

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ

Навчально-науковий інститут Інформаційних технологій

Кафедра інженерії програмного забезпечення

Пояснювальна записка

до магістерської роботи

на ступінь вищої освіти магістр

на тему: **«ЗАСТОСУВАННЯ МЕТОДУ ЗГЛАДЖУВАННЯ ЗОБРАЖЕНЬ В
ТРИВИМІРНІЙ КОМП'ЮТЕРНІЙ ГРАФІЦІ З ВИКОРИСТАННЯМ
АЛГОРИТМУ МАШИННОГО НАВЧАННЯ»**

Виконав: студент 6 курсу, групи ПДМ–61

спеціальності

121-Інженерія програмного забезпечення

(шифр і назва спеціальності)

Пашенко В. Ю.

(прізвище та ініціали)

Керівник Дібрівний О. А.

(прізвище та ініціали)

Рецензент _____

(прізвище та ініціали)

Нормоконтроль _____

(прізвище та ініціали)

Київ – 2022

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ

Навчально-науковий інститут Інформаційних Технологій

Кафедра Інженерії програмного забезпечення

Ступінь вищої освіти - «Магістр»

Напрямок підготовки - 121- Інженерія програмного забезпечення

ЗАТВЕРДЖУЮ

Завідувач кафедри

Інженерії програмного забезпечення

_____ О. В. Негоденко

“ _____ ” _____ 2022 року

З А В Д А Н Н Я

НА МАГІСТЕРСЬКУ РОБОТУ СТУДЕНТУ

_____ Пащенко Владислав Юрійович

(прізвище, ім'я, по батькові)

1. Тема роботи: «Тема роботи за наказом» «Застосування методу згладжування зображень в тривимірній комп'ютерній графіці з використанням алгоритму машинного навчання»

Керівник роботи _____ Дібрівний О.А., доктор філософії, доцент _____,
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу від 12 жовтня 2022 року №122.

2. Строк подання студентом роботи _____ 31.12.2022

3. Вхідні дані до роботи:

_____ Методи згладжування зображень, науково-технічна література

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити).

4.1 Аліасинг та згладжування зображень.

4.2 Штучний інтелект та нейронні мережі.

4.3 Аналіз існуючих методів.

4.4 Розробка методу на основі нейронної мережі.

5. Перелік демонстраційного матеріалу

1. Титульний слайд

2. Об'єкт, предмет та мета дослідження

3. Існуючі методи згладжування зображень

4. Порівняльна характеристика аналогів

5. Метод на основі нейронної мережі

6. Архітектура мережі

6. Структура генератора і дискриміатора

6. Приклад роботи методу

6. Висновки

7. Апробація результатів дослідження

9. Дата видачі завдання 14.10.2022

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів бакалаврської роботи	Строк виконання етапів роботи	Примітка
1	Підбір та вивчення джерел інформації	14.10.2022	Виконано
2	Розробка методу	15.11.2022	Виконано
3	Написання основної частини	12.12.2022	Виконано
4	Вступ, висновки, реферат	13.12.2022	Виконано
5	Розробка обов'язкових демонстраційних матеріалів	14.12.2022	Виконано
6	Попередній захист роботи	15.12.2022	Виконано
7	Подання роботи в деканат	31.12.2022	Виконано

Студент _____ Пащенко В. Ю.
(підпис) (прізвище та ініціали)

Керівник роботи _____ Дібрівний О. А.
(підпис) (прізвище та ініціали)

РЕФЕРАТ

Текстова частина магістерської роботи 56 с., 22 рис., 20 джерел.

ЗГЛАДЖУВАННЯ ЗОБРАЖЕНЬ, КОМП'ЮТЕРНА ГРАФІКА, ШТУЧНИЙ ІНТЕЛЕКТ, УМОВНА ГЕНЕРАТИВНА ЗМАГАЛЬНА НЕЙРОННА МЕРЕЖА.

Об'єкт дослідження – процес згладжування зображень в тривимірній комп'ютерній графіці.

Предмет дослідження – алгоритми машинного навчання, методи згладжування зображень.

Мета дослідження – підвищення ефективності згладжування зображень в тривимірній комп'ютерній графіці за рахунок використання алгоритмів машинного навчання.

Методи дослідження – класифікація, математична статистика, машинне навчання та інтелектуальний аналіз.

Під час виконання поставленого завдання магістерської роботи, було описано суть проблеми ефекту аліасингу в зображеннях, описано процес растирезації зображень в комп'ютерній графіці і показано в чому полягає необхідності згладжування зображень. Було здійснено аналіз існуючих методів згладжування зображень в тривимірній комп'ютерній графіці. На основі отриманих результатів аналізу розроблено новий метод згладжування зображень. Метод розроблено на основі умовної генеративної змагальної нейронної мережі. Було описано спосіб тестування мережі і показники, що були використані для оцінки ефективності роботи мережі. Було показано, як відбувалося тестування методу та надані результати його тестування.

ЗМІСТ

ВСТУП	10
1 АЛІАСИНГ ТА ЗГЛАДЖУВАННЯ ЗОБРАЖЕНЬ	11
1.1 Аліасинг	11
1.2 Згладжування зображень в комп'ютерній графіці	12
1.3 Висновки до розділу	13
2 ШТУЧНИЙ ІНТЕЛЕКТ ТА НЕЙРОННІ МЕРЕЖІ	14
2.1 Штучний інтелект	14
2.2 Нейронні мережі	14
2.3 Історія розвитку нейронних мереж	17
2.4 Архітектура нейронних мереж	18
2.5 Типи нейронних мереж	23
2.5.1 Перцептрон.	25
2.5.2 Багатошаровий перцептрон.	26
2.5.3 Згорткові нейронні мережі.	28
2.5.4 Мережа радіальних базисних функцій.	30
2.5.5 Рекурентні нейронні мережі.	32
2.5.6 Генеративні змагальні мережі.	35
2.5.7 Автоматичні кодувальники.....	37
2.6 Способи навчання нейронних мереж	38
2.6.1 Навчання з учителем.....	39
2.6.2 Навчання без вчителя	40
2.6.3 Напівавтоматичне навчання.....	40
2.6.4 Навчання з підкріпленням.....	40

2.6 Висновки до розділу.....	41
3 АНАЛІЗ ІСНУЮЧИХ МЕТОДІВ	42
3.1 Згладжування за допомогою суперсемплінгу	42
3.2 Згладжування з кількома вибірками	43
3.3 Швидке приблизне згладжування	44
3.4 Темпоральне згладжування	45
3.5 Висновки до розділу.....	47
4 РОЗРОБКА МЕТОДУ НА ОСНОВІ НЕЙРОННОЇ МЕРЕЖІ	48
4.1 Опис алгоритму роботи методу	48
4.2 Умовна генеративна змагальна мережа.....	49
4.3 Архітектура мережі.....	52
4.4 Навчання мережі.....	59
4.5 Тестування методу	61
4.6 Висновки до розділу.....	64
ВИСНОВКИ.....	65
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	66

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

SSAA (англ. Supersampling anti-aliasing) – згладжування за допомогою суперсемплінгу.

MSAA (англ. Multisample anti-aliasing) – згладжування з кількома вибірками.

FXAA (англ. Fast approximate anti-aliasing) – швидке приблизне згладжування.

TAA (англ. Temporal anti-aliasing) – темпоральне згладжування.

GAN (англ. Generative adversarial network) – генеративна змагальна мережа.

cGAN (англ. Conditional generative adversarial network) – умовна генеративна змагальна мережа.

FID (англ. Frechet inception distance) – початкова відстань Фреше.

SSIM (англ. Structural similarity index measure) – індекс структурної подібності.

ВСТУП

Під час генерації зображення за допомогою комп'ютерної графіки на зображенні можуть утворюватися артефакти через які зображення стає нерозбірливим і виглядає неестетично. Такі артефакти називаються ефектом аліасингу. Для їх усунення використовується згладжування зображень.

Актуальність теми полягає в тому, що на сьогоднішній день існує велика кількість різних методів згладжування зображень але у всіх цих методів є суттєві недоліки. Тому зараз розробка методу, що був би оптимальним, а саме дозволяв би отримувати зображення високої якості і працював би достатньо швидко є однією з найбільш актуальних проблем в сучасній комп'ютерній графіці.

Метою магістерської роботи є підвищення ефективності згладжування зображень в тривимірній комп'ютерній графіці за рахунок використання алгоритмів машинного навчання.

Завданням дослідження є проектування та реалізація методу згладжування зображень на основі алгоритмів машинного навчання, який можна було б використовувати в тривимірній комп'ютерній графіці.

Об'єкт дослідження - процес згладжування зображень в тривимірній комп'ютерній графіці.

Предметом дослідження є алгоритми машинного навчання та методи згладжування зображень.

Практична цінність - підвищення ефективності процесу згладжування зображень.

Наукова новизна магістерської роботи полягає в використанні технологій нейронних мереж для реалізації методу згладжування зображень.

1 АЛІАСИНГ ТА ЗГЛАДЖУВАННЯ ЗОБРАЖЕНЬ

1.1 Аліасинг

Аліасинг - це явище, яке ускладнює ідентифікацію різних сигналів при їх дискретизації в обробці сигналів та суміжних галузях. Він також часто відноситься до спотворення або утворенню артефактів, коли безперервний сигнал замінюється сигналом, який був дискретизований.

Аліасинг може виникати в сигналах з часовою дискретизацією, таких як цифрове аудіо, і називається часовим аліасингом. Також аліасинг може виникати в просторових сигналах (наприклад, муарові візерунки на цифрових зображеннях). Такий тип аліасингу відомий як просторовий аліасинг.

Зазвичай аліасингу уникають шляхом застосування фільтрів згладжування до вхідного сигналу перед дискретизацією і при перетворенні сигналу з більш високої на більш низьку частоту дискретизації.

Просторове згладжування - це техніка, що використовується в цифровій обробці сигналів для зменшення артефактів аліасинг при представленні зображення з високою роздільною здатністю в більш низькій роздільній здатності. Згладжування використовується в різних галузях, включаючи цифрову фотографію, комп'ютерну графіку, цифрове аудіо та багато інших.

Згладжування - це процес видалення компонентів сигналу, які мають більш високу частоту, ніж пристрій запису (або дискретизації) може належним чином відобразити.

На рис. 1.1 показано приклад зображення без згладжування і згладженого зображення. Зображення без згладжування з клітинками, що зменшуються, важко розпізнати і воно виглядає непривабливо. При згладжуванні клітинки у верхній частині зображення зливаються з сірим кольором, що, як правило, є бажаним ефектом, коли роздільна здатність недостатня для відображення деталей. Навіть у нижній частині зображення згладжене зображення має набагато плавніші края. Оскільки простір зайнятий як чорними, так і білими плитками, згладжування

інтерполює яскравість пікселів на межах, щоб отримати сірі пікселі при збільшенні. Це сприяє тому, що зображення зі згладжуванням виглядає набагато більш згладженим, ніж оригінал.

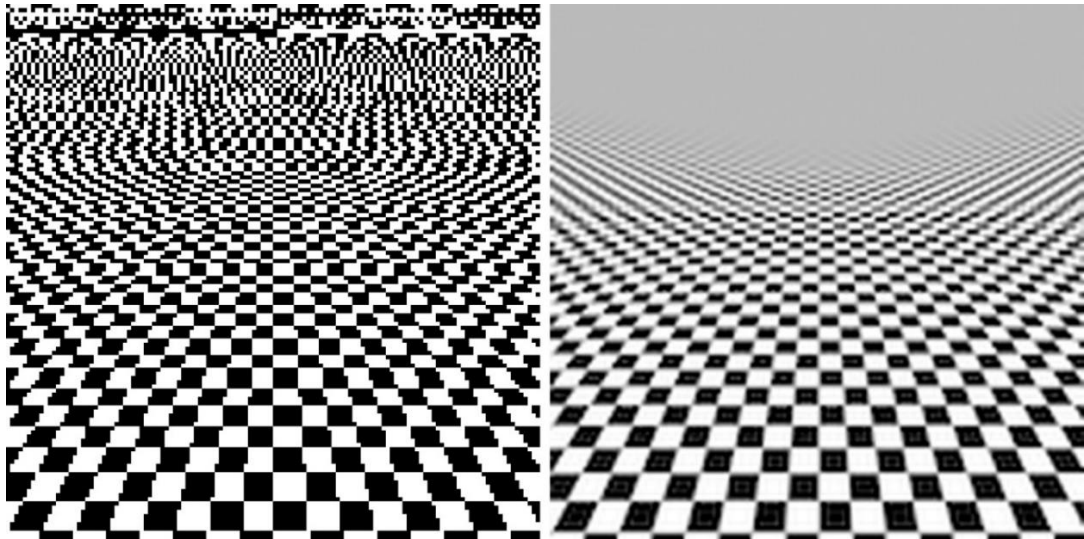


Рисунок 1.1 - Приклад зображення без згладжування і згладженого зображення

1.2 Згладжування зображень в комп'ютерній графіці

Під час генерації цифрового зображення за допомогою комп'ютерної графіки на зображенні можуть утворюватися різні артефакти через які зображення стає нерозбірливим і виглядає неестетично. Ефект аліасингу в комп'ютерній графіці — це поява нерівних країв, нерівностей або інших артефактів у растеризованому зображенні (зображенні, відтвореному за допомогою пікселів). З математичної точки зору ці артефакти утворюються, коли безперервний сигнал перетворюється на дискретний набір значень за допомогою вибірки. Растеризація прямої або кривої лінії спричиняє просторовий аліасинг — ці геометричні фігури фактично складаються з нескінченної кількості точок між двома точками в просторі, і представлення їх за допомогою фіксованої кількості пікселів завжди призведе до апроксимації цієї лінії, незалежно від того, як багато використовується пікселів. На рис 1.2 показано приклад алгоритму растеризації трикутника.

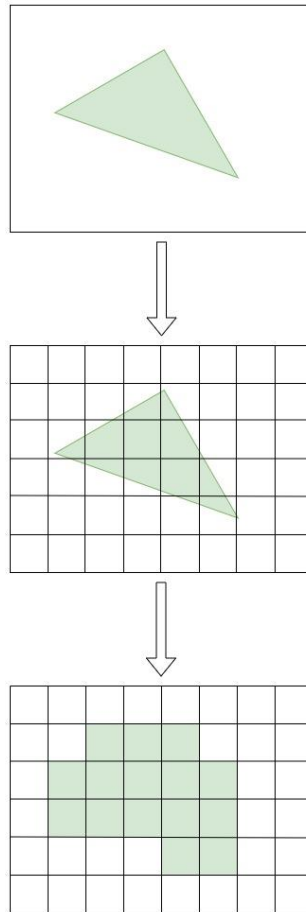


Рисунок 1.2 - Приклад алгоритму растеризації трикутника

Артефакти на зображенні можуть виникати не тільки під час растеризації ліній, а взагалі в будь-якій частині зображення де є контрастні контури. Наприклад такі артефакти можуть утворюватися при використанні контрастних текстур, або текстур, що при використанні були зменшені в розмірі. Згладжування зображень в комп'ютерній графіці — це техніка, яка використовується для усунення всіх подібних артефактів.

1.3 Висновки до розділу

В даному розділі було подано опис предметної області роботи, а саме описано суть проблеми ефекту аліасингу в зображеннях. Описано процес растеризації зображень в комп'ютерній графіці і показано в чому полягає необхідності згладжування зображень.

2 ШТУЧНИЙ ІНТЕЛЕКТ ТА НЕЙРОННІ МЕРЕЖІ

2.1 Штучний інтелект

Штучний інтелект - це інтелект який сприймає, синтезує та формулює інформацію, що демонструється машинами, на відміну від інтелекту, який демонструють тварини та люди. Приклади завдань, в яких він застосовується, включають розпізнавання мови, комп'ютерний зір, переклад між (природними) мовами, а також інші відображення вхідних даних.

Застосування штучного інтелекту включає вдосконалені пошукові системи в інтернеті, рекомендаційні системи, інтерпретацію людської мови, самокеровані автомобілі, автоматизоване прийняття рішень і гру на найвищому рівні в стратегічних ігрових системах. В міру того, як роботи стають більш досвідченими, дії, які вважаються такими, що потребують "інтелекту", часто виключаються з концепції штучного інтелекту - це явище відоме під назвою "ефект штучного інтелекту". Наприклад, хоча оптичне розпізнавання символів і є поширеною технологією, воно, як правило, не включається до поняття штучного інтелекту.

2.2 Нейронні мережі

Нейронні мережі, підмножина машинного навчання, що лежить в основі алгоритмів глибокого навчання, також їх називають штучними нейронними мережами або імітаційними нейронними мережами.

Комп'ютери можуть використовувати їх для побудови адаптивної системи, яка допомагає їм постійно вдосконалюватися, навчаючись на своїх невдачах. В результаті штучні нейронні мережі намагаються вирішувати такі складні завдання, як узагальнення документів або ідентифікація осіб.

Штучна нейронна мережа - це техніка обробки інформації. Вона працює подібно до того, як обробляє інформацію людський мозок. Вона включає в себе велику кількість з'єднаних між собою блоків, які працюють разом для обробки інформації. Вони також генерують значущі результати.

Штучні нейронні мережі призначені для відтворення поведінки нейронних мереж, знайдених в мозку людини або тварин. Віддзеркалюючи та моделюючи поведінку нейронів, машинне навчання отримує архітектуру моделі для обробки все більш складних даних. Існує безліч різних типів штучних нейронних мереж.

Складні штучні нейронні мережі розробляються таким чином, щоб моделі могли відображати нелінійний процес прийняття рішень людським мозком. Це означає, що моделі можна навчити приймати складні рішення або розуміти абстрактні поняття та об'єкти. Модель буде будуватися від низькорівневих ознак до складних ознак, розуміючи складні концепції. Нейронні мережі можуть адаптуватися до зміни вхідних даних; таким чином, мережа генерує найкращий можливий результат без необхідності переробляти вихідні критерії.

Нейронні мережі покладаються на навчальні дані, щоб навчатися і підвищувати свою точність з часом. Однак, належним чином оптимізовані, ці алгоритми навчання стають грізними інструментами в галузі інформатики та штучного інтелекту, дозволяючи здійснювати високошвидкісну класифікацію та кластеризацію даних. У порівнянні з ручною ідентифікацією фахівцями-людьми, завдання з розпізнавання мови або розпізнавання зображень можуть бути виконані за лічені хвилини, а не години.

Нейронні мережі можуть працювати безперервно і є більш ефективними, ніж люди або більш прості аналітичні моделі. Нейронні мережі також можуть бути запрограмовані на навчання на основі попередніх результатів для визначення майбутніх результатів на основі схожості з попередніми вхідними даними.

Нейронні мережі, які використовують хмару онлайн-сервісів, також мають перевагу у зменшенні ризиків у порівнянні з системами, які покладаються на локальне технологічне обладнання. Крім того, нейронні мережі часто можуть виконувати кілька завдань одночасно. Нарешті, нейронні мережі постійно розширюються в нових сферах застосування. Якщо раніше теоретичні нейронні мережі були дуже обмежені в застосуванні до різних галузей, то сьогодні нейронні мережі використовуються в медицині, науці, фінансах, сільському господарстві та безпеці.

Хоча нейронні мережі можуть покладатися на онлайн-платформи, все ще існує апаратний компонент, необхідний для створення нейронної мережі. Це створює фізичний ризик для мережі, яка покладається на складні системи, вимоги до налаштування та потенційне фізичне обслуговування.

Хоча складність нейронних мереж є сильною стороною, це може означати, що на розробку конкретного алгоритму для конкретного завдання можуть піти місяці. Крім того, може бути важко виявити будь-які помилки або недоліки в процесі, особливо якщо результати є оцінками або теоретичними діапазонами.

Нейронні мережі також можуть бути складними для аудиту. Деякі процеси нейронних мереж можуть нагадувати "чорний ящик", в який вводяться вхідні дані, мережі виконують складні процеси, а вихідні дані подаються у вигляді звітів. Окремим особам також може бути важко проаналізувати слабкі місця в процесі обчислення або навчання мережі, якщо мережі не вистачає загальної прозорості щодо того, як модель навчається на основі попередньої діяльності.

Переваги нейронних мереж:

- Часто може працювати ефективніше і довше, ніж людина.
- Можуть бути запрограмовані вчитися на попередніх результатах, щоб намагатися робити більш розумні розрахунки в майбутньому.
- Часто використовують онлайн-сервіси, які зменшують (але не усувають) систематичний ризик.
- Нарешті, нейронні мережі постійно розширюються в нових сферах застосування. Якщо раніше теоретичні нейронні мережі були дуже обмежені в застосуванні до різних галузей, то сьогодні нейронні мережі використовуються в медицині, науці, фінансах, сільському господарстві та безпеці.
- Нейронні мережі часто можуть виконувати кілька завдань одночасно (або, принаймні, розподіляти завдання, які мають виконуватися модульними мережами одночасно).

Недоліки нейронних мереж:

- Все ще покладаються на апаратне забезпечення, для обслуговування якого може знадобитися робоча сила та досвід.
- Розробка коду та алгоритмів може зайняти тривалий час.
- Може бути важко оцінити помилки або адаптацію до припущень, якщо система самонавчається, але їй бракує прозорості.
- Нейронні мережі також можуть бути складними для аудиту.
- Зазвичай повідомляється оціночний діапазон або оціночна сума, яка може не відповідати дійсності.

2.3 Історія розвитку нейронних мереж

Історія нейронних мереж довша, ніж думає більшість людей. У 1943 році Уоррен Маккалох та Уолтер Піттс з Університету Іллінойсу та Чиказького університету опублікували роботу "Логічне обчислення ідей, іманентних нервовій діяльності". У дослідженні проаналізовано, як мозок може створювати складні структури і як їх можна спростити до бінарної логічної структури, що містить лише зв'язки "істина/хибність".

Френк Розенблат з Корнельської аерокосмічної лабораторії в 1958 році розробив перцептон. Його дослідження "Перцептрон: Імовірнісна модель для зберігання і організації інформації в мозку" додало ваги роботі МакКоллоха і Пітта, і Розенблат використовував свою роботу, щоб продемонструвати, як комп'ютер може використовувати нейронні мережі для виявлення уявлень і робити висновки. Розенблату вдалося навчити комп'ютер розрізняти карти, позначені ліворуч, і карти, позначені праворуч.

Хоча численні дослідники внесли свій вклад в ідею зворотного поширення, Пол Вербос був першою людиною в США, яка відзначила його застосування в нейронних мережах в рамках своєї докторської дисертації.

Джон Хопфілд у 1982 році представив роботу про рекурентні нейронні мережі - "Hopfield Net", присвячену рекурентним нейронним мережам. Крім того, з'явилася

концепція зворотного поширення, і багато дослідників почали розуміти її потенціал для нейронних мереж.

У 1989 році Ян Лекун представив роботу, в якій продемонстрував, як обмеження зворотного розповсюдження та їх включення в дизайн нейронної мережі можуть бути використані для навчання алгоритмів. У цьому дослідженні нейронна мережа була успішно застосована для виявлення рукописних цифр поштового індексу, наданого Поштовою службою США.

Останнім часом створюються більш конкретні проекти нейронних мереж для конкретних цілей. Ці типи технологій використовуються для відкриття нових ліків, аналізу тенденцій на фінансових ринках і виконання великих наукових розрахунків.

2.4 Архітектура нейронних мереж

Нейронні мережі - це комплексні структури, що складаються зі штучних нейронів, які можуть отримувати кілька вхідних даних і генерувати один кінцевий результат. Основна функція нейронної мережі полягає в перетворенні вхідних даних у значущий результат. Нейронна мережа зазвичай складається з вхідного і вихідного шарів, а також одного або декількох прихованих шарів.

Нейронна мережа - це орієнтований граф. Він складається з вузлів, які за біологічною аналогією є нейронами. Штучні нейрони або вузли моделюються як спрощена версія нейронів, що знаходяться в мозку.

Вузол - це просто місце, де відбуваються обчислення, схоже на нейрон в людському мозку, який спрацьовує, коли він стикається з достатніми стимулами. Вузол поєднує вхідні дані з набором коефіцієнтів, або ваг, які або підсилюють, або послаблюють цей вхід, тим самим призначаючи значущість вхідних даних щодо завдання, яке алгоритм намагається вивчити.

Добутки вагових коефіцієнтів підсумовуються, а потім сума пропускається через так звану функцію активації вузла, щоб визначити, чи повинен цей сигнал просуватися далі через мережу, щоб вплинути на кінцевий результат, і якщо так, то в якій мірі. На рис 2.1 показано схему того, як може виглядати один вузол.

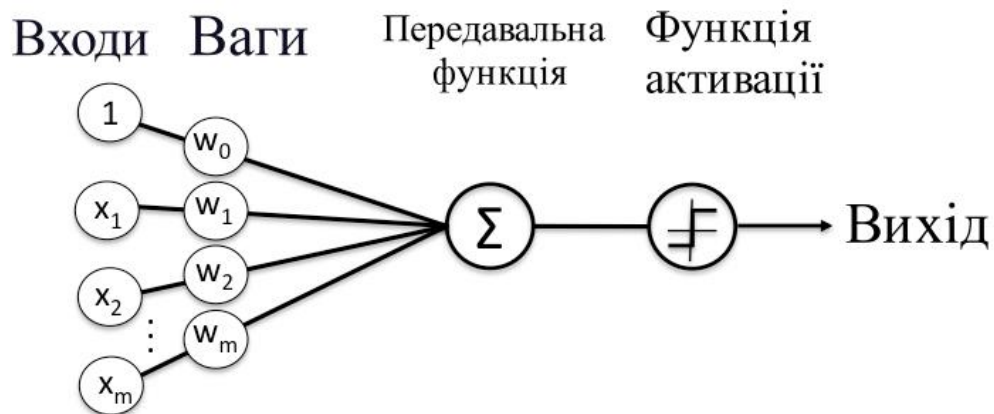


Рисунок 2.1 – Схема нейрону мережі

Отже, кожен нейрон в нейронній мережі складається з ключових компонентів:

- Вхідні дані - це набір ознак, що надаються моделі на етапі навчання. Наприклад, вхідними даними для виявлення об'єктів може бути масив значень пікселів зображення.
- Ваги - їх головна роль полягає в наданні більшої ваги тим елементам, які роблять більший вклад в навчання. Це здійснюється за рахунок використання скалярного множення вхідного значення та матриці ваг. Негативне слово, приміром, матиме сильніший вплив на вибір моделі для аналізу настрою, ніж два нейтральні терміни.
- Передавальна функція - завданням передавальної функції є інтеграція декількох входів в одне вихідне значення, що дозволяє застосувати функцію активації. Це досягається простим додаванням всіх входів до передавальної функції.
- Функція активації - вводить нелінійність у функціонування перцептронів, щоб пристосуватись до різної лінійності вхідних даних. Без цього результатом була б просто лінійна комбінація вхідних значень, без можливості включення нелінійності в мережу.

- Відхилення - роль відхилення полягає у зміщенні значення, що виробляється функцією активації. Його роль подібна до ролі константи в лінійній функції.

Кожен штучний нейрон з'єднаний з іншими вузлами, хоча щільність і кількість зв'язків відрізняються для кожного типу штучної нейронної мережі. Мережа зазвичай групується в шари вузлів, які існують між вхідним і вихідним шарами. Ці різні шари в моделях штучних нейронних мереж можуть вивчати різні особливості даних. Приховані ієрархічні шари дозволяють зрозуміти складні концепції або закономірності з оброблених даних.

Структура штучних нейронних мереж являє собою спрощене відображення складності людського або тваринного мозку. Павутина взаємопов'язаних штучних вузлів імітує поведінку нейронів у нервовій системі. Ці штучні нейронні мережі набагато менш складні, ніж людський мозок, але все ще неймовірно потужні при виконанні таких завдань, як класифікація. Дані починаються у вхідному шарі і виходять з вихідного шару. Але з більш складними штучними нейронними мережами дані будуть рухатися між багатьма різними шарами нелінійним чином.

Як і інші моделі машинного навчання, оптимізація штучних нейронних мереж базується на функції втрат. Функції втрат - різниця між прогнозованим і фактичним виходом. Вага кожного вузла та шару коригується моделлю для досягнення мінімальних втрат. Моделі штучних нейронних мереж можуть розуміти багаторівневі характеристики даних і будь-які ієрархічні відносини між характеристиками. Таким чином, при використанні для проблеми класифікації, модель штучної нейронної мережі може розуміти складні концепції, обробляючи кілька шарів ознак.

Кожна нейронна мережа складається з ключових компонентів:

- Вхідний шар. Дані, які подаються на модель, завантажуються у вхідний шар із зовнішніх джерел. Це єдиний видимий шар у всій архітектурі нейронної мережі, який передає повну інформацію із зовнішнього світу без будь-яких обчислень. Функція вхідного шару полягає в отриманні вхідних значень пояснюючих ознак для кожного спостереження. Кількість вхідних вузлів у

вхідному шарі зазвичай дорівнює кількості пояснювальних змінних. «Вхідний шар» відображає закономірності мережі, яка пов'язана з одним або декількома «прихованими шарами». Вузли вхідного шару є пасивними, тобто вони не змінюють дані. Вони приймають одне значення на вхід і копіюють його на свої численні виходи. Кожне значення з вхідного шару тиражується і передається до всіх прихованих вузлів.

- **Приховані шари.** Приховані шари - це те, що робить машинне навчання тим, чим воно є сьогодні. Це проміжні шари, які виконують всі обчислення і витягують особливості з даних. Прихованих шарів може бути декілька взаємопов'язаних між собою, які відповідають за пошук різних прихованих особливостей в даних. Наприклад, при обробці зображень перші приховані шари відповідають за ознаки більш високого рівня, такі як краї, форми або межі. З іншого боку, пізніші приховані шари виконують більш складні завдання, такі як ідентифікація цілісних об'єктів (автомобіль, будівля, людина). Приховані шари застосовують задані перетворення до вхідних значень всередині мережі. При цьому до кожного вузла підключаються вхідні з'єднання, які йдуть від інших прихованих вузлів або від вхідних вузлів. У прихованому шарі фактична обробка здійснюється через систему зважених з'єднань. Може бути один або декілька прихованих шарів. Значення, що надходять до прихованого вузла, множаться на ваги - набір заздалегідь визначених чисел, що зберігаються в програмі. Потім зважені вхідні дані додаються для отримання єдиного числа.
- **Вихідний шар.** На основі навчання моделі вихідний шар збирає вхідні дані з попередніх прихованих шарів і робить остаточний прогноз. Це найвідповідальніший шар, оскільки саме тут отримується кінцевий результат. У задачах класифікації, як правило, є тільки один вихідний вузол. Активні вузли вихідного шару об'єднують і змінюють дані для отримання вихідних значень. У випадку моделей класифікації/регресії вихідний шар, як правило, має один вузол. Однак він повністю специфічний для конкретної проблеми і залежить від способу побудови моделі.

Приклад структури нейронної мережі з декількома шарами можна побачити на рис 2.2.

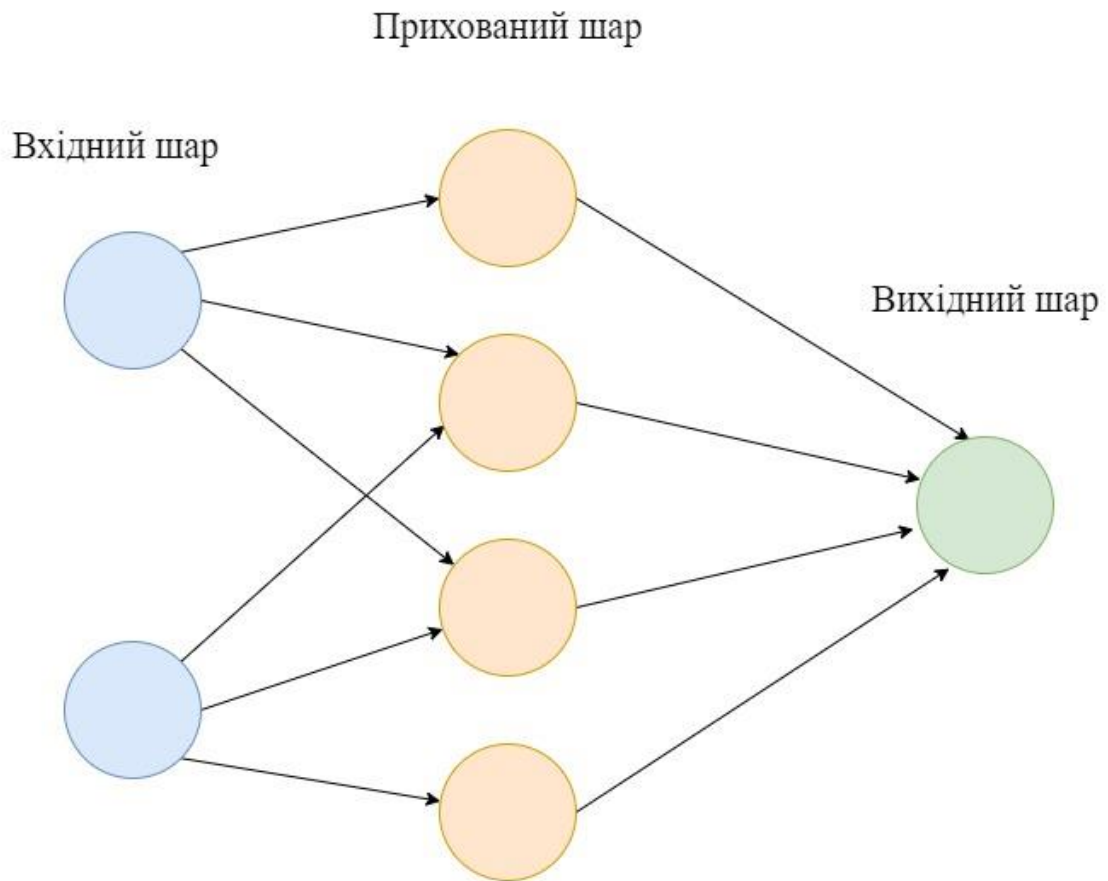


Рисунок 2.2 – Схема структури нейронної мережі

Кожен нейрон отримує помножену версію вхідних даних і випадкових ваг, до яких потім додається статичне значення зсуву (унікальне для кожного нейронного шару), що потім передається відповідній функції активації, яка визначає кінцеве значення, що буде видано нейроном. Існують різні функції активації, доступні відповідно до характеру вхідних значень. Після того, як вихідні дані отримані з останнього шару нейронної мережі, обчислюється функція втрат (вхідні дані по відношенню до вихідних) і виконується зворотне розповсюдження, де ваги підбираються таким чином, щоб зробити втрати мінімальними. Пошук оптимальних значень вагових коефіцієнтів - це те, на чому зосереджується вся робота в цілому.

У нейронній мережі всі нейрони впливають один на одного, а отже, всі вони пов'язані між собою. Мережа може розпізнавати і спостерігати за кожним аспектом наявного набору даних і за тим, як різні частини даних можуть або не можуть бути пов'язані між собою. Саме так нейронні мережі здатні знаходити надзвичайно складні закономірності у величезних обсягах даних.

У нейронній мережі потік інформації відбувається двома шляхами:

- Мережі прямого поширення: у цієї моделі сигнали рухаються тільки в одному напрямку, до вихідного шару. Мережі прямого поширення мають вхідний шар і один вихідний шар з нульовим або декількома прихованими шарами. Вони широко використовуються в розпізнаванні образів.
- Мережі зі зворотним зв'язком: у цієї моделі рекурентні або інтерактивні мережі використовують свій внутрішній стан (пам'ять) для обробки послідовності входів. У них сигнали можуть проходити в обох напрямках через петлі (приховані шари) в мережі. Вони, як правило, використовуються в часових рядах і послідовних завданнях.

2.5 Типи нейронних мереж

Нижче наведені деякі з основних категорій видів нейронних мереж:

- Класифікація. Завдання класифікації, які вимагають маркованих наборів даних для керованого навчання, часто є тими, де нейронні мережі досягають успіху. Наприклад, нейронні мережі можуть швидко і послідовно наносити мітки, ідентифікуючи візуальні закономірності на сотнях зображень. Вони опановують мистецтво вирішення складних, заплутаних проблем за допомогою навчання. Нейронна мережа вчиться самостійно розпізнавати найбільш важливі аспекти. Таким чином, від фахівця з обробки даних не потрібно надавати ознаки для розрізнення собак і котів.
- Вивчення послідовностей. Вивчення послідовностей - це категорія машинного навчання, яка використовує послідовності даних як вхідні або

вихідні дані. Текстові потоки, аудіофайли, відеокліпи, вимірювання та багато іншого є прикладами послідовного навчання.

- Апроксимація функцій. Апроксимація функції - це метод наближення невідомої базової функції з використанням попередніх або поточних спостережень у цій області. Функцію навчаються апроксимувати штучні нейронні мережі.

Моделі штучних нейронних мереж лежать в основі багатьох найскладніших застосувань машинного навчання. Класифікація, регресійні задачі і аналіз настроїв - ось деякі з способів використання штучних нейронних мереж сьогодні. Як галузь, що розвивається, існує багато різних типів штучних нейронних мереж. Вони відрізняються з різних причин, таких як складність, архітектура мережі, щільність і потік даних. Але різні типи мають спільну мету - моделювання та спроби відтворити поведінку нейронів для покращення машинного навчання.

Штучні нейронні мережі мають широкий спектр застосування. Кожен тип моделей штучних нейронних мереж має різні сильні сторони та варіанти використання. В цілому, вони в основному використовуються для вирішення більш складних завдань, ніж це було б можливо за допомогою більш традиційних методів. Приклади можуть включати складну обробку природної мови та машинний переклад мови, які покладаються на штучні нейронні мережі. Рекурентні нейронні мережі також часто використовуються для аналізу настроїв або перекладу тексту. Глибина і масштаб нейронної архітектури означає, що можна досягти нелінійного процесу прийняття рішень. Штучні нейронні мережі використовуються в глибокому навчанні у формі машинного навчання. Це називається глибоким навчанням, оскільки моделі використовують глибоку, багат шарову архітектуру штучної нейронної мережі. Оскільки кожен шар штучної нейронної мережі може обробляти дані, моделі можуть будувати абстрактне розуміння даних. Ця архітектура означає, що моделі можуть виконувати все більш складні завдання, наприклад, розуміти природну мову або класифікувати складні типи файлів. Існує багато різних типів штучних нейронних мереж, що відрізняються за складністю. Їх об'єднує передбачувана мета дзеркального відображення функції людського мозку

для вирішення складних проблем або завдань. Структура кожного типу штучної нейронної мережі певним чином відображає нейрони і синапси. Однак вони відрізняються за рівнем складності, варіантами використання та структурою. Відмінності також включають те, як моделюються штучні нейрони в межах кожного типу штучної нейронної мережі, і зв'язки між кожним вузлом. Інші відмінності включають те, як дані можуть проходити через штучну нейронну мережу, і щільність вузлів.

Існують різні види нейронних мереж, які зараз є доступними або знаходяться на етапі розробки. Їх можна класифікувати відповідно до структури, потоку даних, кількості використовуваних нейронів та їх щільності, глибини шарів та фільтрів активації тощо. Різні типи штучних нейронних мереж служать для виконання різних функцій.

2.5.1 Перцептрон

Френк Розенблат винайшов нейронну мережу перцептрон у 1958 році. Це найпростіший тип нейронної мережі, що складається лише з одного нейрона. Однією з найпростіших і найстаріших моделей нейронів є модель перцептрона. Це найменша одиниця нейронної мережі, яка виконує обчислення з метою виявлення характеристик у вхідних даних. Він виробляє кінцевий результат, беручи зважені вхідні дані та застосовуючи функцію активації. Перцептрон - це алгоритм керованого навчання, який класифікує дані на дві категорії, тобто є бінарним класифікатором. Як лінійна модель він є одним з найпростіших прикладів типу штучної нейронної мережі.

Застосування:

- Стиснення даних: кодування, реорганізація або інша зміна даних з метою зменшення їх обсягу відома як стиснення даних. У своїй найпростішій формі воно передбачає перекодування даних з використанням меншої кількості бітів, ніж початкове представлення.

- Потоккове кодування: техніка кодування відбілює реальні вхідні дані, що надаються першим прихованим одиницям повністю підключеної нейронної мережі, що призводить до швидшого навчання.

Переваги:

- Перцептрони можуть реалізовувати логічні вентиля типу I, АБО або NAND.
- Це дає нам більш надійні підстави для прийняття рішень і покращує нашу здатність передбачати різні результати при розгляді наявних даних.

Недоліки:

- Перцептрони можуть навчатися тільки лінійно роздільним задачам, таким як задача булевого I. Для нелінійних задач, таких як задача булевого XOR, він не працює.
- Через жорстко обмежену передавальну функцію вихідні значення перцептрона можуть приймати лише одне з двох значень (0 або 1).
- Перцептрони можуть класифікувати тільки набори векторів, які можуть бути розділені лінійно.

Структуру перцептрону показано на рис 2.3.

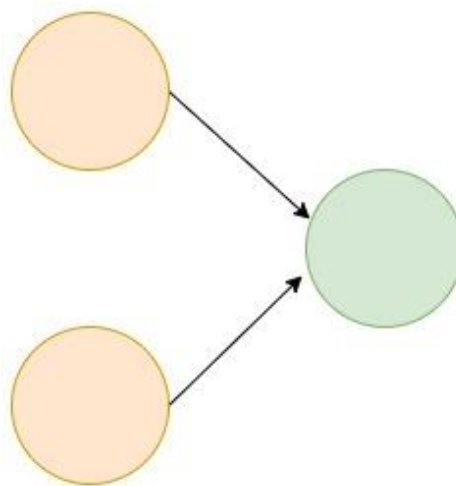


Рисунок 2.3 – Структура перцептрону

2.5.2 Багатошаровий перцептрон

Багатошаровий перцептрон є відправною точкою для складних нейронних мереж, де вхідні дані проходять через багато шарів штучних нейронів. Кожен вузол

сполучений з усіма нейронами в наступному шарі, в наслідок чого створюється повністю пов'язана нейронна мережа. Вхідний і вихідний шари включають кілька прихованих шарів, загалом щонайменше три або більше шарів. Він має двоспрямоване поширення, що означає, що вона може рухатися як вперед, так і назад. Вхідні дані множаться на ваги і подаються на функцію активації, а при зворотному поширенні вони модифікуються для зменшення втрат. Багатошарові перцептрони належать до класу нейронних мереж прямого поширення з декількома шарами персептронів, які мають функції активації. Багатошарові перцептрони складаються з вхідного та вихідного шарів, які повністю з'єднані між собою. Вони мають однакову кількість вхідних і вихідних шарів, але можуть мати кілька прихованих шарів і можуть бути використані для побудови програмного забезпечення для розпізнавання мови, розпізнавання зображень і машинного перекладу. Вони використовуються для більш складних проблем і завдань, таких як складна класифікація або розпізнавання голосу. Через глибину та складність моделі, обробка та підтримка моделі може вимагати значних ресурсів та часу.

Застосування:

- Машинний переклад: для оцінки вірогідності послідовності слів у нейронному машинному перекладі, передовому підході до машинного перекладу, використовуються методи нейронних мереж.
- Комплексна класифікація: група кількісних методів, відомих як комплексна класифікація, використовується для вивчення динаміки і структури складних мережевих систем.

Переваги:

- Використовується для глибокого навчання завдяки наявності щільних повністю пов'язаних шарів та зворотному поширенню.
- Перевага багатошарових персептронів полягає в тому, що вони можуть вивчати нелінійні моделі і навчати моделі в реальному часі.
- Він може обробляти велику кількість вхідних даних.

Недоліки:

- Порівняно складна в розробці та обслуговуванні
- Порівняно повільний (залежить від кількості прихованих шарів)

Структуру багат шарового перцептронну показано на рис 2.4.

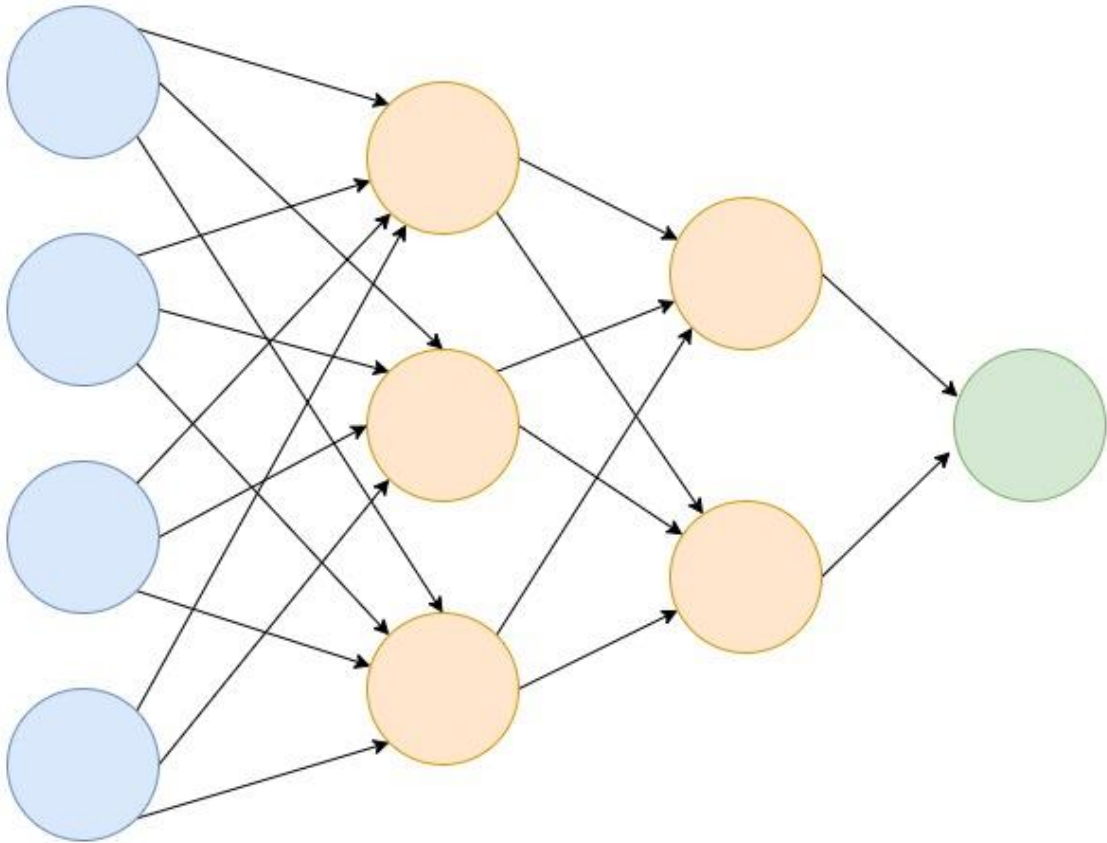


Рисунок 2.4 – Структуру багат шарового перцептронну

2.5.3 Згорткові нейронні мережі

Згорткові нейронні мережі схожі на мережі прямого поширення, але вони зазвичай використовуються для розпізнавання зображень, розпізнавання образів та/або комп'ютерного зору. Ці мережі використовують принципи лінійної алгебри, зокрема, матричне множення, для виявлення закономірностей на зображенні. Згорткові нейронні мережі, також відомі як ConvNets, складаються з декількох шарів і в основному використовуються для обробки зображень і виявлення об'єктів. Ян Лекун розробив першу згорткову нейронну мережу у 1988 році, коли вона називалася LeNet. Вона використовувалася для розпізнавання символів, таких як поштові індекси та цифри.

Замість стандартного двовимірного масиву, згорткова нейронна мережа має тривимірне розміщення нейронів. Перший шар називається згортковим. Кожен нейрон в згортковому шарі аналізує дані з крихітної частини поля зору. Мережа осмислює зображення шматок за шматком і може виконувати ці дії кілька разів, щоб закінчити всю обробку зображення. Зображення обробляється шляхом перетворення його з RGB або HSI в градації сірого. Подальші варіації значення пікселя допомагають у виявленні країв, і зображення можуть бути класифіковані на багато груп.

Поширення може бути односпрямованим (згорткова нейронна мережа має один або декілька згорткових шарів з подальшим об'єднанням) або двоспрямованим (вихід згорткового шару подається на повністю підключену нейронну мережу для категоризації зображення). Фільтри використовуються для виділення певних ділянок зображення. Згорткові нейронні мережі широко використовуються для ідентифікації супутникових знімків, обробки медичних зображень, прогнозування часових рядів та виявлення аномалій. Згорткові нейронні мережі мають декілька шарів, які обробляють та виділяють ознаки з даних:

- Шар згортки, який має декілька фільтрів для виконання операції згортки.
- Випрямлений лінійний блок (ReLU) для виконання операцій над елементами. Результатом є випрямлена карта елементів.
- Шар об'єднання. Виправлена карта елементів далі подається на шар об'єднання. Об'єднання - це операція зменшення вибірки, яка зменшує розміри карти елементів. Потім шар об'єднання перетворює отримані двовимірні масиви з об'єднаної карти особливостей в один довгий, безперервний, лінійний вектор шляхом його згладжування.
- Повністю з'єднаний шар Повністю зв'язаний шар формується, коли на вхід подається сплющена матриця з шару об'єднання, який класифікує та ідентифікує зображення.

Застосування:

- Обробка природної мови - це галузь комп'ютерних наук, пов'язана зі штучним інтелектом, яка надає комп'ютерам можливість розуміти письмові та усні слова так само, як це робить людина.
- Виявлення аномалій: Процес виявлення аномальних значень у послідовності даних відомий як виявлення аномалій.

Переваги:

- Використовується для глибокого навчання з невеликою кількістю параметрів.
- Менше параметрів для навчання в порівнянні з повністю підключеним шаром.

Недоліки:

- Порівняно складний в проектуванні та обслуговуванні
- Порівняно повільна (залежить від кількості прихованих шарів)

Структуру згорткової нейронної мережі показано на рис. 2.5.

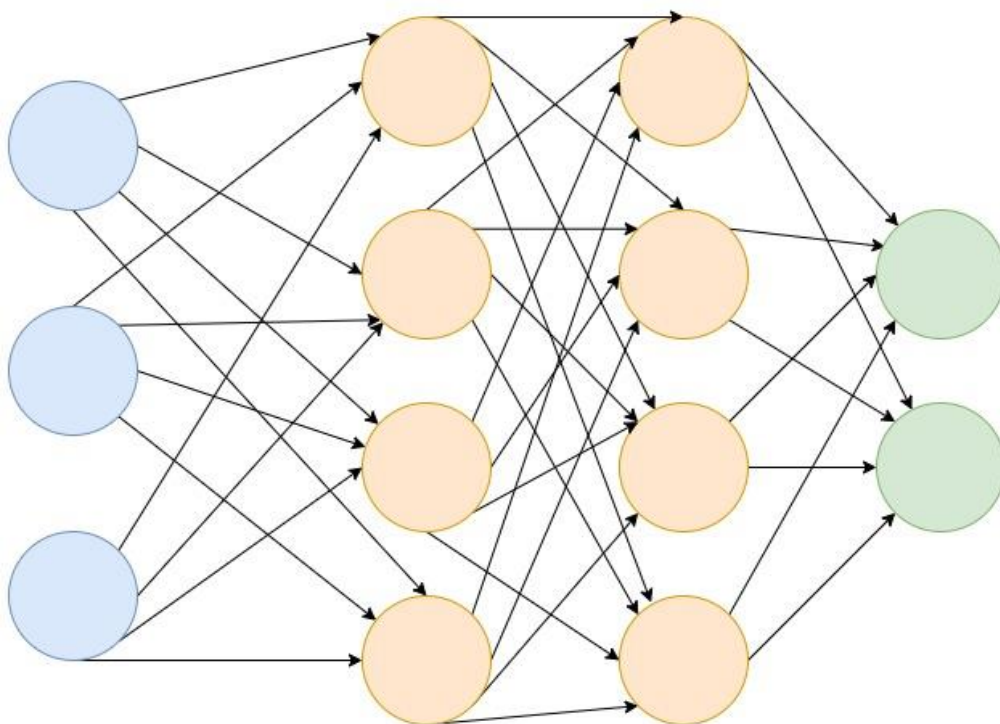


Рисунок 2.5 – Структура згорткової нейронної мережі

2.5.4 Мережа радіальних базисних функцій

Мережа радіальних базисних функцій - це спеціальні типи нейронних мереж прямого поширення, які використовують радіальні базисні функції як функції

активації. Вони мають вхідний шар, прихований шар і вихідний шар і в основному використовуються для класифікації, регресії та прогнозування часових рядів. Радіальні базисні функції обчислюють абсолютне значення між центральною точкою та заданою точкою. У випадку класифікації радіально-базисна функція обчислює відстань між вхідними даними та вивченою класифікацією. Якщо вхідні дані є найближчими до певної мітки, вони класифікуються як такі. Мережі радіальних базисних функцій широко використовуються в управлінні системами, наприклад, системами, які контролюють відновлення живлення після відключення електроенергії. Штучна нейронна мережа може зрозуміти пріоритетний порядок відновлення електропостачання, визначаючи пріоритетність ремонту для найбільшої кількості людей або основних послуг. Мережа радіальних базисних функцій складається з вхідного вектора, вихідного шару з одним вузлом для кожної категорії, шару нейронів. Процес класифікації передбачає порівняння вхідних даних з прикладами з навчальної вибірки, де кожен нейрон має прототип. Мережі радіальних базисних функцій виконують класифікацію, вимірюючи схожість вхідних даних з прикладами з навчальної бази.

Застосування:

- Апроксимація функції: апроксимація функції - це підхід для вимірювання невідомої базової функції, яка невідома, використовуючи попередні або поточні спостереження з області.
- Прогнозування часових рядів: здійснення наукових прогнозів на основі даних з історичними позначками часу відоме як прогнозування часових рядів. Воно передбачає створення моделей шляхом історичного дослідження, використання їх для отримання висновків та прийняття стратегічних рішень у майбутньому.

Переваги:

- Проектування адаптивних систем управління - чудова ідея.
- Нещодавно створений алгоритм представлений для створення невеликих мереж і проведення ефективних процедур навчання.

Недоліки:

- Через проблему з градієнтом її складно навчати.
- Проблема зникаючих градієнтів впливає на нейронну мережу.

Структуру мережі радіальних базисних функцій показано на рис. 2.6.

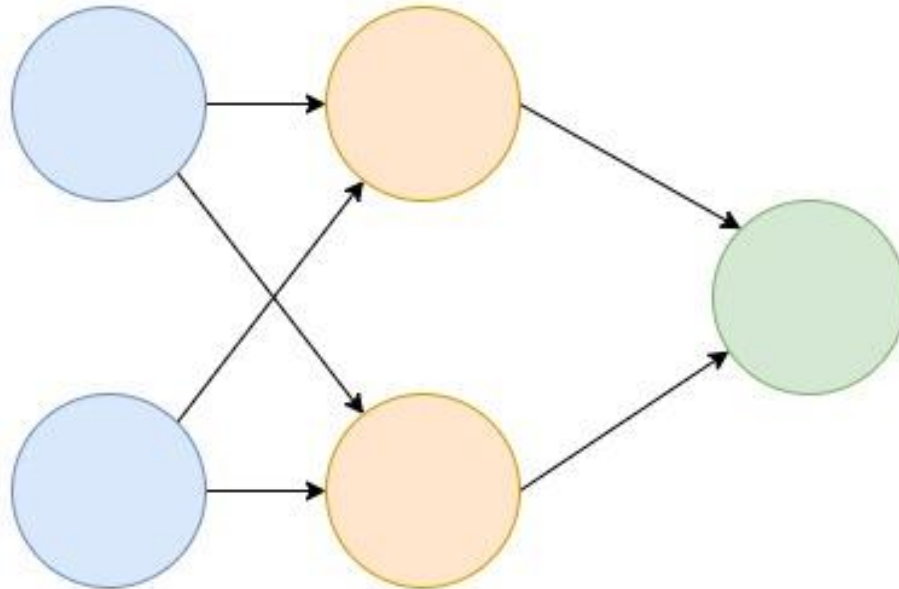


Рисунок 2.6 – Структура мережі радіальних базисних функцій

2.5.5 Рекурентні нейронні мережі

Рекурентні нейронні мережі визначають петлі зворотного зв'язку. Ці алгоритми навчання, як правило, використовуються для оцінки майбутніх подій з використанням даних часових рядів, таких як прогнози фондового ринку або прогнози продажів. Рекурентні нейронні мережі мають зв'язки, які формують спрямовані цикли, що дозволяють подавати вихідні дані як вхідні дані в поточну фазу. Рекурентні нейронні мережі широко використовуються для підпису зображень, аналізу часових рядів, обробки природної мови, розпізнавання рукописного тексту та машинного перекладу. Рекурентна нейронна мережа призначена для зберігання результату шару і відправляється назад на вхід, щоб допомогти спрогнозувати висновок шару. Першим шаром часто є нейронна мережа прямого поширення, за яким слідує рекурентний шар нейронної мережі, який запам'ятовує частину інформації, яку він мав на попередньому часовому кроці, за

допомогою функції пам'яті. У цьому сценарії використовується пряме поширення. Це зберігає інформацію, необхідну для подальшого використання. Якщо прогноз невірний, швидкість навчання використовується для внесення невеликих змін. Таким чином, при зворотному поширенні вона поступово збільшується до правильного прогнозу. Рекурентні нейронні мережі є потужним інструментом, коли модель призначена для обробки послідовних даних. Модель просуває дані вперед і повертає їх назад до попередніх кроків у штучній нейронній мережі, щоб найкращим чином досягти поставленої мети та покращити прогнози. Шари між вхідним і вихідним шарами є рекурентними, тобто відповідна інформація повертається назад і зберігається. Пам'ять про вихідні дані з шару повертається назад до входу, де вона зберігається для покращення процесу для наступного входу. Потік даних подібний до штучних нейронних мереж прямого поширення, але кожен вузол зберігає інформацію, необхідну для покращення кожного кроку. Завдяки цьому моделі можуть краще розуміти контекст вхідних даних і уточнювати прогноз виходу. Наприклад, система прогнозування тексту може використовувати пам'ять про попереднє слово в рядку слів, щоб краще передбачити результат наступного слова. Рекурентна штучна нейронна мережа буде краще підходити для розуміння настрою, що стоїть за цілим реченням, порівняно з більш традиційними моделями машинного навчання. Рекурентні нейронні мережі також використовуються в моделях «від послідовності до послідовності», які застосовуються для обробки природної мови. У цих моделях використовуються дві рекурентні нейронні мережі, які складаються з одночасного кодера і декодера. Ці моделі використовуються для реактивних чат-ботів, перекладу мови або для узагальнення документів. Рекурентні нейронні мережі будуються для розуміння часових або послідовних даних. Рекурентні нейронні мережі покращують свої прогнози, використовуючи додаткові точки даних у послідовності. Щоб змінити вихід, вони приймають вхідні дані і повторно використовують активації більш ранніх або пізніх вузлів у послідовності.

Застосування:

- Створення підписів до зображень: процес створення письмового опису зображення називається підписом до зображення.
- Прогнозування коливань на фондовому ринку: можна визначити майбутню вартість акцій підприємств та інших фінансових активів, що торгуються на біржі, використовуючи прогнозування цін на акції на основі машинного навчання.

Переваги:

- Перевагою є моделювання послідовних даних, де кожна вибірка може вважатися залежною від історичних.
- Використовується з шарами згортки для збільшення ефективності пікселів.

Недоліки:

- Градієнтні зникаючі та вибухаючі задачі.
- Навчання рекурентних нейронних мереж може бути складним завданням
- Важко обробляти довгі послідовні дані, використовуючи ReLU як функцію активації.

Структуру рекурентної нейронної мережі показано на рис. 2.7.

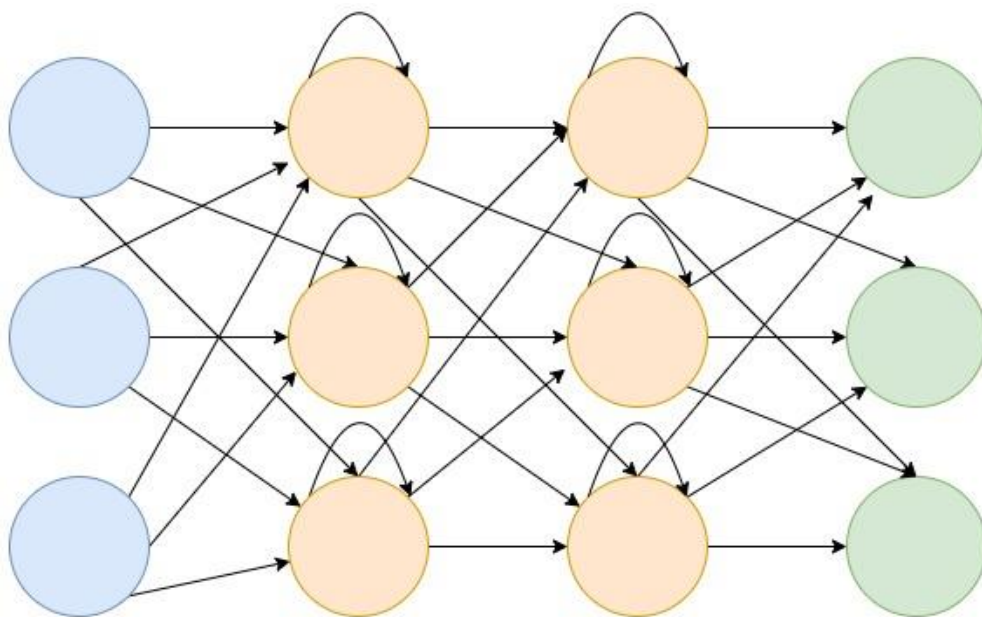


Рисунок 2.7 – Структура рекурентної нейронної мережі

2.5.6 Генеративні змагальні мережі

Генеративні змагальні мережі - це генеративні алгоритми глибокого навчання, які створюють нові екземпляри даних, що нагадують навчальні дані. Генеративні змагальні мережі мають два компоненти: генератор, який навчається генерувати фальшиві дані, та дискримінатор, який навчається на цій фальшивій інформації. Використання генеративних змагальних мереж з часом збільшилось. Вони можуть бути використані для покращення астрономічних зображень та імітації гравітаційного лінзування для дослідження темної матерії. Розробники відеоігор використовують генеративні змагальні мережі для підвищення якості 2D текстур низької роздільної здатності в старих відеоіграх, відтворюючи їх у 4K або вищій роздільній здатності за допомогою навчання зображень. Генеративні змагальні мережі допомагають генерувати реалістичні зображення та мультиплікаційних персонажів, створювати фотографії людських облич та рендерити 3D-об'єкти. Під час роботи генеративної змагальної нейронної мережі дискримінатор навчається відрізнити підроблені дані генератора від реальних даних зразка. Під час початкового навчання генератор видає фальшиві дані, і дискримінатор швидко вчиться визначати, що це неправда. Генеративна змагальна мережа надсилає результати генератору та дискримінатору для оновлення моделі. Навчання без вчителя, при якому синтетичні дані генеруються на основі закономірностей, що виявляються у вхідних даних, називається генеративним моделюванням. Генеративна мережа - це генеративна модель, яка використовує закономірності навчання для отримання нових синтетичних даних і, таким чином, є важливою темою досліджень в галузі штучного інтелекту.

Генеративні мережі суперництва - це широке сімейство дуальних мереж, які включають в себе генератор і дискримінатор. Вони безперервно прагнуть обдурити один одного: генератор виробляє певні дані, а дискримінатор, одержуючи вибірку даних, старається відокремити згенеровані дані від вибірки. Така своєрідна модель штучної нейронної мережі, яка постійно розвивається, може створювати реальні картини, якщо підтримувати баланс навчання між цими двома нейронними мережами.

Застосування:

- Генеративні змагальні мережі корисні в маркетингу, рекламі, електронній комерції, іграх, лікарнях тощо.
- Можна генерувати зображення для романів та інших матеріалів для написання історій.
- Генеративні змагальні мережі в основному використовуються для створення зображень та відео.

Переваги:

- Генеративні змагальні мережі генерують дані, які виглядають схожими на оригінальні дані. Якщо ви надаєте мережі зображення, то вона згенерує нову версію зображення, яка виглядає схожою на оригінальне зображення. Аналогічно, він може генерувати різні версії тексту, відео, аудіо.
- Генеративні змагальні мережі заглиблюються в деталі даних і можуть легко інтерпретувати в різні версії, тому вони корисні для машинного навчання.

Недоліки:

- Складніше навчати: потрібно постійно надавати різні типи даних, щоб перевірити, чи працює вона точно чи ні.
- Генерування результатів з тексту або мови є дуже складним.

Структуру генеративної змагальної мережі показано на рис. 2.8.

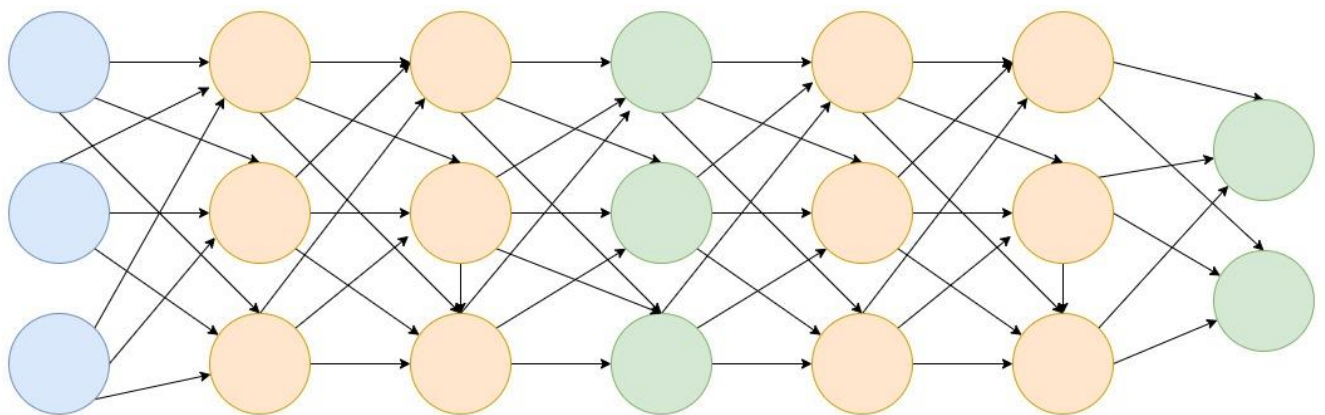


Рисунок 2.8 – Структура генеративної змагальної мережі

2.5.7 Автоматичні кодувальники

Автоматичні кодувальники - це особливий тип нейронних мереж прямого поширення, в яких вхідні та вихідні дані ідентичні. Джеффри Хінтон розробив автокодеру у 1980-х роках для вирішення проблем неконтрольованого навчання. Вони являють собою навчені нейронні мережі, які реплікують дані з вхідного шару на вихідний. Автоматичні кодувальники використовуються для таких цілей, як фармацевтичне відкриття, прогнозування популярності та обробка зображень. Автоматичний кодувальник складається з трьох основних компонентів: кодувальника, коду та декодувальника. Автоматичні кодувальники структуровані таким чином, щоб отримувати вхідні дані і перетворювати їх в інше представлення. Потім вони намагаються якомога точніше реконструювати оригінальний вхідний сигнал. Коли зображення цифри не є чітко видимим, воно подається на нейронну мережу автоматичного кодувальника. Автоматичні кодувальники спочатку кодують зображення, потім зменшують розмір вхідних даних до меншого представлення. Нарешті, автоматичний кодувальник декодує зображення, щоб генерувати реконструйоване зображення. Автоматичні кодувальники використовуються для класифікації, кластеризації та стиснення ознак.

Застосування:

- Усунення складності
- Проблеми неконтрольованого навчання
- Виявлення аномалій

Переваги:

- Використання автоматичних кодувальників дає нам перевагу за рахунок зменшення розмірності даних, які використовуються, а також скорочення часу навчання.
- Компактність і швидкість кодування з використанням зворотного поширення.

Недоліки:

- Багато даних, часу на обробку, налаштування гіперпараметрів та перевірка моделі.

- Автоматичний кодувальник вчиться збирати якомога більше інформації, а не якомога більше релевантної інформації.

Структуру автоматичного кодувальника показано на рис. 2.9.

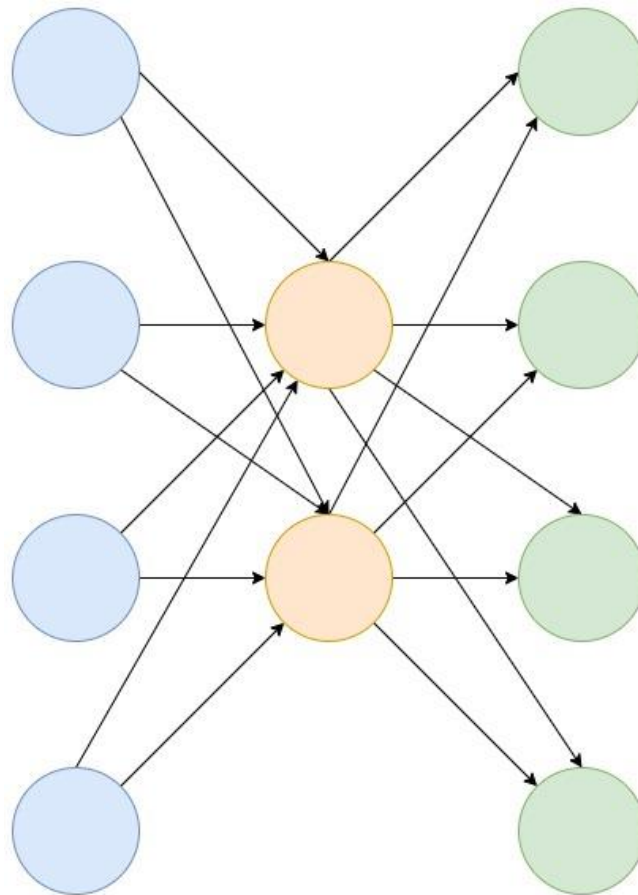


Рисунок 2.9 – Структура автоматичного кодувальника

2.6 Способи навчання нейронних мереж

Нейронні мережі навчаються шляхом аналізу прикладів, кожен з яких має відомий «ввід» і «вивід», створюючи зважені за ймовірністю зв'язки між ними, які зберігаються в структурі даних мережі. Навчання нейронної мережі часто здійснюється шляхом обчислення різниці між обробленим виходом мережі (часто прогнозом) і цільовим виходом. У нейронній мережі процес навчання (або тренування) починається з поділу даних на три різні набори:

- Навчальний набір даних - цей набір даних дозволяє нейронній мережі зрозуміти ваги між вузлами.
- Перевірочний набір даних - цей набір даних використовується для точного налаштування продуктивності нейронної мережі.
- Тестовий набір даних - цей набір даних використовується для визначення точності та похибки нейронної мережі.

Після того, як дані сегментовані на ці три частини, до них застосовуються алгоритми нейронної мережі для навчання нейронної мережі. Процедура, що використовується для покращення процесу навчання нейронної мережі, відома як оптимізація, а алгоритм, що використовується, називається оптимізатором. Існують різні типи алгоритмів оптимізації, кожен з яких має свої унікальні характеристики та аспекти, такі як вимоги до пам'яті, чисельна точність та швидкість обробки.

2.6.1 Навчання з учителем

Навчання з учителем - це парадигма машинного навчання для ситуацій, коли наявні дані складаються з маркованих екземплярів, тобто кожна точка даних містить характеристики (коваріати) та пов'язану з ними мітку. Метою алгоритмів навчання з учителем є розробка функції, яка переводить вектори ознак (входи) у мітки (виходи) на основі прикладів пар введення-виведення. Алгоритм виводить функцію з маркованих навчальних даних, які складаються з набору навчальних прикладів. Кожен приклад у керованому навчанні - це пара, що складається з вхідного елемента (зазвичай вектора) і бажаного вихідного значення (керуючим сигналом). Алгоритм навчання з учителем досліджує навчальні дані і генерує функцію виведення, яка може бути використана для відображення нових зразків. В ідеальному сценарії алгоритм буде здатний точно передбачати мітки класів для невідомих прикладів. Для цього необхідно, щоб алгоритм навчання "розумно" узагальнював навчальні дані на раніше зустрінуті сценарії. Для оцінки статистичної якості алгоритму використовується так звана помилка узагальнення.

2.6.2 Навчання без вчителя

Навчання без вчителя - це алгоритм, який вивчає закономірності на основі немаркованих даних. Мета полягає в тому, що за допомогою імітації, яка є основним способом навчання у людини, машина буде спонукатися до розробки компактного опису свого оточення, а потім генерувати з нього уявний матеріал. На відміну від навчання з учителем, в якому дані маркуються експертом, підходи навчання без вчителя демонструють здатність до самостійної роботи, яка фіксує закономірності у вигляді щільності ймовірності або суміші параметрів нейронних функцій, що зберігаються у вагах і активаціях системи. Інші ступені контролю включають навчання з підкріпленням, в якому машині дається лише числовий показник продуктивності в якості керівництва, і напів контрольоване навчання, в якому лише невеликий відсоток даних маркується. Під час фази навчання мережа без вчителя намагається імітувати дані, які їй надаються, і використовує помилку в імітованому результаті, щоб виправити себе (тобто виправити свої ваги та упередження). Помилка може бути заявлена як низька ймовірність того, що помилковий результат відбудеться, або вона може бути виражена як нестабільний високоенергетичний стан в мережі.

2.6.3 Напіваавтоматичне навчання

Напіваавтоматичне навчання - це підгалузь машинного навчання, в якій використовуються зашумлені, обмежені або неточні джерела для подачі сигналів контролю для категоризації великих обсягів навчальних даних у сценарії навчання з учителем. Цей метод полегшує тягар отримання наборів даних з ручним маркуванням, що може бути дорогим або складним. Замість цього використовуються доступні слабкі мітки з усвідомленням того, що вони неточні, але все одно можуть бути використані для створення потужної моделі прогнозування.

2.6.4 Навчання з підкріпленням

Навчання з підкріпленням - це підгалузь машинного навчання, яка розглядає, як інтелектуальні агенти повинні поводитися в певному середовищі, щоб

максимізувати концепцію кумулятивної винагороди. Навчання з підкріпленням є однією з трьох фундаментальних парадигм машинного навчання, поряд з навчанням з учителем і навчанням без учителя. Навчання з підкріпленням відмінне від навчання з учителем тим, що воно не вимагає наявності маркованих пар даних або явної корекції небажаної поведінки. Замість цього акцент робиться на досягненні балансу між розвідкою не досліджуваного регіону та експлуатацією наявних знань. Метою навчання з підкріпленням є навчання агента оптимальній або майже оптимальній стратегії, яка максимізує "функцію винагороди" або інший наданий користувачем сигнал підкріплення, який накопичується з безпосередніх винагород. Це схоже на механізми, що спостерігаються в психології тварин. Біологічний мозок, наприклад, призначений для сприйняття таких сигналів, як біль і голод, як негативне підкріплення, а задоволення і споживання їжі - як позитивне підкріплення. Тварини можуть навчитися брати участь у діяльності, яка максимізує ці винагороди в певних умовах. Це означає, що тварини можуть навчатися через підкріплення.

2.6 Висновки до розділу

В даному розділі було подано опис концепцій з теорії штучного інтелекту, штучних нейронних мереж. Було досліджено історію розвитку штучних нейронних мереж. Було проаналізовано структуру, ключові компоненти і алгоритми роботи нейронних мереж. Було досліджено різноманітні архітектури нейронних мереж та були представлені їх переваги і недоліки. Також було досліджено структуру штучного нейрона та різні методики навчання нейронних мереж.

3 АНАЛІЗ ІСНУЮЧИХ МЕТОДІВ

3.1 Згладжування за допомогою суперсемплінгу

Згладжування за допомогою суперсемплінгу (SSAA) — метод, що є найстарішим, найпростішим і найбільш ефективним з усіх. Він полягає в тому, що генерація зображення спочатку виконується з роздільною здатністю, вищою за цільову, а потім отримане зображення зменшується до цільової роздільної здатності за допомогою вибірка та змішування. Змішування зазвичай відбувається за допомогою простого пошуку середнього арифметичного значення сусідніх пікселів. Зображення нижче демонструє застосування класичного 4x SSAA в дії. Під 4x мається на увазі те, що 4 зразки (пікселі) змішуються разом шляхом обчислення середнього арифметичного значень кольорів для кінцевого результату. Для цього роздільність зображення під час генерації збільшується в 2 рази по обох осях. Діаграма роботи методу згладжування за допомогою суперсемплінгу представлена на рис. 3.1.

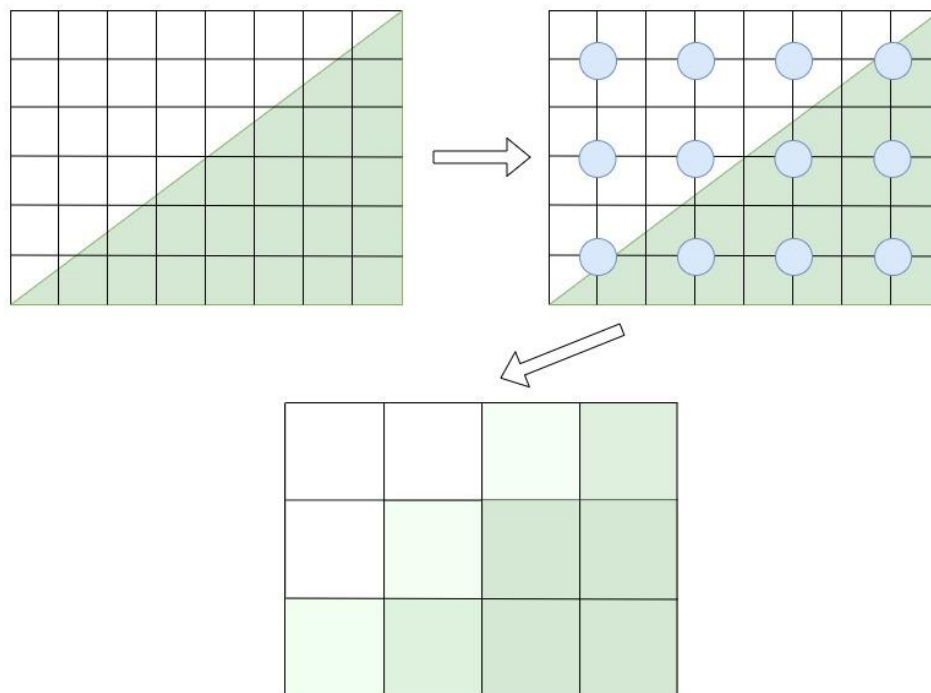


Рисунок 3.1 – Діаграма роботи методу згладжування за допомогою суперсемплінгу

Метод змішування, описаний вище, відомий як коробковий фільтр. Головним недоліком SSAA є те, що після збільшення роздільної здатності всі додаткові пікселі потрібно обробити, що в свою чергу потребує багато часу і може легко призвести до різкого падіння частоти кадрів. Саме тому з часом SSAA було замінено на більш швидкі алгоритми.

3.2 Згладжування з кількома вибірками

Згладжування з кількома вибірками (MSAA) — метод що працює за тим же самим принципом що і SSAA, але на відміну від суперсемплінгу, замість того щоб збільшувати роздільну здатність всього зображення цей алгоритм застосовується лише там, де це дійсно потрібно. MSAA виявляє краї полігонів і лише збільшує кількість вибірок на краях. Межі трикутників відображаються не як один піксель, а як усереднене значення суми значень декількох пікселів. Робиться вибірка з певної кількості пікселів, і знаходиться остаточний колір пікселя. Діаграма роботи методу згладжування з кількома вибірками представлена на рис. 3.2.

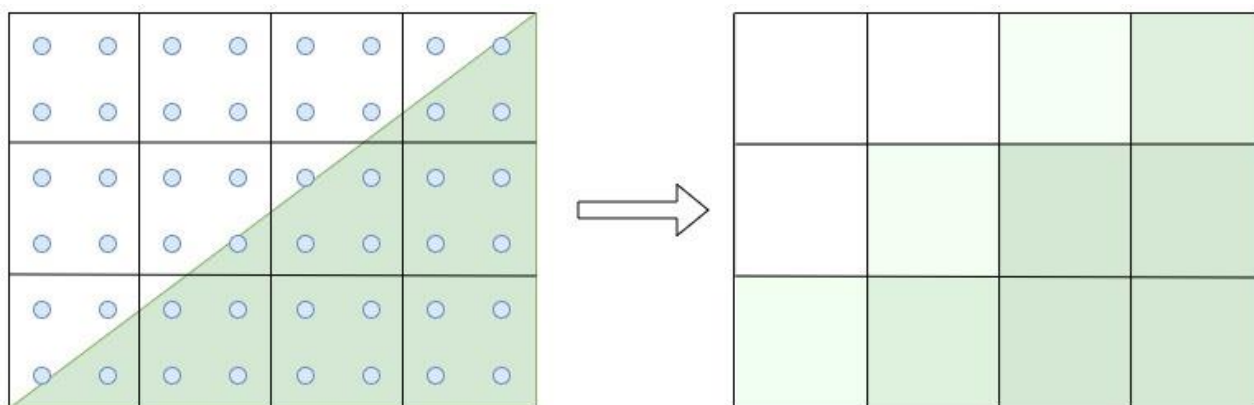


Рисунок 3.2 – Діаграма роботи методу згладжування з кількома вибірками

Цей алгоритм працює досить швидко і він не такий вимогливий, як SSAA, оскільки суперсемплінг відбувається лише на межах полігонів.

Проте це і є одним із основних недоліків цього алгоритму. Він добре працює лише

на межах багатокутників і тому не усуває артефакти, що виникають на текстурах в середині полігонів. Також цей метод згладжування добре працює лише зі статичними зображеннями тому, що рух робить різницю між згладженими лініями та розмитими текстурами трохи помітнішою. Ще одна проблема цього методу полягає в тому, що алгоритм погано працює з відкладеним рендерингом, алгоритмом що застосовується в більшості ігрових рушіїв.

3.3 Швидке приблизне згладжування

Швидке приблизне згладжування (FXAA) — найдешевший і один з найпростіших алгоритмів згладжування. Діаграма роботи методу швидкого приблизного згладжування представлена на рис. 3.3.

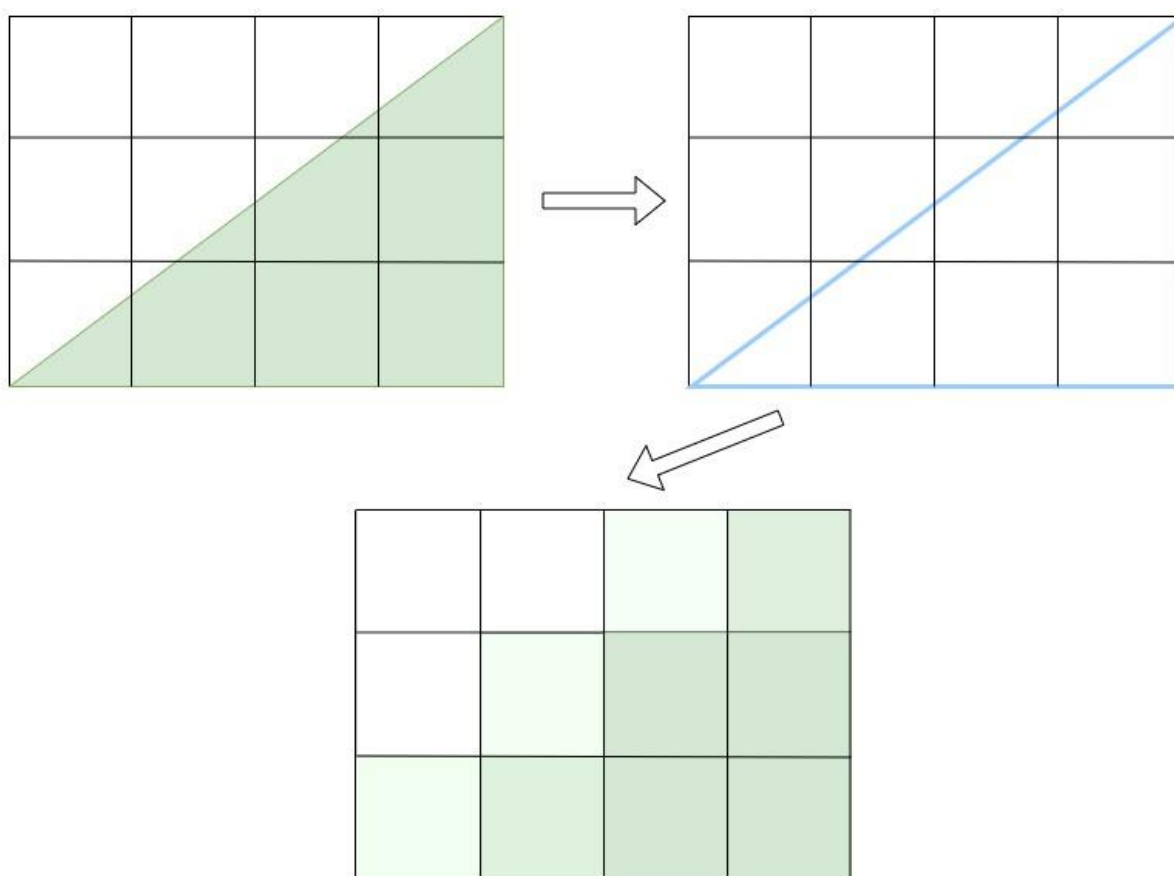


Рисунок 3.3 – Діаграма роботи методу швидкого приблизного згладжування

Цей метод використовується під час пост-обробки остаточного зображення.

Він просто розмиває групи пікселів із різкими змінами яскравості. Алгоритм роботи цього методу досить простий. Спочатку sRGB значення кожного пікселя перетворюється на лінійну оцінку яскравості цього пікселя. Наступним кроком є перевірка відносного контрасту навколишніх пікселів зі зразком — якщо є велика різниця, то це місце, швидше за все, знаходиться на краю. Після цього пікселі, що знаходиться на краю розмиваються. Цей метод є більш швидким порівняно з MSAA і майже будь-який навіть дуже старий графічний процесор може його використовувати, але у нього є суттєві недоліки. Він розмиває не тільки краї, але й важливі деталі текстур. В результаті погіршується загальна контрастність зображення. Крім того, цей метод розмиває зображення, орієнтуючись на колір пікселів, він не розпізнає вміст зображення.

3.4 Темпоральне згладжування

Темпоральне згладжування (TAA) — найпопулярніший алгоритм згладжування на сьогоднішній день. Цей гібрид поєднує в собі переваги алгоритмів SSAA і MSAA. Алгоритм роботи цього методу є одним з найскладніших в порівнянні з іншими методами. Спочатку генерується і відображається зображення без згладжування. Це зображення зберігається у фрагменті пам'яті, який називається буфером історії. Потім генерується наступний кадр, але перед відображенням значення пікселів поточного кадру змішується з минулим кадром з буферу історії. Після цього результат відображається і зберігається в буфері історії. Потім усі наступні кадри дотримуються цього шаблону: генерація зображення, змішування з зображенням в буфері історії, оновлення та відображення. В результаті накопичення послідовних кадрів відбувається згладжування. Тобто замість того, щоб використовувати більшу кількість вибірок під час генерації одного кадру цей метод розбиває цей процес на декілька кадрів. Однак, якщо послідовні кадри абсолютно однакові і між ними не відбувається жодних змін цей метод не матиме жодного ефекту тому зазвичай генерація зображення відбувається з довільним

зміщенням камери на невелику відстань. Злегка зміщені позиції розташування пікселів потім використовуються для вибірки з буфера історії, а тремтіння камери потім усувається під час фінальної обробки кадру. Таким чином, коли справа доходить до змішування значень із буфера історії із поточними майже завжди змішуються дискретизовані розташування субпікселів, які не зовсім у тому самому місці, що призводить до певного ступеня згладжування. Діаграма роботи методу темпорального згладжування представлена на рис. 3.4.

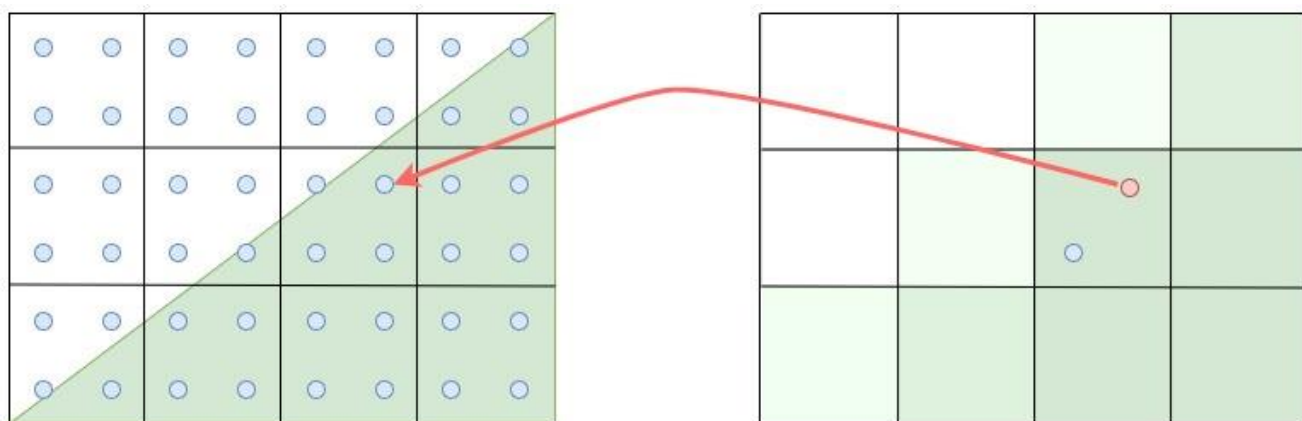


Рисунок 3.4 – Діаграма роботи методу темпорального згладжування

Темпоральне згладжування також може створювати надмірне розмиття, а також проблему, яка називається ореолом, коли краї рухомих об'єктів здаються розмазаними, а не згладженими. Один із поширених методів, який використовується, щоб обійти це, передбачає використання обчислювального шейдера для обчислення векторів руху об'єктів, збереження інформації в пам'яті, а потім порівняння відносних швидкостей поточного пікселя з відібраними з буферу історії – якщо вони помітно відрізняється, то вибірка з буферу історії відхиляється. Остаточне поєднання минулих і поточних пікселів також можна зважити, використовуючи порівняльні значення кольору, яскравості або швидкості. Під час остаточної копії оновленого буфера історії для відображення можна використовувати різні фільтри розмиття для подальшого зменшення видимості ореолів. ТAA також добре працює з відкладеним рендерингом і може використовуватися разом із FXAA та SMAA, забезпечуючи ще кращу якість

зображення. Однак через неточність роботи цього методу, особливо в динаміці, зображення виходить розмитим, і вимагає додаткової обробки. Незважаючи на те, що це пост-ефект, він набагато повільніший, ніж інші подібні алгоритми, через необхідність зберігання буфера векторів руху для всіх об'єктів на зображенні.

3.5 Висновки до розділу

В даному розділі було розглянуто найбільш популярні і актуальні методи згладжування зображень, що використовуються в сучасній комп'ютерній графіці, а саме: згладжування за допомогою суперсемплінгу, згладжування з кількома вибірками, швидке приблизне згладжування і темпоральнезгладжування. Було описано і проаналізовано принцип роботи наведених методів та виділено їх основні переваги та недоліки, що в подальшому були використані для формування вимог для створення нового методу.

4 РОЗРОБКА МЕТОДУ НА ОСНОВІ НЕЙРОННОЇ МЕРЕЖІ

4.1 Опис алгоритму роботи методу

Очевидно що для реалізації методу згладжування зображень в сучасній комп'ютерній графіці доцільно використовувати алгоритми штучного інтелекту, а саме штучну нейронну мережу, оскільки саме штучні нейронні мережі використовуються тоді, коли розробити алгоритм роботи програми занадто складно, швидкість роботи розробленого алгоритму буде занадто повільною через його складність, або алгоритм взагалі неможливо розробити через складність самої задачі яку алгоритм повинен вирішувати. Для реалізації методу згладжування зображень пропонується використати умовну генеративну змагальну мережу, оскільки такі штучні нейронні мережі часто використовуються для вирішення подібних задач. Багато проблем в обробці зображень, комп'ютерній графіці та комп'ютерному зорі можуть бути сформульовані як «переклад» вхідного зображення у відповідне вихідне зображення і умовна генеративна змагальна мережа є одним із методів що використовується для вирішення таких проблем. Умовна генеративна змагальна мережа (англ. Generative Adversarial Networks, GAN) є формою навчання без вчителя. Така мережа приймає певне вхідне зображення і на його основі генерує вихідне зображення. Цей підхід може бути успішно використаний у багатьох сферах застосування, в тому числі для згладжування зображень. Суть методу полягає в генерації зображення без згладжування і використання його в якості вхідного зображення нейронної мережі для генерації згладженого зображення високої якості. Для навчання мережі пропонується використати набір навчальних даних, що складається з пар зображень. Кожна пара зображень включає зображення згенероване без згладжування і зразкове згладжене зображення ідеальної якості. Після навчання мережі даний метод згладжування можна буде використовувати для згладжування будь-яких зображень і як основний метод згладжування зображень в ігрових рушіях та рушіях візуалізації. Алгоритм роботи методу представлений на рис. 4.1.



Рисунок 4.1 – Алгоритм роботи методу

4.2 Умовна генеративна змагальна мережа

Умовні генеративні змагальні мережі - це генеративні моделі, які навчаються побудові зображення зі спостережуваного зображення x та випадкового вектора шуму r , в y ,

$$G: \{x, r\} \rightarrow y \quad (4.1)$$

Умовна генеративна змагальну мережа складається з двох основних частин: генератора та дискримінатора. Генератор застосовує деяке перетворення до вхідного зображення, щоб отримати вихідне зображення. Дискримінатор

порівнює вхідне зображення з невідомим зображенням (цільовим зображенням із набору даних або вихідним зображенням із генератора) і намагається вгадати, чи було воно створене генератором. Діаграма навчання мережі представлена на рис.4.2.

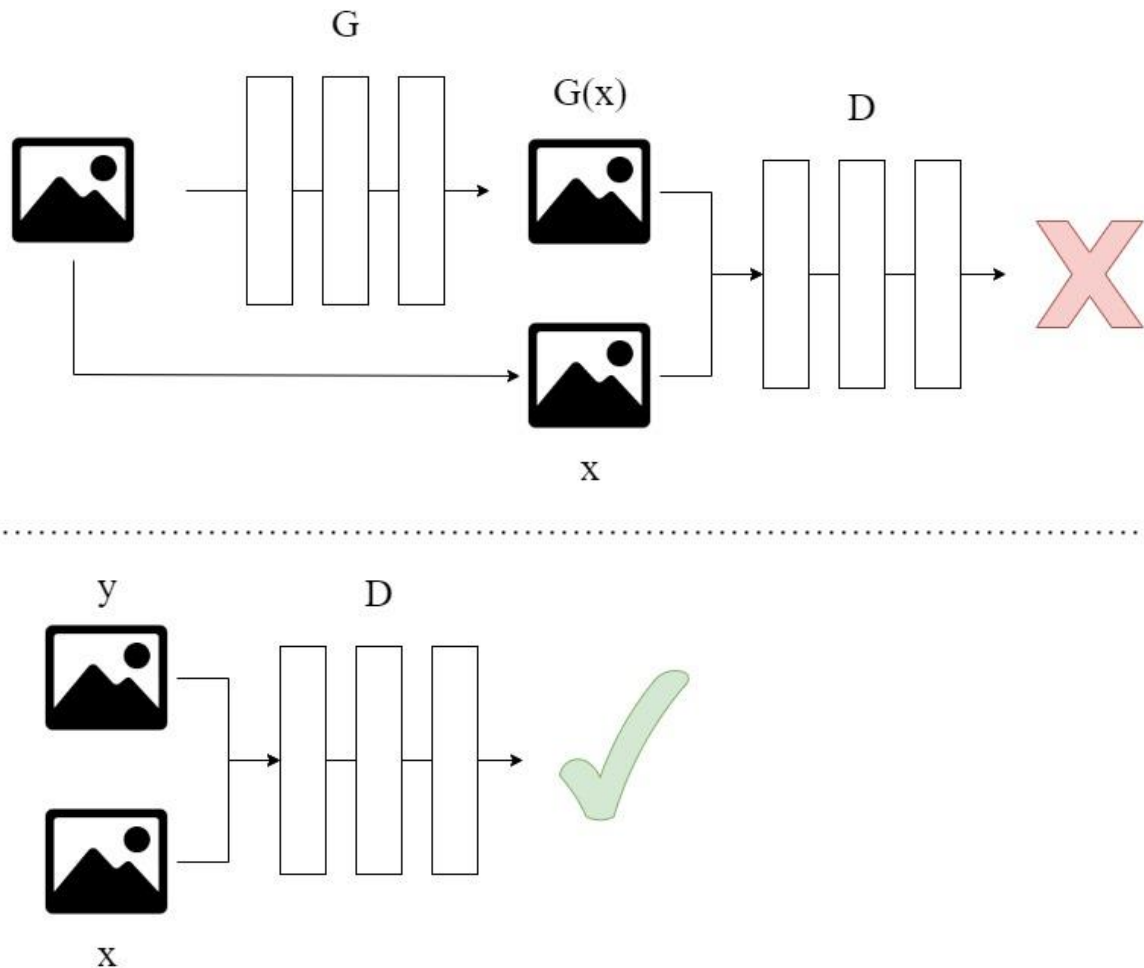


Рисунок 4.2 – Діаграма навчання мережі

Умовні генеративні змагальні мережі призначені для моделювання умовного розподілу реальних зображень при заданих вхідних зображеннях за допомогою наступної мінімаксної гри:

$$\min_G \max_D \mathcal{L}_{CGAN}(G, D) \quad (4.2)$$

де генератор (G) намагається мінімізувати цільову функцію, а дискримінатор (D)

намагається максимізувати її. Цільова функція задається формулою:

$$\mathcal{L}_{cGAN}(G, D) = E_{x,y}[\log D(x, y)] + E_{x,r}[\log(1 - D(x, G(x, r)))] \quad (4.3)$$

де x - вхідне зображення, r - вектор випадкового шуму, y - цільове зображення. Для перевірки важливості обумовленості дискримінатора, також проведене порівняння з варіантом без умов, в якому дискримінатор не спостерігає x :

$$\mathcal{L}_{cGAN}(G, D) = E_y[\log D(y)] + E_{x,r}[\log(1 - D(G(x, r)))] \quad (4.4)$$

Існуючі підходи до використання умовних генеративних змагальних мереж показали, що корисно поєднувати цільову функцію з більш традиційними функціями втрат, такими як функція втрат L2. Робота дискримінатора залишається незмінною, але генератор має завдання не тільки обдурити дискримінатор, але й також бути близьким до вихідного результату істини в сенсі функції втрат L2. В даному методі також пропонується цей варіант, але замість функції втрат L2 використовується функція втрат L1, оскільки вона сприяє меншій розмитості зображення.

$$\mathcal{L}_{L1}(G) = E_{x,y,r}[\|y - G(x, r)\|_1] \quad (4.5)$$

Отже фінальна цільова функція виглядає так:

$$\min_G \max_D \mathcal{L}_{cGAN}(G, D) + \lambda \mathcal{L}_{L1}(G) \quad (4.6)$$

Без r мережа все ще може вивчити перетворення від x до y , але видаватиме детерміновані результати, а отже, не відповідатиме жодному розподілу, окрім дельта-функції. Минулі умовні генеративні змагальні мережі визнавали це і надавали Гаусів шум r як вхідні дані для генератора, на додачу до x . Але ця

стратегія не є ефективною тому, що генератор просто навчиться ігнорувати шум. Замість цього в даному методі подається шум лише на декілька шарів генератора як під час навчання, так і під час тестування. Незважаючи на шум в роботі мережі спостерігається лише незначна стохастичність. Проектування умовних генеративних змагальних мереж, які виробляють високо стохастичні результати, і, таким чином, охоплюють повну ентропію умовних розподілів, які вони моделюють, є важливим питанням, яке все ще залишається відкритим.

4.3 Архітектура мережі

Генератор і дискримінатор є основними компонентами умовної генеративної змагальної мережі. Архітектура мережі представлена на рис. 4.3.

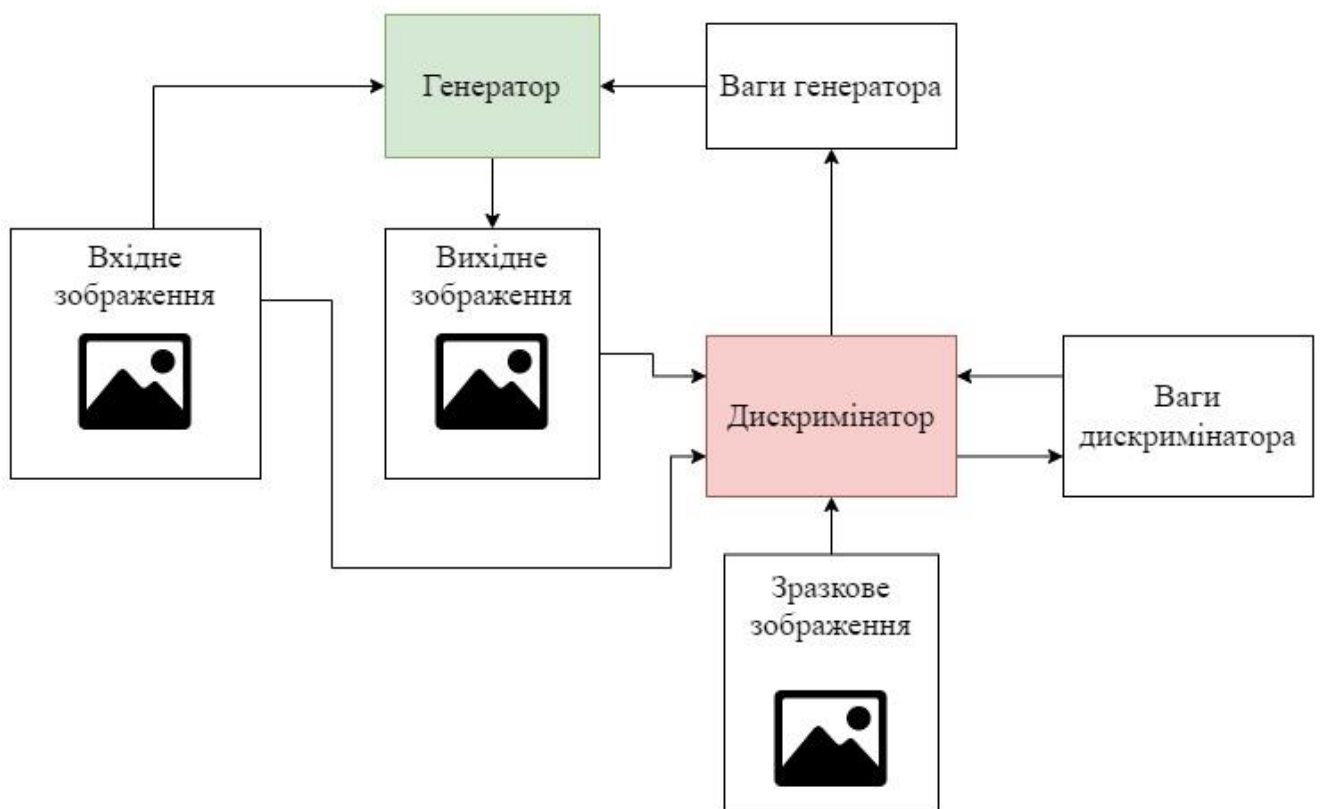


Рисунок 4.3 – Архітектура мережі

Генератор перетворює вхідне зображення для отримання вихідного зображення. Дискримінатор порівнює вхідне зображення з невідомим

зображенням (або цільовим зображенням з набору даних, або вихідним зображенням генератора) і намагається визначити, чи було воно створено генератором. Дискримінатор надає функцію втрат для навчання генератора, тому його не потрібно вводити вручну, що є одним з основних аспектів генеративної змагальної мережі.

Завданням генератора є виконання необхідних перетворень над вхідним зображенням для отримання цільового зображення. Прикладом вхідного зображення може зображення без згладження, а на виході ми хочемо отримати згладжену версію цього зображення. Структура генератора називається «кодер-декодер», і виглядає так:

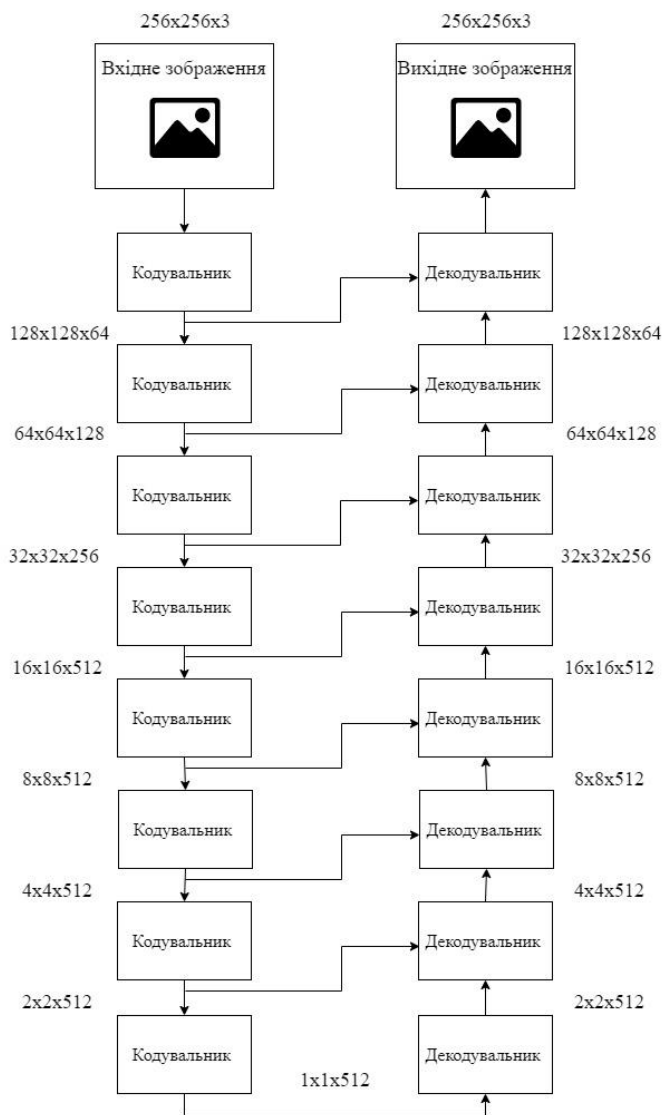


Рисунок 4.3 – Структура генератора

Генератор складається з шарів кодувальників, що стискають зображення до значно меншого представлення, і відповідних шарів декодувальників що виконують зворотну операцію. Вхідними даними в цьому прикладі є зображення розміром 256x256 із 3 кольоровими каналами (червоний, зелений і синій), і вихідні дані є такими ж самими. Генератор приймає деякі вхідні дані та намагається зменшити їх за допомогою серії кодувальників (конволюція + функція активації) до набагато меншого представлення.

Приклад структури шарів кодування і декодування представлено на рис 4.4.

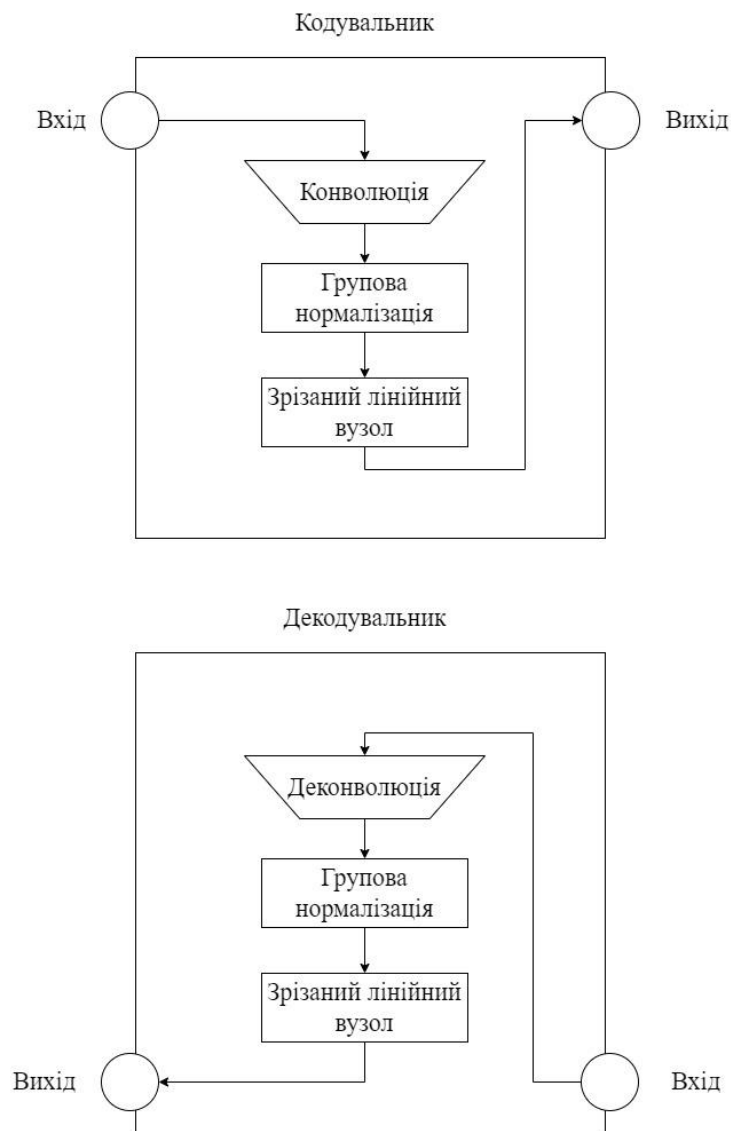


Рисунок 4.4 – Структура шарів кодування і декодування

Ідея полягає в тому, що, зменшуючи таким чином дані, можна отримати

більш високе представлення даних після останнього шару кодування. Шари декодування діють протилежно (деконволюція + функція активації) і повертають дію кодуєчих шарів у зворотному напрямку.

Визначальною особливістю задач перекладу зображення в зображення є те, що вони перетворюють вхідну сітку з високою роздільною здатністю у вихідну сітку з високою роздільною здатністю. Крім того, для задач, які ми розглядаємо, вхідні та вихідні дані відрізняються за зовнішнім виглядом, але вони є відображенням однієї і тієї ж базової структури. Тому структура на вході приблизно відповідає структурі на виході. Розроблена архітектура генератора ґрунтується на цих міркуваннях. Багато попередніх рішень проблем в цій області використовували мережу кодер-декодер. У такій мережі вхідні дані проходять через ряд шарів, які поступово зменшують дискретизацію, до тих пір, поки не буде досягнуто «вузького місця», після чого процес змінюється на протилежний. Така мережа вимагає, щоб весь інформаційний потік проходив через всі шари, включаючи вузьке місце. Для багатьох проблем перекладу зображень між входом і виходом є велика кількість спільної низькорівневої інформації, і було б бажано передавати цю інформацію безпосередньо через мережу. Для того, щоб дати генератору можливість обійти «вузьке місце» для такої інформації, додаються пропускні зв'язки, що відповідають загальній формі «U-мережі». Зокрема, додаються пропускні зв'язки між кожним шаром i та шаром $n - i$, де n - загальна кількість шарів. Кожне з'єднання просто об'єднує всі канали на шарі i з каналами на шарі $n - i$. Нормалізація покращує швидкість навчання. Batch Normalisation (BatchNorm) виконує глобальну нормалізацію за розміром пакету, де розміри набору даних знаходяться в кортежі: (розмір пакету, розмір каналу, висота, ширина). Нормалізація екземплярів (InstanceNorm) подібна до пакетної нормалізації, але обчислюється для кожного зразка, а не для пакетів. Нормалізація шару діє вздовж розміру каналу. Групова нормалізація (GroupNorm) є сумішшю між нормалізацією шарів та нормалізацією екземплярів, але вхідні канали розбиваються на групи. В даному методі використовується групова нормалізація, оскільки навчання мережі проводилося на одиничному розмірі партії через

обмеження пам'яті. Було виявлено, що групова нормалізація працює краще, коли розмір партії невеликий. Для компактності прогнозоване фальшиве зображення позначається як pf , а реальне як v . Оновлення генераторної мережі здійснюється за допомогою функцію втрат:

$$Loss = BCE(pf, v) + \lambda L1 \quad (4.7)$$

де $\lambda = 100$. Генераторна мережа має вісім шарів кодування від 3 каналів до 512 каналів, а потім вісім шарів декодування від 512 каналів до 3 каналів на виході. Структуру дискримінатора показано на рис.3.5

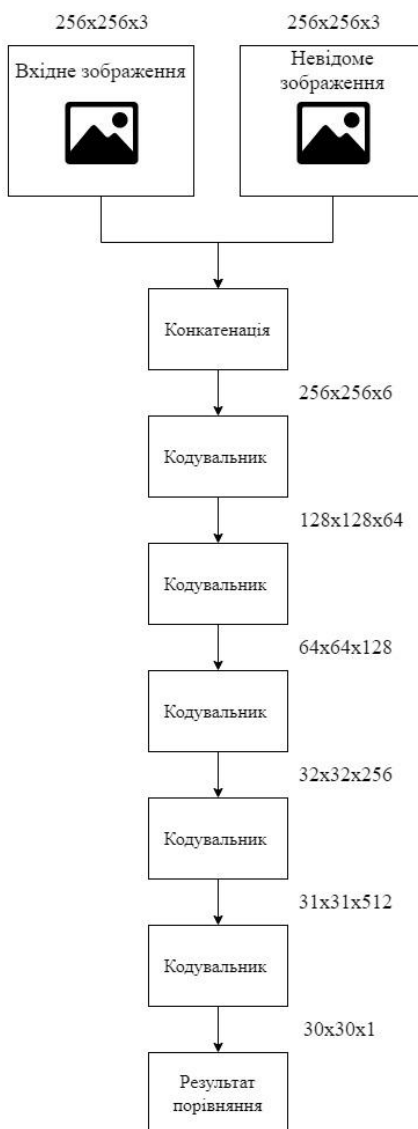


Рисунок 4.5 – Структура дискримінатора

Дискримінатор має завдання отримати два зображення, вхідне зображення і невідоме зображення (яке буде або цільовим, або вихідним зображенням генератора), і вирішити, чи було друге зображення вироблено генератором, чи ні.

Структура дискримінатора дуже схожа на секцію кодера генератора, але працює дещо інакше. Результатом є зображення розміром 30x30, де значення кожного пікселя (від 0 до 1) відображає, наскільки правдоподібним є відповідний розділ невідомого зображення. Кожен піксель цього зображення розміром 30x30 відповідає правдоподібності фрагмента вхідного зображення розміром 70x70. Така архітектура називається «Марківський дискримінатор», або «PatchGAN». Архітектура дискримінатора використовує мережу «PatchGan» для визначення того, чи є зображення справжнім чи підробленим. Для компактності прогнозоване реальне зображення позначається як p_r , прогнозоване фальшиве зображення як p_f , реальне як v , а фальшиве як f . Для оновлення мережі використовується наступна функція втрат:

$$Loss = 0.5(BCE(p_r, v) + BCE(p_f, f)) \quad (4.8)$$

Дискримінатор складається з 5 блоків дискримінатора, де блок складається з конволюції, опціональної нормалізації та ReLU. Потім застосовується нульове заповнення і виконується остаточна процедура конволюції. Добре відомо, що функції втрати L2 і L1 призводять до розмиття результатів при створенні зображень. Хоча ці функції втрати не сприяють підвищенню чіткості на високих частотах, у багатьох випадках вони, тим не менш, точно фіксують низькі частоти. Для проблем, де це має місце, немає необхідності в абсолютно новій архітектурі для забезпечення коректності на низьких частотах. Функції втрати L1 буде достатньо. Тому доцільно обмежити дискримінатор умовної генеративної змагальної нейронної мережі лише моделюванням високочастотної структури, використовуючи функцію втрат L1 для забезпечення низькочастотної коректності. Для моделювання високих частот достатньо обмежити увагу на структурі в локальних ділянках зображення. Тому було розроблено архітектуру

дискримінатора, яка називається «PatchGAN», що враховує структуру лише в масштабі окремих ділянок зображення. Цей дискримінатор намагається класифікувати, чи є кожна ділянка $N \times N$ пікселів на зображенні справжньою чи підробленою. Такий дискримінатор конволютно застосовується до зображення, усереднюючи всі відповіді для отримання кінцевого результату дискримінатора.

Було встановлено, що N може бути набагато менше повного розміру зображення і при цьому отримувати високу якість результатів. Це є перевагою, оскільки менший «PatchGAN» має менше параметрів, працює швидше і може бути застосований до зображень довільного розміру. Такий дискримінатор ефективно моделює зображення як марковське випадкове поле, припускаючи незалежність між пікселями, розділеними більш ніж діаметром ділянки.

Таким чином, даний «PatchGAN» можна розглядати як різновид функції втрати текстури/стилю.

Для оптимізації мереж використовується звичайна стратегія: чергуються один крок градієнтного спуску дискримінатора і один крок генератора. Замість того, щоб навчати генератор на мінімізацію

$$\log(1 - D(x, G(x, r))) \quad (4.9)$$

як пропонувалося в оригінальному дослідженні генеративних змагальних нейронних мереж, відбувається навчання на максимізацію

$$\log D(x, G(x, r)) \quad (4.10)$$

Крім цього, при оптимізації дискримінатора цільова функція ділиться на 2, що сповільнює швидкість навчання дискримінатора відносно генератора. Також використовується стохастичний градієнтний спуск і адаптивне оцінювання моменту зі швидкістю навчання 0,0002, і параметрами імпульсу $\beta_1 = 0,5$ та $\beta_1 = 0,999$. Під час виведення генераторна мережа запускається точно так само, як і під час тренувальної фази.

4.4 Навчання мережі

Навчання цієї мережі відбувається в два етапи: навчання дискримінатора та навчання генератора. Для навчання дискримінатора спочатку генератор генерує вихідне зображення. Дискримінатор дивиться на пару вхідне/цільове і пару вхідне/вихідне і виробляє своє припущення про те, наскільки реалістично вони виглядають. Потім ваги дискримінатора коригуються на основі помилки класифікації пари вхідне/вихідне і пари вхідне/цільове. Потім ваги генератора коригуються на основі вихідного сигналу дискримінатора, а також різниці між вихідним і цільовим зображенням. Особливість такого навчання полягає в тому, що коли тренується генератор на результаті роботи дискримінатора, фактично обчислюються градієнти через дискримінатор, що означає, що в той час як дискримінатор покращується, генератор навчається, щоб перевершити дискримінатор. Теорія полягає в тому, що в міру того, як дискримінатор покращується, покращується і генератор. Якщо дискримінатор добре виконує свою роботу, а генератор здатний вивчити правильну функцію перетворення за допомогою градієнтного спуску, результат буде непоганим. Мережа була реалізована за допомогою PyTorch і була навчена на 80% даних, в той час як 10% були збережені для валідації, а інші 10% були використані в якості тестових даних. Загалом мережа була навчена до 1200 епох. Конкретне апаратне забезпечення, яке використовувалося для навчання моделі, наведено нижче: Процесор Intel Core i5-9600K (6 ядер), графічний процесор Nvidia GeForce GTX 1070 (8 ГБ) та 16 ГБ системної оперативної пам'яті.

Створення набору даних було виконано за допомогою ігрового рушія Unreal Engine 5. Спочатку була створена стандартна сцена з підлогою. Потім було обрано випадкову кількість об'єктів для відображення на екрані - від 1 до 100. Значення від 1 до 100 повинні забезпечити необхідну випадковість сцени при переході від одного зображення до іншого. Для кожного з об'єктів випадковим чином обирається положення, кут обертання та колір, а також одна з шести примітивних фігур. Серед примітивних фігур були куб, циліндр, конус, сфера, капсула та тор.

Після створення об'єктів камера випадковим чином повертається на ± 10 градусів по осі x , а потім від 0 до 360 градусів по осі z . Потім за допомогою рушія візуалізації Unreal Engine 5, який є рушієм растеризації, здійснюється відображення зображення без згладжування. Потім рушій відтворює високоякісне вихідне зображення для навчання використовуючи суперсемплінг в якості методу згладжування. Результат зберігається, сцена скидається, а потім вищеописані кроки повторюються для кожного зображення. Всього було створено 2560 наборів зображень, де один набір зображень містить 2 зображення, загалом 5120 зображень. Зображення генеруються з роздільною здатністю 1024×512 пікселів. Прикладом набору даних для навчання мережі може бути набір пар зображень, в яких вхідне зображення є зображенням без згладження, а цільове зображення — версією зображення після згладжування, як показано на рис 3.6.

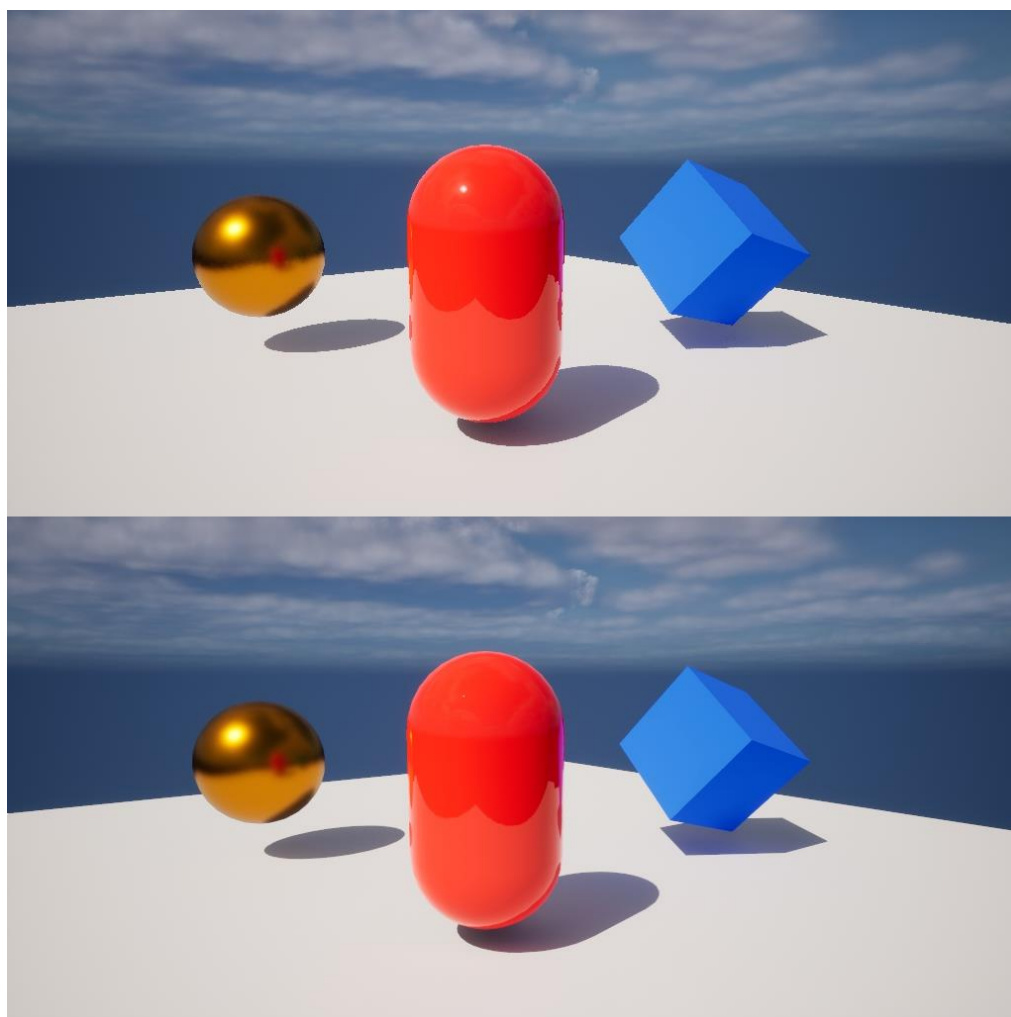


Рисунок 4.6 – Приклад пари зображень в наборі даних для навчання мережі

Генератор у цьому випадку намагається навчитися згладжувати зображення, а дискримінатор дивиться на результат роботи генератора та намагається навчитися розпізнавати різницю між зображеннями, які надає генератор, і справжнім згладженим цільовим зображенням, наданим у наборі даних.

4.5 Тестування методу

Під час тестування методу згладжування зображень на основі умовної генеративної змагальної нейронної мережі в якості зразкового згладженого зображення використовувалося зображення згладжене методом згладжування за допомогою суперсемплінгу оскільки це єдиний, метод результатом роботи якого є згладжене зображення ідеальної якості. Для генерації зображень, а також для порівняння існуючих методів використовувався ігровий рушій Unreal Engine 5. Конкретне апаратне забезпечення, яке використовувалося для тестування методу, наведено нижче: Процесор Intel Core i5-9600K (6 ядер), графічний процесор Nvidia GeForce GTX 1070 (8 ГБ) та 16 ГБ системної оперативної пам'яті. Чотири показника було використано для оцінки ефективності цього методу.

1. Функція втрат L1:

$$L1 = \sum_{i=1}^n |y_{\text{реальне}} - y_{\text{прогнозоване}}| \quad (4.11)$$

Функція втрат L1 - це сумарна абсолютна різниця між кожним пікселем на прогнозованому зображенні від пікселя на зразковому зображенні.

2. Функція втрат L2:

$$L2 = \sum_{i=1}^n (y_{\text{реальне}} - y_{\text{прогнозоване}})^2 \quad (4.12)$$

Функція втрат L2 - це зведена в квадрат абсолютна різниця між кожним пікселем прогнозованого зображення та зразкового зображення.

3. Індекс структурної подібності:

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)} \quad (4.13)$$

Індекс структурної подібності (SSIM) використовується для вимірювання подібності між двома зображеннями. Замість того, щоб дивитися на різницю між пікселями, SSIM вимірює деградацію як сприйняту зміну в структурній інформації.

4. Початкова відстань Фреше:

$$FID(r, g) = \|\mu_r - \mu_g\|_2^2 + Tr(\Sigma_r + \Sigma_g - 2 \cdot \sqrt{\Sigma_r \Sigma_g}) \quad (4.14)$$

Початкова відстань Фреше (FID) між двома розподілами використовується для оцінки якості згенерованих зразків, де менший FID означає меншу відстань між реальним і згенерованим розподілами. На таблиці 4.1 показано результат тестування ефективності методу:

Таблиця 4.1 – Результат тестування ефективності методу

Показник	Метод згладжування на основі умовної генеративної змагальної мережі	Метод згладжування за допомогою суперсемплінгу
L1	45,158.14	0.0
L2	2135.72	0.0
SSIM	0.9879	1.0
FID	0.0029	0.0

Середнє значення функції втрат L1 для методу становить 45,158.14. Це показує, що метод згладжування на основі умовної генеративної змагальної мережі дуже близький до результату методу згладжування за допомогою суперсемплінгу, оскільки функція втрат L1 демонструє абсолютну сумарну різницю між двома результатами.

Функція втрат L2 показує сумарну квадратичну різницю між двома зображеннями i , отже, є більш чутливою до відхилень. Середнє значення функції втрат L2 для методу становить 2135.72.

Середнє значення SSIM за результатами методу згладжування на основі умовної генеративної змагальної мережі становить 0.9879. Це свідчить про те, що структурно результат методу ближчий до результату методу згладжування за допомогою суперсемплінгу. Беручи до уваги час, необхідний для отримання зображення за допомогою нейронної мережі у порівнянні з суперсемплінгом, метод згладжування на основі умовної генеративної змагальної мережі дає порівнянну якість зображення за менший проміжок часу.

Середнє значення FID показує, що відстань між результатами методу згладжування за допомогою суперсемплінгу і методу згладжування на основі умовної генеративної змагальної мережі становить лише 0.0029. Це свідчить про те, що метод на основі нейронної мережі хоч і не є точно таким же, як зображення, отримане методом суперсемплінгу, але обробляється за менший проміжок часу.

Для оцінки швидкості роботи методу було проведено його порівняння з існуючими методами. Для тесту використовувалося зображення 1024x512 пікселів. На таблиці 4.2 показано результат порівняння швидкості методів. В таблиці в окремий стовпчик винесено час витрачений на rasterизацію зображення оскільки різні методи згладжування зображень використовують різні алгоритми візуалізації зображення, що в свою чергу впливає на загальний час отримання зображення.

Отже метод метод згладжування на основі умовної генеративної змагальної мережі набагато швидший за суперсемплінг і за швидкістю приближений до темпорального згладжування, але за якістю він наближений до суперсемплінгу

оскільки на відміну від інших методів він повністю усуває всі артефакти аліасингу і не розмиває зображення.

Таблиця 4.2 – Результат тестування швидкості роботи методу

Метод	Растеризація	Згладжування	Всього
Згладжування за допомогою суперсемплінгу	20.73 мс	14.53 мс	35.26 мс
Згладжування з кількома вибіркам	11.82 мс	2.41 мс	14.23 мс
Швидке приблизне згладжування	20.73 мс	0.36 мс	21.09 мс
Темпоральне згладжування	20.73 мс	2.47 мс	23.2 мс
Метод згладжування на основі умовної генеративної змагальної мережі	20.73 мс	2.9 мс	23.63 мс

4.6 Висновки до розділу

В даному розділі описано алгоритм роботи методу згладжування зображень на основі умовної генеративної змагальної мережі. Представлено особливості архітектури мережі і основних її компонентів. Описано спосіб навчання і тестування мережі.

ВИСНОВКИ

Під час виконання цієї дипломної роботи було проаналізовано найбільш популярні і актуальні з існуючих методів згладжування зображень в комп'ютерній графіці, а саме згладжування за допомогою суперсемплінгу, згладжування з кількома вибірками, швидке приблизне згладжування та темпоральне згладжування. Було описано і порівняно принцип роботи наведених методів та виділено їх основні переваги та недоліки, що в подальшому були використані для формування вимог для створення нового методу.

На основі дослідження спеціалізованої літератури та інтернет-ресурсів були отримані, засвоєні та застосовані концепції та знання з теорії штучного інтелекту, штучних нейронних мереж, комп'ютерної графіки та згладжування зображень.

Були досліджені різноманітні архітектури нейронних мереж, а також структура штучного нейрона та різні методики навчання нейронних мереж та були представлені їх переваги і недоліки.

Розроблено метод на основі штучної нейронної мережі, а саме умовної генеративної змагальної мережі для вирішення проблеми згладжування зображень в комп'ютерній графіці.

Проведено аналіз роботи методу та його ефективності в порівнянні з іншими методами.

Зображення згладжені за допомогою розробленого методу структурно ближчі до результату методу згладжування за допомогою суперсемплінгу, оскільки індекс структурної подібності становить 0.9879. В той же час за швидкістю розроблений метод подібний до темпорального згладжування, оскільки швидкість його роботи становить 2.9 мілісекунд. На відміну від інших методів згладжування зображень метод на основі умовної генеративної змагальної мережі повністю усуває всі артефакти аліасингу і не розмиває зображення. Отже розроблений метод дозволяє отримувати зображення майже ідеальної якості і робити це дуже швидко.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Lei Yang, Shiqiu Liu, Marco Salvi. A Survey of Temporal Antialiasing Techniques. – 2020.
2. Adam Marrs, Josef Spjut, Holger Gruen, Rahul Sathe. Adaptive temporal antialiasing – 2018.
3. Thomas Auzinger, Przemyslaw Musialski, Reinhold Preiner, Michael Wimmer. Non-Sampled Anti-Aliasing. – 2013.
4. Paru Raj. A Novel Technique for Removing Jagged Edges of Image Due to Aliasing. – 2017.
5. Jae-Ho Nah, Sunho Ki, Yk Lim, Park Jinhong. AXAA: adaptive approximate anti-aliasing.– 2016.
6. Siam, F. M. Jamius, Prince, Zahidul Islam, Bari, Ahmed Na sul. Anti-aliasing for real-time applications in 3D using deep convolutional neural network. – 2020.
7. O. Nalbach, E. Arabadzhiyska, D. Mehta, H.-P. Seidel, T. Ritschel. Deep Shading: Convolutional Neural Networks for Screen Space Shading. – 2017.
8. Hallison Paz, Tiago Novello, Vinicius Silva, Luiz Schirmer, Guilherme Schardong, Fabio Chagas, Helio Lopes, Luiz Velho. Multiresolution Neural Networks for Imaging. – 2022.
9. Anjul Patney, Aaron Lefohn. Detecting aliasing artifacts in image sequences using deep neural networks. – 2018.
10. Dissanayake, J. M. T. C. RSR - Real-time Super sampling for Rendering using Machine Learning. – 2021.
11. Alexander Keller, Jaroslav Křivánek, Jan Novák, Anton Kaplanyan, Marco Salvi. Machine learning and rendering. – 2018.
12. Ketan Kapse. An Overview of Current Deep Learned Rendering Technologies. – 2021.
13. Lei Xiao, Salah Nouri, Matt Chapman, Alexander Fix, Douglas Lanman, Anton Kaplanyan. Neural supersampling for real-time rendering. – 2020.

14. Jinjin Gu, Haoming Cai, Chenyu Dong, Ruofan Zhang, Yulun Zhang, Wenming Yang, Chun Yuan. Super-Resolution by Predicting Offsets: An Ultra-Efficient Super-Resolution Network for Rasterized Images. – 2022.
15. Grahn Alexander. An Image and Processing Comparison Study of Antialiasing Methods. – 2016.
16. Ye Chengchun, Lai Shunnan, Li Sheng. An efficient spatial anti-aliasing method for deferred shading. – 2016.
17. Hyltegård, Simon. Two Anti-aliasing Methods for Creating a Uniform Look. – 2016.
18. Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, Alexei A. Efros. Image-to-Image Translation with Conditional Adversarial Network. – 2016.
19. Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, Yoshua Bengio. Generative Adversarial Networks. – 2014.
20. Leon A. Gatys, Alexander S. Ecker, Matthias Bethge. Texture Synthesis Using Convolutional Neural Networks. – 2015.

ДОДАТОК



ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ
 НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ
 ТЕХНОЛОГІЙ



Кафедра інженерії програмного забезпечення

МАГІСТЕРСЬКА РОБОТА

**«ЗАСТОСУВАННЯ МЕТОДУ ЗГЛАДЖУВАННЯ ЗОБРАЖЕНЬ В
 ТРИВИМІРНІЙ КОМП'ЮТЕРНІЙ ГРАФІЦІ З ВИКОРИСТАННЯМ
 АЛГОРИТМУ МАШИННОГО НАВЧАННЯ»**

Виконав: студент групи ПДМ-61 Пашенко Владислав Юрійович

Керівник: доктор філософії, доцент кафедри ІІЗ Дібрівний О.А.

Київ - 2022

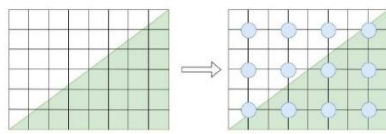
МЕТА, ОБ'ЄКТ, ПРЕДМЕТ ДОСЛІДЖЕННЯ

Мета роботи: підвищення ефективності згладжування зображень в тривимірній комп'ютерній графіці за рахунок використання алгоритмів машинного навчання.

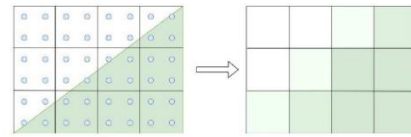
Об'єкт дослідження: процес згладжування зображень в тривимірній комп'ютерній графіці.

Предмет дослідження: алгоритми машинного навчання та методи згладжування зображень.

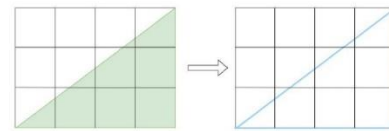
ІСНУЮЧІ МЕТОДИ ЗГЛАДЖУВАННЯ ЗОБРАЖЕНЬ В ТРИВИМІРНІЙ КОМП'ЮТЕРНІЙ ГРАФІЦІ



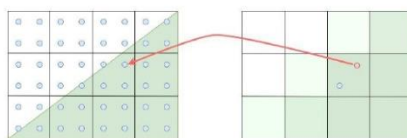
Згладжування за допомогою суперсемплінгу



Згладжування з кількома вибіркам



Швидке приближене згладжування



Темпоральне згладжування

3

ПОРІВНЯННЯ МЕТОДІВ ЗГЛАДЖУВАННЯ ЗОБРАЖЕНЬ В ТРИВИМІРНІЙ КОМП'ЮТЕРНІЙ ГРАФІЦІ

Метод	Переваги	Недоліки
Згладжування за допомогою суперсемплінгу	Єдиний метод результатом роботи якого є згладжене зображення ідеальної якості.	Метод дуже повільний.
Згладжування з кількома вибіркам	Швидше за суперсемплінг, імплементований в драйвері відеокарти.	Добре працює лише на межах полігонів і тому не усуває артефакти, що виникають на текстурах в середині полігонів.
Швидке приближене згладжування	Швидше за суперсемплінг, просто імплементувати.	Розмиває важливі деталі текстур.
Темпоральне згладжування	Найкраща якість зображення в порівнянні з іншими швидкими методами.	Може створювати надмірне розмиття, а також проблему, яка називається ореолом (краї рухомих об'єктів здаються розмазаними, а не згладженими).

4

МЕТОД НА ОСНОВІ УМОВНОЇ ГЕНЕРАТИВНОЇ ЗМАГАЛЬНОЇ НЕЙРОННОЇ МЕРЕЖІ

Умовна генеративна змагальна нейронна мережа - мережа, що складається з генератора і дискримінатора. Завданням генератора G полягає у переведенні вхідного зображення у вихідне, в той час як дискримінатор D має на меті відрізнити реальні зображення від згенерованих.

Умовні генеративні змагальні мережі призначені для моделювання умовного розподілу реальних зображень при заданих вхідних зображеннях за допомогою наступної мінімаксної гри:

$$\min_G \max_D \mathcal{L}_{cGAN}(G, D)$$

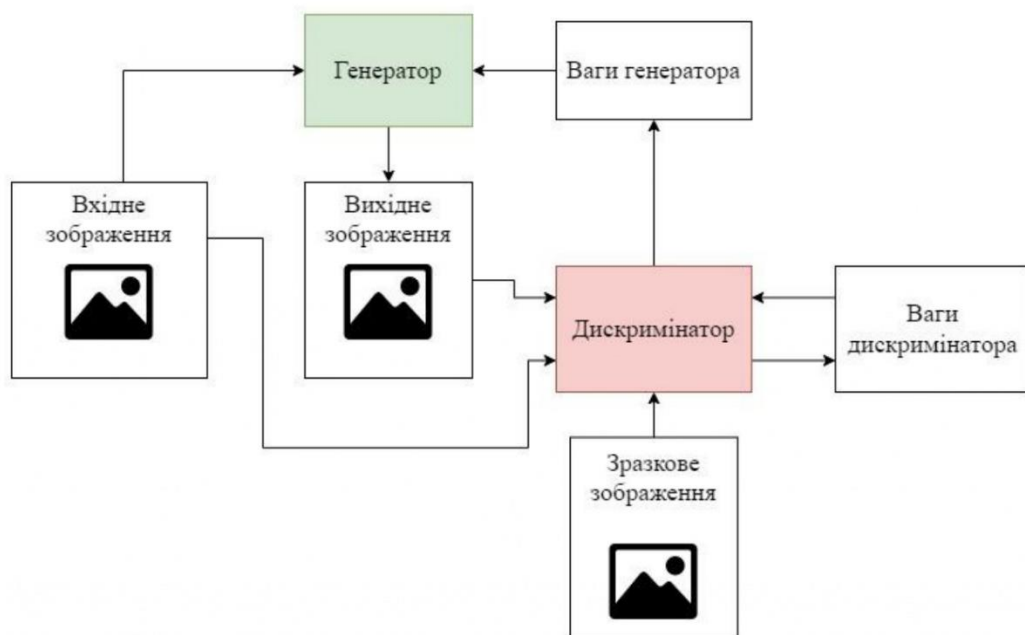
де цільова функція задається формулою:

$$\mathcal{L}_{cGAN}(G, D) = E_{x,y}[\log D(x, y)] + E_{x,r}[\log(1 - D(x, G(x, r)))]$$

де x - вхідне зображення, r - вектор випадкового шуму, y - цільове зображення

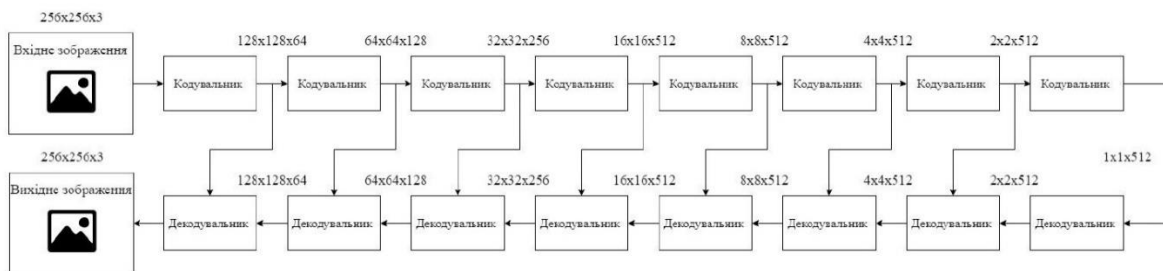
5

АРХІТЕКТУРА МЕРЕЖІ



6

СТРУКТУРА ГЕНЕРАТОРА І ДИСКРИМІНАТОРА



Структура генератора



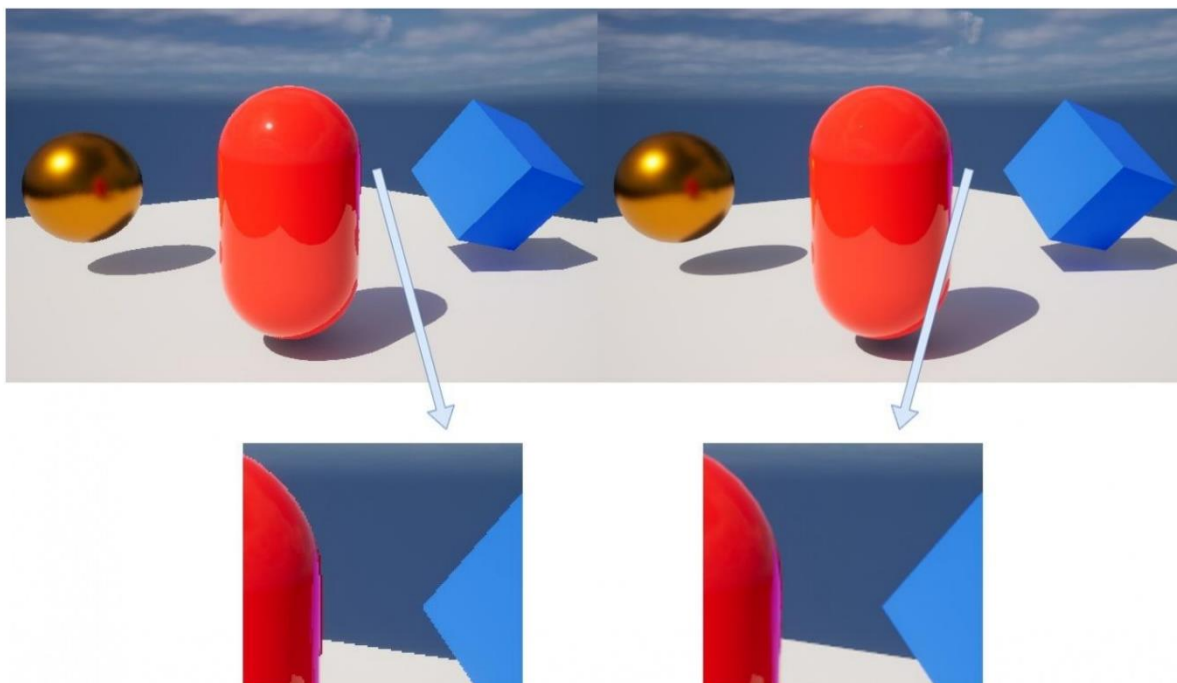
Структура дискримінатора

7

ПРИКЛАД РОБОТИ МЕТОДУ

Зображення без згладжування

Зображення після згладжування



8

ПОРІВНЯЛЬНИЙ АНАЛІЗ

	Растрезація	Згладжування	Всього
Згладжування за допомогою суперсемплінгу	20.73 мс	14.53 мс	35.26 мс
Згладжування з кількома вибіркама	11.82 мс	2.41 мс	14.23 мс
Швидке приблизне згладжування	20.73 мс	0.36 мс	21.09 мс
Темпоральне згладжування	20.73 мс	2.47 мс	23.2 мс
Метод представлений в дипломній роботі	20.73 мс	2.9 мс	23.63 мс

9

ВИСНОВКИ

1. Проаналізовано існуючі методи згладжування зображень в комп'ютерній графіці.
2. На основі дослідження та опрацювання спеціалізованої літератури та інтернет-ресурсів були отримані, засвоєні та застосовані концепції та знання з теорії штучного інтелекту, штучних нейронних мереж, комп'ютерної графіки та згладжування зображень.
3. Були досліджені різноманітні архітектури нейронних мереж, а також структура штучного нейрона та різні методики навчання нейронних мереж.
4. Розроблено модель на основі штучних нейронних мереж для вирішення проблеми згладжування зображень в комп'ютерній графіці.
5. Проведено аналіз роботи методу та його ефективності в порівнянні з іншими методами.

10

ПУБЛІКАЦІЇ ТА АПРОБАЦІЯ РОБОТИ

Тези доповідей:

1. Пашенко В.Ю. Розробка методу згладжування зображень в тривимірній комп'ютерній графіці з використанням алгоритму машинного навчання. // XV Науково-технічна конференція «Сучасні інфокомунікаційно технології» – Київ: ДУТ, 2022.

2. Пашенко В.Ю. Метод згладжування зображень в сучасній комп'ютерній графіці. // III Всеукраїнської науково-практична конференція молодих вчених та студентів «Інженерія програмного забезпечення і передові інформаційні технології (SoftTech-2022) – Київ: КПІ ім. Ігоря Сікорського, 2022.

ДЯКУЮ ЗА УВАГУ!