисунок А.3 – Результати реконструкції зображення на 75 кадрі сцени 0005_00 ScanNet моделей (зліва направо): реальне зображення, PNeRF, модифікований PNeRF

Рисунок А.4 – Результати реконструкції карт глибин на 75 кадрі сцени 0005_00 ScanNet моделей (зліва направо): реальне зображення, PNeRF, модифікований PNeRF

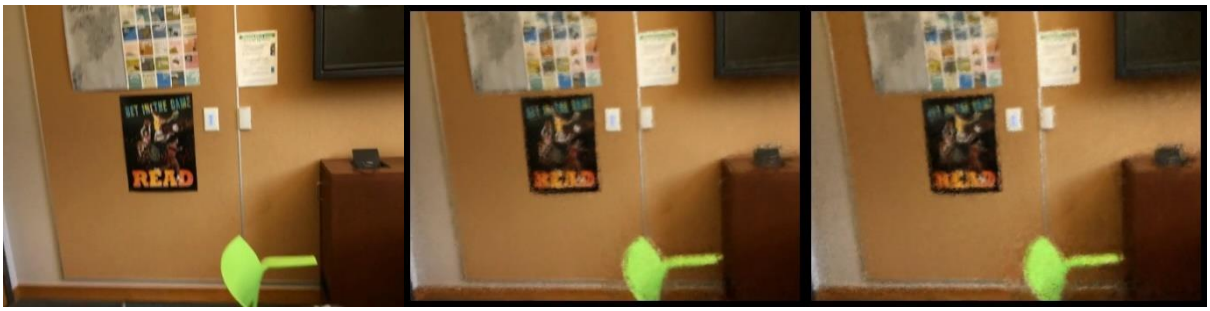

Рисунок А.5 – Результати реконструкції зображення на 150 кадрі сцени 0005_00 ScanNet моделей (зліва направо): реальне зображення, PNeRF, модифікований PNeRF

Рисунок А.6 – Результати реконструкції карт глибин на 150 кадрі сцени 0005_00 ScanNet моделей (зліва направо): реальне зображення, PNeRF, модифікований PNeRF

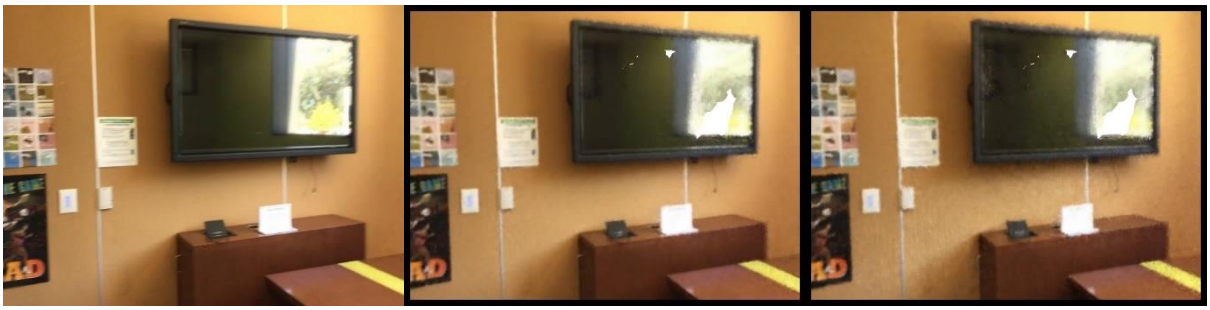

Рисунок А.7 – Результати реконструкції зображення на 190 кадрі сцени 0005_00 ScanNet моделей (зліва направо): реальне зображення, PNeRF, модифікований PNeRF

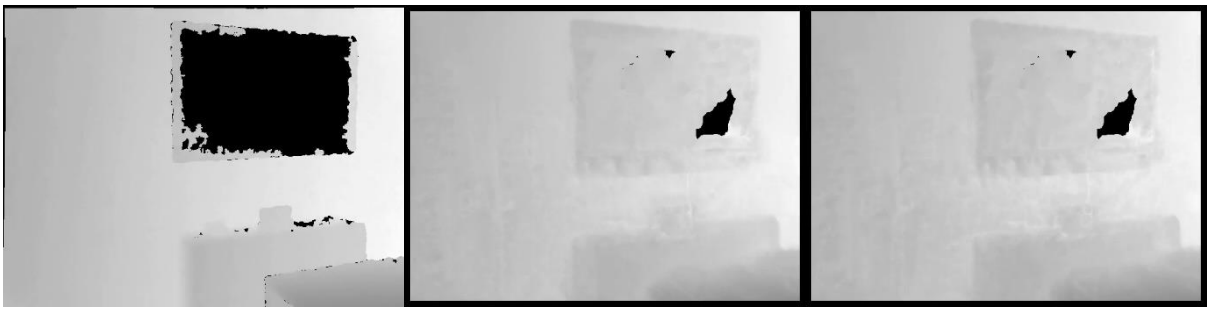

Рисунок А.8 – Результати реконструкції карт глибин на 190 кадрі сцени 0005_00 ScanNet моделей (зліва направо): реальне зображення, PNeRF, модифікований PNeRF

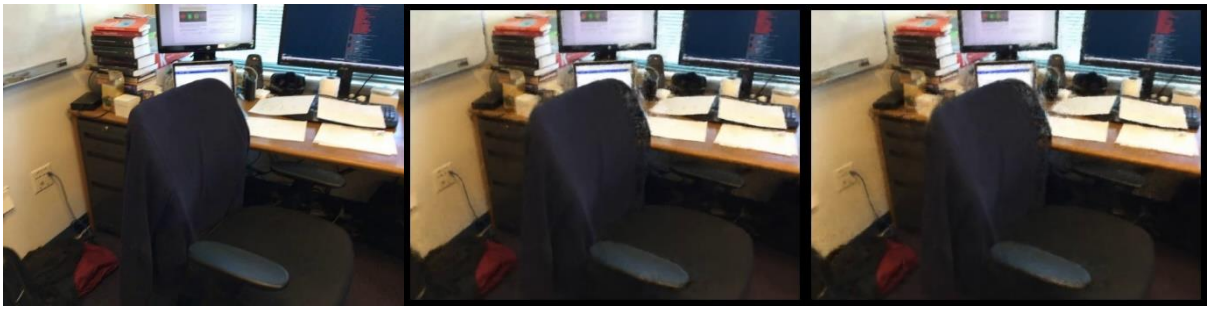

Рисунок А.9 – Результати реконструкції зображення на 1 кадрі сцени 0010_00 ScanNet моделей (зліва направо): реальне зображення, PNeRF, модифікований PNeRF

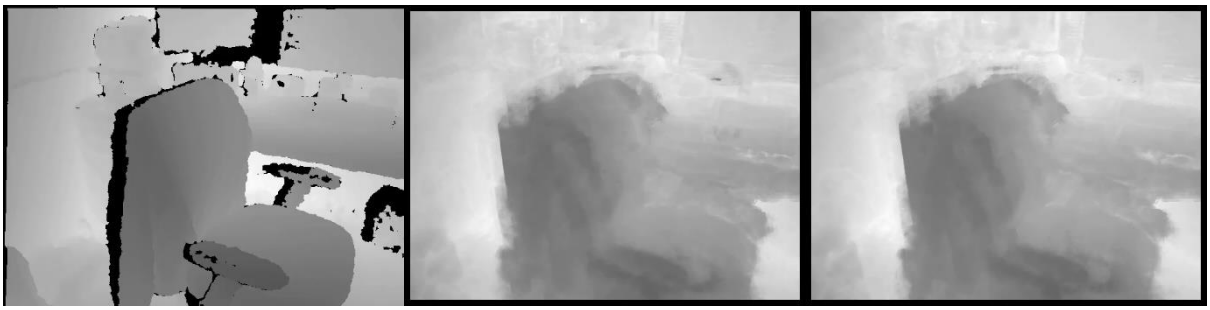

Рисунок А.10 – Результати реконструкції карт глибин на 1 кадрі сцени 0010_00 ScanNet моделей (зліва направо): реальне зображення, PNeRF, модифікований PNeRF

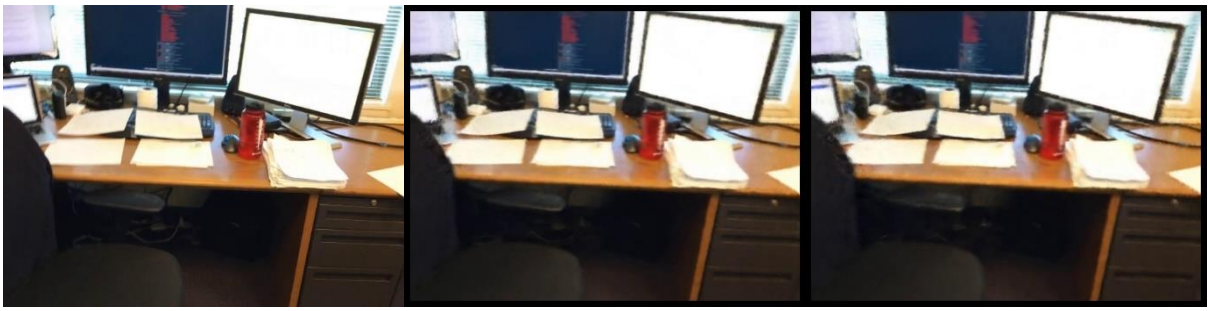

Рисунок А.11 – Результати реконструкції зображення на 200 кадрі сцени 0010_00 ScanNet моделей (зліва направо): реальне зображення, PNeRF, модифікований PNeRF

Рисунок А.12 – Результати реконструкції карт глибин на 200 кадрі сцени 0010_00 ScanNet моделей (зліва направо): реальне зображення, PNeRF, модифікований PNeRF

ДОДАТОК Б

Відомість кваліфікаційної роботи

Позначення	Найменування	Дод. відомості
	Текстові документи	
1.	Пояснювальна записка	81 с.
	Інші документи	
2.	Презентаційні матеріали	плакатів

.	.	Прізвище та ініціали.	Підп.	Дата	

Розробив	Шашко О.В.			Дослідження Point-based Neural Radiance Fields для реконструкції великих 3Д сцен з невідомими позами камер за допомогою нейронного bundle adjustment	Шифр групи	Код напр./спец.
Перевірив	Семенець В.В.				СШІм-21-1	122
Н.контр.	Малєєва І.А.				ХНУРЕ кафедра ШІ	
Затв.	Філатов В.О.					