Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет комп’ютерної інженерії та управління
(повна назва)

Кафедра електронних обчислювальних машин
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА

Пояснювальна записка

Рівень вищої освіти перший (бакалаврський)

Комп'ютерна система обробки зображень з використанням машинного навчання

(тема)

Виконав:
здобувач 4 року навчання,
групи КІУКІ-21-3
Іван ТОМАЩУК
(власне ім’я, прізвище)

Спеціальність
123 «Комп’ютерна інженерія»
(код і повна назва спеціальності)

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

Освітня програма
Комп’ютерна інженерія
(повна назва освітньої програми)

Керівник: ас. Максим БОНДАРЕНКО
(посада, власне ім’я, прізвище)

Допускається до захисту

Завідувач кафедри ЕОМ _____ Андрій КОВАЛЕНКО
(підпис) (власне ім’я, прізвище)

2025 р.

Харківський національний університет радіоелектроніки

Факультет комп’ютерної інженерії та управління

Кафедра електронних обчислювальних машин

Рівень вищої освіти перший (бакалаврський)

Спеціальність 123 «Комп’ютерна інженерія»
(код і повна назва)

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

Освітня програма Комп’ютерна інженерія
(повна назва)

ЗАТВЕРДЖУЮ:
Зав. кафедри _____
(підпис)
“____” _____ 20__ р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві Томащуку Івану Ігоровичу
(прізвище, ім’я, по батькові)

1. Тема роботи Комп'ютерна система обробки зображень з використанням машинного навчання

затверджена наказом по університету від “ 26 ” травня 2025 р. № 424 Ст

2. Термін подання здобувачем роботи до екзаменаційної комісії 17 червня 2025 р.

3. Вхідні дані до роботи
U-Net
машинне навчання
зґорткова нейронна мережа
Глибоке навчання

4. Перелік питань, що потрібно опрацювати у роботі
Аналіз предметної області
Глибоке навчання
Алгоритми глибокого навчання та їх програмна реалізація

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій 13 слайдів

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

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

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Строк / терміни виконання етапів роботи	Примітка
1	Отримання завдання та аналіз літератури	26.05.2025–30.05.2025	
2	Огляд існуючих рішень та алгоритмів	31.05.2025–03.06.2025	
3	Вибір датасетів та архітектури системи	04.06.2025–06.06.2025	
4	Вибір програмних засобів	07.06.2025–08.06.2025	
5	Програмна реалізація	09.06.2025–11.06.2025	
6	Аналіз отриманих результатів	12.06.2025–13.06.2025	
7	Оформлення записки	14.06.2025–16.06.2025	

Дата видачі завдання " 26 " травня 2025 р.

Здобувач _____
(підпис)

Керівник роботи _____ ас. Максим БОНДАРЕНКО
(підпис) (посада, власне ім'я, прізвище)

РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи: 52 с., 10 рис., 2 дод., 27 джерел.

КОМП’ЮТЕРНА СИСТЕМА, ОБРОБКА ЗОБРАЖЕНЬ, МАШИННЕ НАВЧАННЯ, ГЛИБОКЕ НАВЧАННЯ, ЗГОРТКОВА НЕЙРОННА МЕРЕЖА, CNN, PYTHON, GOOGLE COLAB, КЛАСИФІКАЦІЯ ЗОБРАЖЕНЬ, ШТУЧНИЙ ІНТЕЛЕКТ.

Метою кваліфікаційної роботи є розробка комп’ютерної системи обробки зображень з використанням методів машинного навчання та генетичних алгоритмів.

У ході виконання кваліфікаційної роботи здійснено створення програмного забезпечення для обробки зображень, основою якого стали сучасні підходи машинного навчання, включно з методами глибокого навчання та генетичними алгоритмами. Особливу увагу зосереджено на вивченні принципів функціонування алгоритмів навчання глибоких нейронних мереж, що дозволило обґрунтувати доцільність використання еволюційних підходів для оптимізації їхньої структури. У результаті було сформульовано підхід до структурного синтезу згорткових нейронних мереж із залученням генетичних алгоритмів як засобу автоматизованого пошуку ефективної архітектури. Відповідно до сформованих вимог до системи було здійснено обґрунтований вибір технологій і середовищ розробки, що забезпечили реалізацію програмного продукту з належним рівнем функціональності та адаптивності.

ABSTRACT

Bachelor’s thesis: 52 pages, 10 figures, 2 appendices, 27 sources.

COMPUTER SYSTEM, IMAGE PROCESSING, MACHINE LEARNING, DEEP LEARNING, CONVOLUTIONAL NEURAL NETWORK, CNN, PYTHON, GOOGLE COLAB, IMAGE CLASSIFICATION, ARTIFICIAL INTELLIGENCE.

The major goal of this thesis is the development of software tools for image processing using machine learning methods and genetic algorithms.

In order to develop software for processing visual data, built upon contemporary approaches in machine learning, including deep learning techniques and genetic algorithms. Emphasis was placed on the study of the operational principles of deep neural network training algorithms, which made it possible to justify the use of evolutionary strategies for optimizing network structures. As a result, a methodology was proposed for the structural synthesis of convolutional neural networks using genetic algorithms as a means of automated architecture optimization. Based on the defined system requirements, a reasoned selection of technologies and development environments was carried out, ensuring the implementation of a software solution with a sufficient level of functionality and adaptability.

ЗМІСТ

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ 7
ВСТУП 8
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ 10
1.1 Загальні відомості 10
1.2 Огляд аналогів 11
2 ГЛИБОКЕ НАВЧАННЯ 14
2.1 Поняття глибокого навчання 14
2.2 Згорткові нейронні мережі 15
2.3 LSTM 17
2.4 Аналіз аналогів 19
3 ПРОГРАМНА РЕАЛІЗАЦІЯ АЛГОРИТМІВ ГЛИБОКОГО НАВЧАННЯ 23
3.2 Підготовка даних 30
3.3 Аналіз результатів 33
ВИСНОВКИ 37
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ 38
ДОДАТОК А Графічний матеріал кваліфікаційної роботи 41
ДОДАТОК Б Програмний код 49

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

ПЗ – програмні засоби

AI – штучний інтелект

AUC – площа під кривою

CNN – згорткова нейронна мережа

Colab – хмарне середовище Google Colaboratory

CSV – формат табличних даних, розділених комами

GAN – генеративна змагальна мережа

GPU – графічний процесор

PR – точність-повнота

ReLU – випрямлена лінійна одиниця

ResNet – залишкова нейронна мережа

RNN – рекурентна нейронна мережа

ROC – крива операційних характеристик приймача

TPU – тензорний процесор

XAI – пояснюваний штучний інтелект

ВСТУП

Оптимізація топології набула широкого визнання як ефективний інструмент удосконалення конструкційних рішень і підвищення функціональних характеристик інженерних компонентів. Проте сучасні фреймворки цього типу характеризуються високою обчислювальною складністю, зумовленою необхідністю багаторазового проведення аналізу методом скінченних елементів для оцінки якості конструктивних рішень у процесі оптимізації. У відповідь на ці виклики в науковому середовищі зросла зацікавленість у використанні машинного навчання як засобу зниження обчислювального навантаження. Однак більшість запропонованих підходів були протестовані лише на елементарних двовимірних прикладах з низькою геометричною деталізацією, що суттєво обмежує їх застосовність.

Додатковою проблемою є опора на єдину модель машинного навчання, що виконує наскрізне прогнозування та потребує значного обсягу навчальних даних, що ускладнює масштабування до задач з високою роздільною здатністю. У межах цієї дисертаційної роботи пропонується альтернативна стратегія, яка передбачає інтеграцію методів глибокого навчання зі стандартними алгоритмами оптимізації топології. Запропонований підхід передбачає використання кількох нейронних мереж, кожна з яких моделює окремий етап загального процесу оптимізації, що забезпечує більшу узгодженість із традиційними алгоритмами. Застосування розробленої методології продемонстровано як на двовимірних, так і на тривимірних структурах, причому результати засвідчують її переваги в точності прогнозування оптимізованих конфігурацій у порівнянні з існуючими машинно-орієнтованими методами.

Метою роботи є розробка комп’ютерної системи обробки зображень з використанням методів машинного навчання та генетичних алгоритмів.

Завдання:

- аналіз алгоритмів навчання глибоких нейронних мереж;
- розробка алгоритму структурного синтезу згорткових нейронних мереж з використанням генетичних алгоритмів;
- вибір технологій та програмно-апаратних засобів для реалізації ПЗ для обробки зображень;
- розробка програмних засобів оптимізації топології на основі розглянутих алгоритмів.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Загальні відомості

Упродовж останніх десятиліть спостерігається зростання інтересу до створення компонентів з максимальною функціональною ефективністю, серед яких особливе місце займає оптимізація топології. Цей напрям, що належить до галузі методів оптимізації дизайну і був започаткований Бендспе та Кікучі [4], передбачає застосування чисельних підходів для визначення оптимального розподілу матеріалу в межах заданої проєктної області з метою досягнення бажаних геометричних характеристик. Під ефективністю в цьому контексті маються на увазі різноманітні фізичні властивості – від механічної міцності до теплопровідності, динаміки рідини, акустичних, електромагнітних або оптичних характеристик. Проєктна область, як правило, представлена у вигляді дво- або тривимірної сітки, що використовується у комп’ютерному моделюванні, зокрема для аналізу методом скінченних елементів.

Серед численних методів, що застосовуються в оптимізації топології, найбільш відомими є SIMP (твердий ізотропний матеріал із штрафом) [3], метод рівневих множин і підходи еволюційної оптимізації. Ці стратегії реалізуються у задачах, що вимагають досягнення оптимальних функціональних характеристик об'єктів за умов обмеженого об’єму матеріалу, шляхом поетапного вилучення малоефективних зон.

Найсерйознішим викликом при реалізації подібних оптимізацій залишається значна обчислювальна вартість процесу. Ефективність конструкції слід переоцінювати після кожної ітерації оптимізації, що зазвичай потребує використання трудомісткого чисельного моделювання на основі методів розв’язання диференціальних рівнянь, таких як аналіз скінченних елементів. Це призводить до того, що оптимізація геометрій

високої деталізації може тривати від кількох годин до днів. У зв’язку з цим зросла зацікавленість у використанні глибокого навчання як альтернативи, що здатна значно зменшити або повністю усунути потребу в чисельному аналізі.

Хоча глибоке навчання демонструє високі результати в різноманітних сферах, дана робота зосереджується на його застосуванні до задач моделювання фізичних процесів. Зокрема, зростає інтерес до використання нейронних мереж для апроксимації фізичних явищ із мінімальними вимогами до обсягу даних. Один із дієвих підходів полягає у модифікації функції втрат таким чином, щоб у процесі навчання нейромережі враховували фізичні обмеження, наприклад, граничні умови. Подібні методи вже показали високу ефективність у розв’язанні складних диференціальних рівнянь, зокрема рівняння Бюргерса, Нав’є–Стокса та Кана–Хілліарда.

Ці досягнення свідчать про ефективність стратегій, які забезпечують узгодженість між архітектурою нейромереж і фізичною природою моделюваних процесів. Виходячи з такої мотивації, у цій роботі пропонується новий підхід до структурної оптимізації топології, що базується на глибокому навчанні та передбачає алгоритмічну узгодженість між мережею й досліджуваними фізичними процесами.

1.2 Огляд аналогів

Алгоритмічно узгоджена модель глибокого навчання для оптимізації структурної топології (STO) має задовольняти кілька ключових вимог: вона повинна відображати фундаментальні фізичні закономірності, відслідковувати еволюцію топологічних змін у процесі оптимізації, а також формувати результати, що дотримуються заданих геометричних та граничних обмежень. Щоб надати чітку структуру дослідженню, розглянуто три критично важливі компоненти будь-якого підходу, що базується на даних: репрезентацію даних, навчальні алгоритми та структуру моделі. Для

досягнення алгоритмічної узгодженості ці компоненти повинні бути гармонізовані з класичною методологією топологічної оптимізації – зокрема, із застосуванням методу твердого ізотропного матеріалу з штрафом (SIMP).

У межах запропонованої архітектури DLTO вхідними параметрами є початкова відповідність геометрії та цільове значення об’ємної частки. За допомогою цього підходу прогнозується оптимальний розподіл густини без потреби в ітераційних оцінках методом скінченних елементів. Після цього прогнозована щільність трансформується у поверхневу сітку шляхом застосування алгоритму marching cubes, що дозволяє реконструювати остаточну форму об’єкта.

Першим критичним аспектом є вибір представлення, що відповідає алгоритмічній логіці STO. У класичних методах відповідність кожного елемента має ключове значення в аналізі чутливості, що визначає подальші зміни щільності. На цьому тлі саме така відповідність, на відміну від альтернативних підходів – таких як представлення у вигляді воксельної щільності або тензорів деформації, – виявляється більш адекватною та масштабованою для цілей глибокого навчання. Відтак у цій роботі відповідність елемента розглядається як базове представлення геометрії, навантаження та граничних умов.

Далі, навчальна й інференційна частини моделі повинні наслідувати логіку класичного процесу STO. У ході дослідження було встановлено, що еволюція густин у процесі оптимізації є складною і нетривіальною, що унеможливлює пряме навчання від початкової до остаточної топології. Тому як вхідні дані використовуються не лише початкова відповідність і цільовий об’єм, а й проміжні густини, отримані на етапі генерації навчального набору. У ролі цільового значення використовується остаточна щільність.

Модель також повинна відповідати двом основним обмеженням STO: дотриманню цільової об’ємної частки та мінімізації відповідності. Хоча перше з цих обмежень виконується тривіально, обчислення відповідності

зазвичай вимагає аналізу скінченних елементів. Щоб уникнути цього, запропоновано побудову сурогатної моделі, яка дозволяє передбачати відповідність на основі проміжних щільностей.

На основі викладеного сформульовано два варіанти алгоритмічно узгоджених структур: модель послідовного прогнозування щільності (DS) та модель комбінованого прогнозування щільності й відповідності (CDCS). Перший підхід передбачає навчання без урахування відповідності, тоді як другий включає сурогатну модель для прогнозування відповідності на кожному кроці. Обидві моделі порівнюються з базовим підходом прямого прогнозування остаточної щільності (DOD), у якому прогноз базується виключно на початковій відповідності та об’ємній частці.

Модель DS використовує дві згорткові нейронні мережі для прогнозування кінцевої конфігурації, тоді як у CDCS передбачено три згорткові мережі, що працюють ітеративно на етапі висновку, покращуючи точність остаточного результату.

2 ГЛИБОКЕ НАВЧАННЯ

2.1 Поняття глибокого навчання

Глибоке навчання (Deep Learning, DL) є підрозділом машинного навчання (Machine Learning, ML), що, своєю чергою, входить до ширшої категорії штучного інтелекту (Artificial Intelligence, AI). Якщо AI загалом спрямований на моделювання людської поведінки машинами, то ML застосовує статистичні методи для того, щоб алгоритми навчалися на досвіді. DL, в основі якого лежить принцип моделювання нейронної структури мозку людини, має на меті відтворити логіку людського мислення через аналіз структурованих даних.

Основою DL є глибокі нейронні мережі (Deep Neural Networks, DNN) – багаторівневі архітектури, що складаються з взаємопов’язаних штучних нейронів. Ці мережі моделюють складні функціональні залежності між вхідними даними X та очікуваним виходом Y шляхом апроксимації функції F(W,X, де W – параметри мережі (тобто ваги зв’язків). Кожен рівень обчислює свою відповідь уіуі з урахуванням вхідного сигналу хіхі за формулою:

$$yi=a(wi\cdot xi+bi) \tag{2.1}$$

тут a(·) – нелінійна функція активації, яка надає моделі здатність відображати складні залежності, wiwi – ваговий коефіцієнт, а bi – зсув (bias).

Мета навчання – знайти оптимальні значення параметрів W*, які мінімізують різницю між прогнозованим виходом та справжнім значенням YY, тобто значення функції втрат. Цей процес реалізується за допомогою алгоритмів оптимізації, які ґрунтуються на методі зворотного поширення

похибки (backpropagation) з використанням градієнтного спуску. Найпоширенішими варіантами останнього є оптимізатори Adam, RMSProp, Adagrad, Adadelta тощо.

Підвищення ефективності моделі забезпечується низкою технік: пакетна нормалізація (batch normalization), яка стандартизує вихідні значення кожного шару, регуляризація, що вводить штраф за складність моделі для уникнення перенавчання, та вилучення зв'язків (dropout), коли окремі нейронні з'єднання тимчасово вимикаються для покращення узагальнення.

У подальших розділах розглядатимуться найважливіші архітектури глибоких нейронних мереж, зокрема згорткові нейронні мережі (CNN) – ключові для обробки просторових даних, та LSTM – модифікація рекурентних нейронних мереж (RNN), здатна працювати з послідовними залежностями в часі.

2.2 Згорткові нейронні мережі

Як зазначалося раніше, нейронна мережа (N) прагне апроксимувати функцію F(), яка встановлює відповідність між вхідними даними та відповідними виходами. Це досягається шляхом послідовного застосування кількох шарів нейронів. У традиційних мережах кожен нейрон певного шару з'єднаний з усіма нейронами попереднього й наступного шару – так звані повнозв'язані (fully connected) шари. Проте в умовах роботи з великими вхідними обсягами така архітектура призводить до експоненційного зростання кількості параметрів, що ускладнює обчислення та підвищує ризик перенавчання. Це свідчить про те, що не всі зв'язки між нейронами є суттєвими чи корисними.

Замість повного з'єднання, згорткові нейронні мережі застосовують локальні зв'язки: кожен нейрон взаємодіє лише з обмеженим фрагментом вхідного простору – своєю локальною рецептивною зоною (рисунок 2.1). Саме ця особливість визначає суть згорткового шару. У ньому вихідна

активація обчислюється як результат згортки вхідного об’єму з фільтрами – наборами ваг, що повторно застосовуються до кожної локальної ділянки. Завдяки спільному використанню ваг кількість параметрів у CNN значно знижується порівняно з повнозв’язаними шарами.

Таким чином, на згортковому шарі ll, коли обробляється тривимірний вхід (що описується шириною, висотою та кількістю каналів), вихід визначається як результат згорткової операції між вхідними даними та відповідними ядрами згортки (фільтрами), які є сталими по всьому вхідному простору.

Цей механізм, що комбінує просторову локальність, зменшену кількість параметрів та здатність до автоматичного вилучення ознак, робить CNN потужним інструментом для аналізу зображень, обробки сигналів та багатьох інших завдань у сфері глибокого навчання.

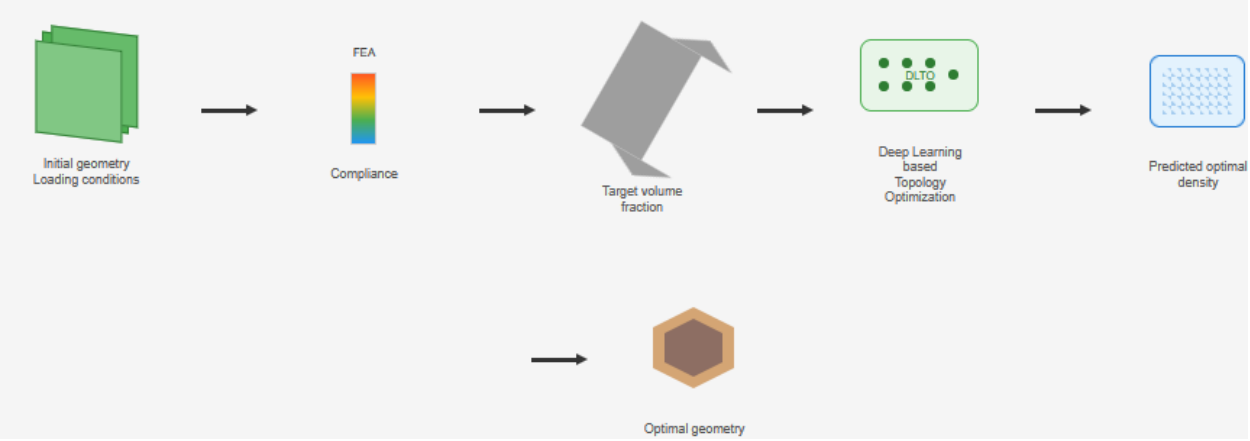

Рисунок 2.1 – Шар ЗНМ

У CNN кожен шар обробляє вхідні дані Xi, застосовуючи операцію згортки з ядром (або фільтром), що має вагові коефіцієнти та зміщення, після чого результат проходить через нелінійне перетворення a(·), тобто функцію активації. Вихід Yi, відомий як карта ознак (feature map), відображає просторові та семантичні характеристики вхідного сигналу на заданому рівні абстракції.

Ядро згортки характеризується формою k1,k2,n,m, де k1 та k2

відповідають просторовій роздільності фільтра (висота і ширина), n – кількість каналів на вході, а m – кількість ядер, які визначають кількість каналів на виході. Це забезпечує здатність CNN витягувати багаторівневі ознаки з вхідних даних.

Окрім згорткових операцій, важливою складовою CNN є операція об’єднання (pooling). Вона зменшує розмірність просторового представлення ознак, зберігаючи при цьому найважливішу інформацію. Найпоширенішим видом об’єднання є максимальне об’єднання (max-pooling), у якому з кожної локальної області вибирається максимальне значення. Цей процес не має параметрів для навчання і сприяє зменшенню обчислювальної складності та зменшенню ймовірності перенавчання.

Зворотною до об’єднання є операція підвищення дискретизації (upsampling), яка застосовується для збільшення просторової роздільності карт ознак, наприклад, коли потрібно відновити вихідну форму зображення або передати дані в умовах сегментації. Підвищення дискретизації необхідне після послідовного застосування операцій, що зменшують розмірність, і забезпечує повернення до бажаної геометричної структури результату.

Згорткові нейронні мережі будуються шляхом послідовного накладання згорткових шарів, шарів об’єднання, шарів підвищення дискретизації та, за необхідності, повнозв’язаних шарів. Ця багаторівнева структура дозволяє моделі ефективно витягати, зменшувати, а потім реконструювати ознаки, що мають високу прогностичну цінність у завданнях розпізнавання образів, класифікації, сегментації та генерації даних.

2.3 LSTM

Як було розглянуто раніше, згорткові нейронні мережі демонструють високу ефективність при обробці просторових структур. Однак у випадках, коли дані мають часову структуру або послідовну природу, доцільнішим є використання рекурентних нейронних мереж, зокрема архітектури

довготривалої короткочасної пам’яті (LSTM). Цей підхід вирішує проблему короткочасної пам’яті, притаманну класичним RNN, де передавання інформації на великі часові інтервали ускладнюється. Модель LSTM, вперше представлена Hochreiter і Schmidhuber у 1997 році, дозволяє зберігати й керувати інформацією в часі завдяки внутрішньому механізму з фіксованою структурою пам’яті – станом комірки.

Центральною ідеєю цієї архітектури є контрольований доступ до пам’яті, що реалізується за допомогою кількох функціональних блоків, які умовно називаються воротами. Ці елементи регулюють, які дані надходять до пам’яті, які зберігаються, а які видаляються. Ворота ґрунтуються на сигмоїдних функціях активації, що надає кожному елементу стану значення в межах [0, 1], інтерпретоване як ступінь збереження або забуття інформації. Під час обробки кожного нового вхідного сигналу модель спочатку визначає, яка частина існуючої пам’яті є неактуальною та повинна бути видалена, а потім – яку нову інформацію потрібно додати, адаптуючи тим самим стан комірки до актуального контексту. Завдяки поєднанню сигмоїдних і гіперболічних тангенсових функцій, ця трансформація забезпечує баланс між пам’яттю та адаптивністю.

У подальшому, оновлений стан пам’яті використовується для формування нового прихованого стану, який впливає як на поточний вихід, так і на обчислення на наступному часовому кроці. Така динамічна взаємодія між станом комірки та прихованим станом дозволяє мережі не лише зберігати релевантні залежності у часі, але й гнучко реагувати на нові дані. Відтак, LSTM виявляється особливо ефективною в тих випадках, де для успішного прогнозування необхідне врахування контексту на тривалих часових інтервалах – наприклад, у моделюванні мовлення, обробці часових рядів або аналізі динамічних процесів.

2.4 Аналіз аналогів

Типовий процес оптимізації топології на основі методу SIMP (твердого ізотропного матеріалу із штрафом) ілюструється як у формі покрокового алгоритму, так і через відповідну блок-схему. Цей підхід передбачає поступову еволюцію геометрії конструкції, яка змінюється на кожному етапі, що наочно представлено праворуч у вигляді серії форм, отриманих протягом ітерацій.

Оптимізаційний процес починається з ініціалізації проєктної області, у якій визначаються навантаження та граничні умови, що задають фізичний контекст задачі. Далі, за допомогою методу скінченних елементів (FEA), виконується обчислення відповідності кожного елемента проєктної сітки. Ці обчислення є критично важливими для наступного кроку – аналізу чутливості, який визначає вплив змін щільності окремих елементів на загальний показник ефективності.

На основі результатів цього аналізу щільності елементів оновлюються в межах кожної ітерації, наближаючи топологію до оптимального розподілу матеріалу. Візуалізація змін на кожному кроці дозволяє не лише контролювати збіжність методу, але й краще розуміти логіку динамічної трансформації конструкції під впливом фізичних і функціональних обмежень.

Процес оптимізації топології методом SIMP триває доти, доки зміни в геометрії проєкту між послідовними ітераціями не зменшаться до прийнятного порогу. Попри відносну простоту реалізації, SIMP не є єдиним підходом: альтернативами є метод рівневих множин і методи еволюційної оптимізації, які також широко застосовуються у практиці структурного проектування. Проте, незалежно від використовуваного методу, всі ці підходи мають спільну характеристику – потребу в численних ітераціях з обчисленням відповідності за допомогою аналізу кінцевих елементів, що значно підвищує обчислювальні витрати. Крім того, ці методи можуть

приводити до множинних, хоч і порівнюваних за якістю, рішень, оскільки задача оптимізації топології, як правило, має неєдине оптимальне рішення.

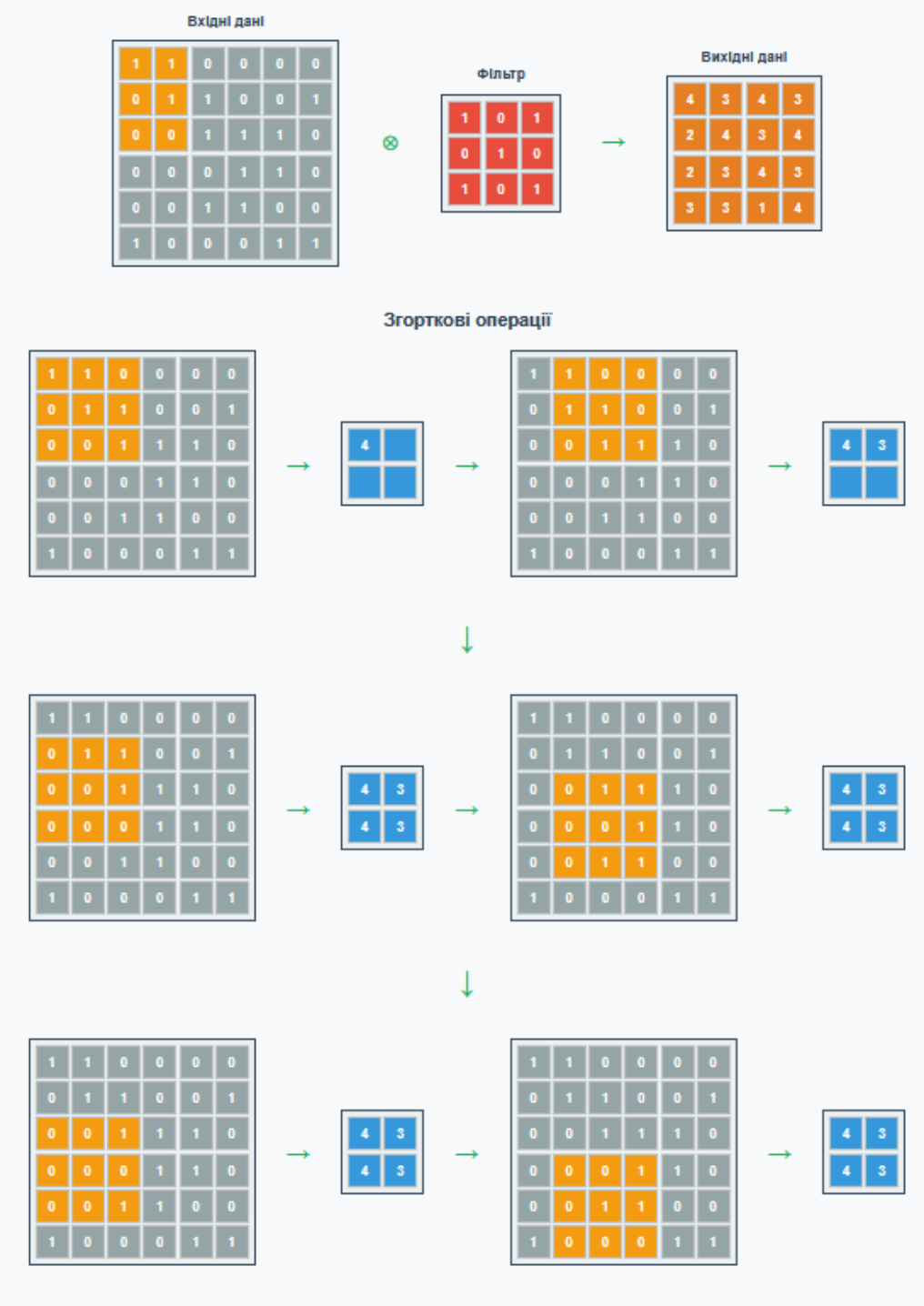

Рисунок 2.2 – Блок-схема

У цьому контексті методи, засновані на глибокому навчанні, мають значний потенціал для прискорення процесу, що стало предметом численних

досліджень. Зокрема, деякі з перших підходів, запропоновані Banga та колегами, а також Сосновиком і Оселедцем, передбачали використання глибоких згорткових автокодерів для тонкого уточнення геометрії, що зазвичай потребує декількох ітерацій аналізу. Підхід Сосновика й Оселедця полягав у використанні щільностей після п’яти ітерацій SIMP як вхідних даних для мережі, яка прогнозує остаточний розподіл щільності. Банга та його команда адаптували цю ідею до тривимірних задач, включаючи граничні умови, проте лише на прикладах з низькою просторовою роздільністю.

Інший підхід передбачав прогнозування цільової форми на основі вхідної геометрії, граничних умов і заданої об’ємної частки. У межах цього ж підходу було реалізовано генеративну модель, яка дозволяє отримати множинні допустимі варіанти. Проте модель виявилася обмеженою через роботу лише з одним типом граничних умов, а її розширення на загальний випадок вимагало би непропорційно великого навчального набору.

З метою подолання цих обмежень Zhang та колеги запропонували вдосконалене представлення вхідних даних, що включало тензори деформацій та зміщень. Вони реалізували модель у двовимірному просторі, представляючи компоненти тензорів як окремі канали вхідного зображення для CNN. Хоча їхній підхід демонструє поліпшення результатів, масштабування до 3D ускладнене через збільшення розміру тензорних представлень і обчислювальне навантаження, зумовлене використанням воксельних сіток.

Останніми роками Чандрасекар і Суреш запропонували новий підхід, у якому нейронна мережа використовується на кожній ітерації процесу для визначення щільності елементів, що забезпечує швидшу збіжність. Однак навіть у цьому варіанті зберігається потреба в FEA-обчисленнях, хоча й менш інтенсивна.

Попри всі ці досягнення, залишається мало робіт, у яких би розглядалося ефективне використання інформації про відповідність і

проміжні щільності в процесі навчання. Крім того, більшість наявних підходів обмежені 2D-представленнями або дуже низькою роздільністю в 3D. Це створює нагальну потребу в масштабованій тривимірній структурі глибокого навчання, алгоритмічно узгодженій з класичними методами оптимізації топології.

3 ПРОГРАМНА РЕАЛІЗАЦІЯ АЛГОРИТМІВ ГЛИБОКОГО НАВЧАННЯ

3.1 Реалізація мережі

базова конфігурація глибокого навчання, що використовується для порівняння з розробленими структурами. Крім того, оцінюється ефективність як запропонованих моделей, так і цієї базової лінії у співставленні з класичним методом SIMP. Після викладу концепції прямого прогнозування розглядаються дві авторські схеми: прогнозування послідовності щільності (DS) і прогнозування пов’язаної послідовності щільності та відповідності (CDCS). На одному з рисунків демонструється, як запропоновані рішення структурно та алгоритмічно узгоджуються з класичною процедурою SIMP.

Рисунок 3.1 – Топологія мережі

Базовий підхід ґрунтується на U-Net – архітектурі, що довела свою ефективність у завданнях сегментації та реконструкції зображень. У цьому контексті мережа отримує на вхід два тензори однакової форми: перший представляє початкову відповідність, а другий – константний тензор, елементи якого дорівнюють заданій об’ємній частці. На вхідних шарах послідовно виконуються згортка та нормалізація, після чого дані поетапно зменшуються в розмірності та зберігаються для подальших з’єднань у структурі пропусків. Після завершення зворотного проходження (підвищення дискретизації) мережа формує остаточну щільність, яка наближається до бажаного розподілу матеріалу.

Для підготовки даних передбачено логарифмічну трансформацію відповідності з наступним масштабуванням значень у діапазон [0,1]. Щоб зробити модель нечутливою до різних просторових орієнтацій навантаження, здійснюється обширне геометричне доповнення вхідного тензора за рахунок поворотів та дзеркального відображення по трьох основних площинах. Метою моделі є побудова двійкового поля щільності, де одиниця означає наявність матеріалу, а нуль – його відсутність. На відміну від більш складних методів, у даному випадку мережа навчається лише на основі початкової відповідності та цільової щільності, без проміжних щільностей або значень відповідності, що робить підхід наскрізним. Навчання здійснюється за допомогою оптимізатора Adam із адаптивною швидкістю навчання, а функцією втрат виступає бінарна крос-ентропія.

У традиційній реалізації SIMP кожна ітерація містить етапи FEA, аналізу чутливості, фільтрації та оновлення щільності. На відміну від цього, підхід DLTO пропонує замінити повторювані етапи оцінки відповідності та оновлення щільності на єдиний блок прогнозування, що забезпечує значне зниження обчислювального навантаження та підвищує ефективність процесу.

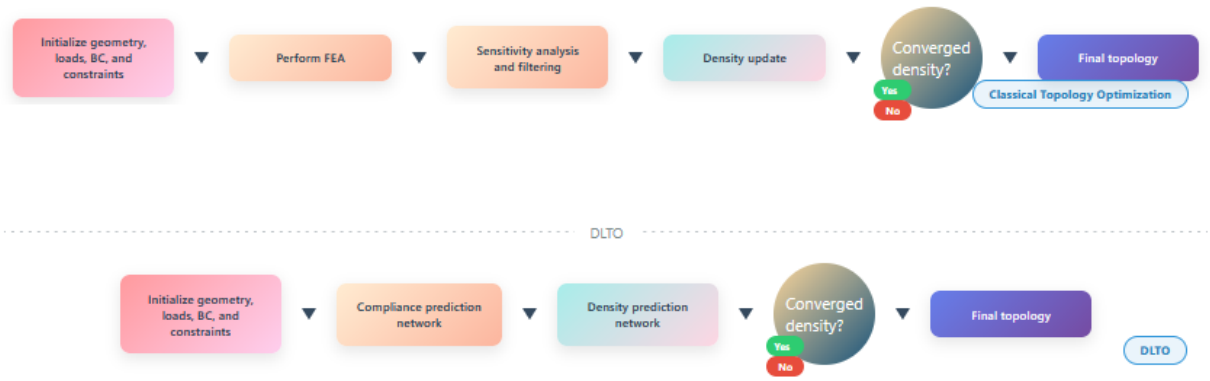

Рисунок 3.2 – U-net

З метою досягнення алгоритмічної узгодженості в поданні даних, оптимізація структурної топології в цій роботі здійснюється на основі відповідності, яка характеризує початкову геометрію. Водночас, варто зазначити, що відповідність змінюється впродовж ітераційного процесу, як і щільність матеріалу. Унаслідок цього між початковою відповідністю та остаточною оптимізованою щільністю не існує прямого функціонального зв'язку, що ускладнює задачу прямого прогнозування.

Для підвищення точності передбачень розроблено дворівневий фреймворк, візуалізований на рисунку 3.3. На першому етапі задіюється мережа прогнозування початкової щільності (IDPN), яка генерує попередній розподіл щільності на основі відповідності, пов'язаної з вихідною геометрією. Цей етап виконує первинне наближення до оптимізованого стану, враховуючи лише локальну відповідність елементів на старті.

Далі, з урахуванням того, що оптимізація топології є процесом поступового трансформування щільності, використовується другий компонент – мережа перетворення щільності (DTN). Вона бере на вхід попередню щільність, сформовану IDPN, і трансформує її в остаточну конфігурацію, яка максимально наближається до результату класичного методу оптимізації. Варто наголосити, що DTN функціонує незалежно від відповідності, зосереджуючи увагу лише на динаміці щільності в межах ітераційного процесу.

Таким чином, загальна структура включає дві архітектури, кожна з яких відображає окрему фазу: прогнозування початкової щільності та її трансформацію у фінальну оптимізовану форму. Завдяки поділу завдання на два послідовні етапи вдається досягти більшої точності прогнозів, оскільки враховується як первинна відповідність, так і характер змін, що відбуваються в процесі оптимізації.

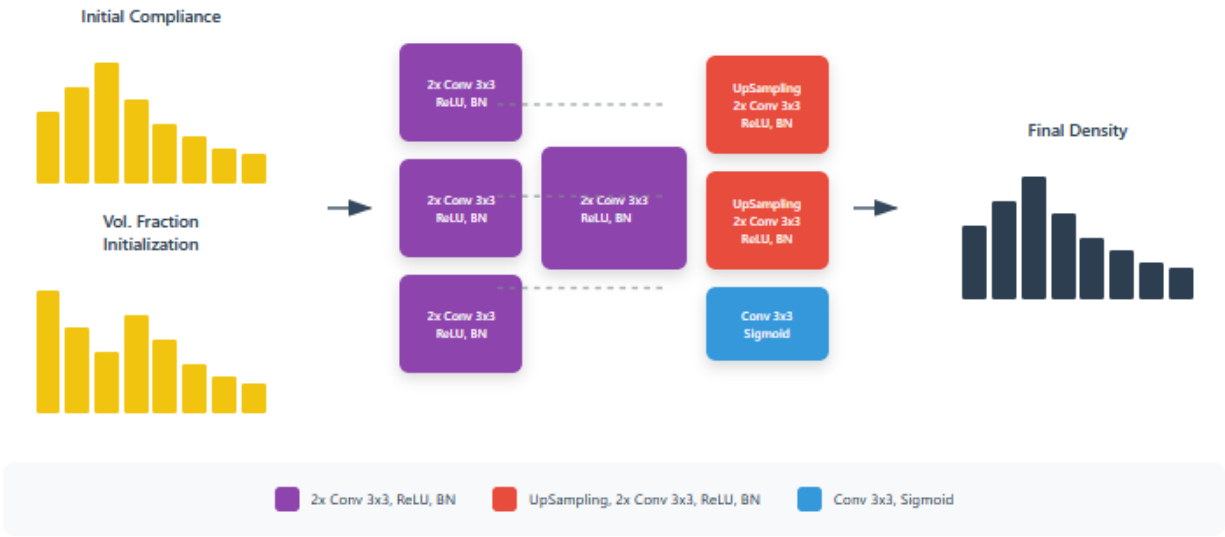

Рисунок 3.3 – DS

Прогнозування послідовності щільності (DS): у цій структурі ми У межах даного дослідження реалізовано дворівневу стратегію прогнозування, яка дозволяє ефективно моделювати оптимізацію структурної топології за допомогою глибокого навчання. На першому етапі система приймає як вхідні дані початкову карту податливості та ініціалізовану об’ємну частку, на основі яких виконується передбачення початкової щільності топології. Цей етап реалізується за допомогою архітектури U-Net, яка забезпечує збереження просторової структури вхідних даних, а її вихід репрезентує щільність після першої ітерації класичного методу SIMP. У двовимірному випадку вхідними каналами виступають відповідність і об’ємна частка, тоді як у тривимірному – тензори з двома об’ємними каналами.

Другий етап пов’язаний із прогнозуванням динаміки щільності у часі.

Тут застосовується модель, яка поєднує згорткові нейронні мережі з елементами довготривалої короткочасної пам’яті (CNN-LSTM), що дозволяє захоплювати як просторову структуру, так і часові залежності, притаманні ітераційному процесу оптимізації. На вхід цьому модулю подається послідовність щільностей, отриманих протягом кількох ітерацій, причому з метою ефективності використовуються лише унікальні представлення, визначені через норму L2. CNN-компонент моделі відповідає за витяг ознак, тоді як LSTM-модуль фіксує логіку їхнього перетворення у часі. Вихідна щільність формується за допомогою декодера з рівнів деконволюції.

Навчання обох мереж відбувається з урахуванням специфіки задачі. Для IDPN застосовуються дві функції втрат – середньоквадратична помилка для кожного елемента і додаткова функція, яка забезпечує відповідність загальної об’ємної частки. DTN навчається з урахуванням додаткового обмеження на бінарність щільностей: для цього додається функція бінарної крос-ентропії. Обидві моделі оптимізуються за допомогою алгоритму Adam зі стохастичним градієнтним спуском, а на етапі підготовки даних реалізовано обширне доповнення, включно з геометричними трансформаціями.

Після завершення тренування параметри обох моделей інтегруються в єдиний конвеєр, який забезпечує наскрізне прогнозування: достатньо задати лише початкову відповідність та цільову об’ємну частку. Такий підхід дозволяє передбачати оптимальну щільність без необхідності у проміжному чисельному аналізі, що суттєво знижує обчислювальні витрати.

У межах роботи також розглянуто третій, більш складний підхід, який надихнутий ітераційною природою методу SIMP. З урахуванням того, що перші кілька ітерацій демонструють найбільші трансформації (грубе уточнення), було запропоновано три спеціалізовані мережеві архітектури, здатні обробляти як проміжні відповідності, так і проміжні щільності для досягнення високоточної реконструкції фінального рішення. Цей підхід ілюструє потенціал нейромережевих моделей для відтворення як грубих, так

і тонких аспектів еволюції топології, що забезпечує високу якість та надійність результатів.

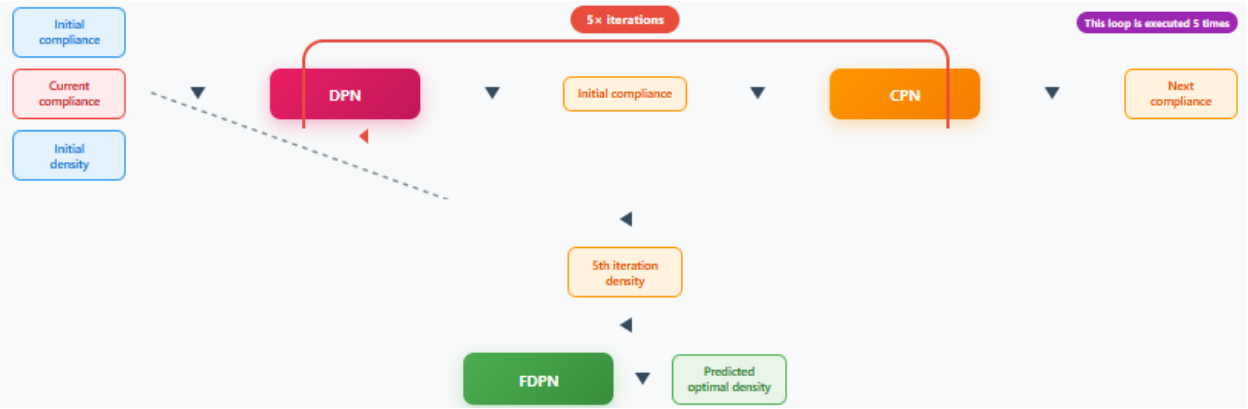

Рисунок 3.4 – CDSC

У межах структури прогнозування зв’язаної послідовності щільності та відповідності (CDCS) реалізовано комплексний підхід до моделювання процесу топологічної оптимізації, який відтворює багатокрокову взаємодію між фізичною відповідністю та розподілом матеріалу. Структура поєднує три ключові компоненти: мережу прогнозування щільності (DPN), мережу прогнозування відповідності (CPN) та фінальну мережу прогнозування щільності (FDPN), організовані в ітеративний ланцюг з п’яти кроків.

На першому етапі ініціалізується початкова відповідність і об’ємна частка, з яких формується перша карта щільності. Далі мережа DPN, що ґрунтується на архітектурі U-SE-ResNet, прогнозує нову щільність на основі поточних вхідних даних. Особливістю цієї архітектури є інтеграція механізму Squeeze-and-Excitation, який дозволяє моделювати вагу каналів ознак, зберігаючи при цьому переваги архітектури U-Net.

Водночас мережа CPN, реалізована на основі архітектури типу Encoder-Decoder, передбачає відповідність, що відповідає новій щільності. Її кодувальна частина знижує розмірність вхідного простору до латентного представлення, яке потім декодується у просторову карту відповідності. Для кожної ітерації прогнозовані щільність і відповідність оновлюються у

взаємозалежній послідовності, що повторюється п’ять разів, відтворюючи логіку класичної оптимізації SIMP.

Після п’яти ітерацій цикл завершується, і отримана щільність подається на вхід фінальної мережі FDPN, яка прогнозує остаточний розподіл щільності. Цей модуль побудований на архітектурі U-Net і має спільну структуру з CPN, що забезпечує архітектурну узгодженість усієї моделі.

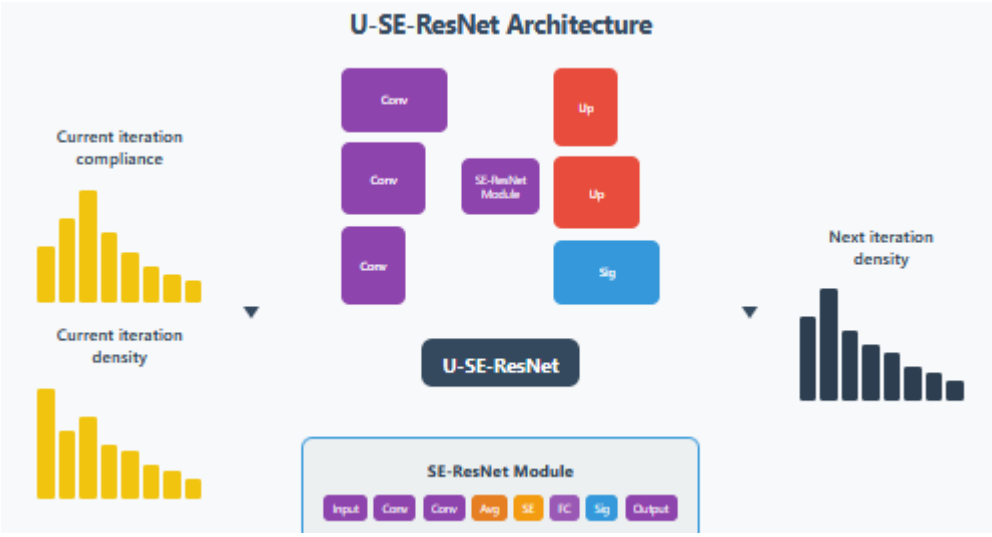

Рисунок 3.5 – DPN

Для стабільності та коректності роботи всіх мереж відповідність нормалізується до діапазону [0,1], а дані піддаються систематичному розширенню з використанням геометричних перетворень. Усі три мережі навчаються окремо із застосуванням оптимізатора Adam та адаптивної швидкості навчання. DPN і FDPN, що працюють із бінарними картами щільності, навчаються за допомогою функції втрат на основі перехресної ентропії, тоді як для CPN використовується середня абсолютна похибка.

На етапі висновку запускається повний ітераційний цикл: з початкової податливості та щільності формується перша ітерація, яка поетапно оновлюється, доки не буде отримано п’яту карту щільності. Саме вона подається на вхід FDPN, що забезпечує остаточний прогноз без необхідності

в подальших фізичних обчисленнях.

Цей підхід поєднує ітераційну логіку класичної топологічної оптимізації з перевагами глибокого навчання, забезпечуючи точність і масштабованість за значно меншої обчислювальної вартості.

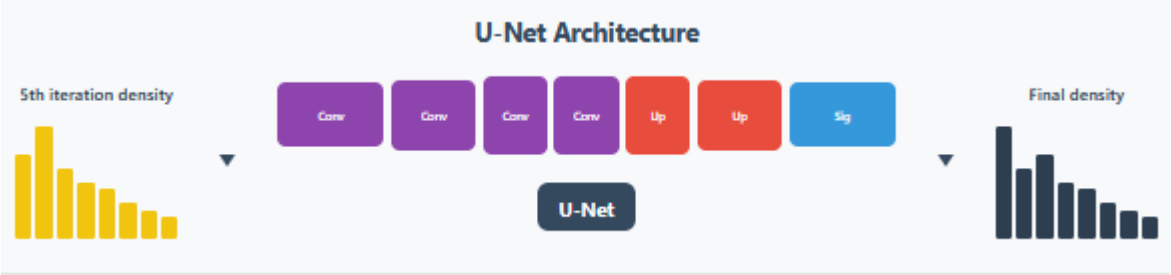

Рисунок 3.6 – FDPN

3.2 Підготовка даних

У процесі підготовки навчальних даних для нейронних мереж було здійснено масштабне моделювання оптимізації топології із залученням широкого спектра конструктивних варіантів та обмежень об’ємної частки. Кожну конструкцію було формалізовано у вигляді регулярної двовимірної сітки, елементи якої є чотирикутними, що дало змогу однозначно зіставити кожен елемент із пікселем відповідного зображення. У такий спосіб просторову структуру геометрії було адаптовано для подання у форматі, зручному для використання у згорткових нейронних мережах, де інтенсивність пікселя відображає відповідність або щільність відповідного елемента сітки. Необхідна для навчання інформація включає необроблені значення відповідностей, цільові обмеження об’ємної частки, проміжні щільності елементів, отримані на різних етапах ітераційного процесу оптимізації, а також остаточний розподіл щільності.

Для генерації вказаних даних було виконано понад 30 тисяч симуляцій структурної топологічної оптимізації із випадково варійованими параметрами навантаження:його напрямком, величиною, місцем

прикладання та довільним розташуванням фіксованих вузлів із нульовими зміщеннями, що забезпечило високу варіативність вхідних умов. Кожна симуляція тривала до 150 ітерацій відповідно до алгоритму SIMP, а отримані результати були збережені для подальшого використання у процесі навчання.

У контексті тривимірного аналізу, що реалізується в межах підходу DLTO, було використано ANSYS Mechanical APDL v19.2. Початкову область проєктування визначали у вигляді куба з довжиною ребра один метр, який дискретизували у вигляді тривимірної сітки з понад 30 тисячами вузлів і понад 150 тисячами твердотільних елементів, кожен з яких складається з восьми вузлів. Щоб забезпечити репрезентативне покриття різних топологій у вибірці, було застосовано велику множину граничних умов і варіантів навантаження, доступних у середовищі ANSYS, включаючи точкові, поверхневі та розподілені сили, тиск, моменти та фіксацію вузлів. Для виключення жорсткотілих переміщень на початковому етапі вибиралися три випадкові неколінеарні вузли на одній грані куба, положення яких фіксувалися. Подальше навантаження прикладалося в довільно вибраній точці, що не належала до області фіксації, із випадковими характеристиками сили та напрямку.

Таке комплексне моделювання дало змогу створити різноманітний, структурно насичений набір даних, придатний для навчання моделей глибокого навчання в контексті оптимізації топології.

Для забезпечення унікальності кожного прикладу в навчальному наборі даних було застосовано стратегію вибірки з відхиленням, що дозволило уникнути дублювання топологічних конфігурацій. У результаті було сформовано 1500 унікальних комбінацій навантажень і граничних умов, з яких шляхом варіювання об’ємної частки було отримано 13500 повноцінних зразків. Згідно зі статистикою, процес оптимізації топології в середньому потребував 13 ітерацій, хоча в окремих випадках кількість ітерацій коливалася від 6 до 72, що пояснюється варіативністю просторової дискретизації, граничних умов і встановлених цільових параметрів. Усі

релевантні результати – від проміжних щільностей до енергії деформації – зберігалися у форматі, сумісному з первинним сітковим представленням.

```
# 🧠 Встановлення залежностей
!pip install deap tensorflow numpy matplotlib
```

```
import tensorflow as tf
from tensorflow.keras import layers, models
import numpy as np
import matplotlib.pyplot as plt
from deap import base, creator, tools, algorithms
import random
```

```
# 📥 Завантаження та підготовка даних (CIFAR-10)
(x_train, y_train), (x_test, y_test) = tf.keras.datasets.cifar10.load_data()
x_train, x_test = x_train / 255.0, x_test / 255.0
```

```
# 🦴 Побудова CNN за хромосомою
def build_model(genes):
    model = models.Sequential()
    model.add(layers.Input(shape=(32, 32, 3)))
    for num_filters in genes:
        model.add(layers.Conv2D(num_filters, (3, 3), activation='relu', padding='same'))
        model.add(layers.MaxPooling2D((2, 2)))
    model.add(layers.Flatten())
    model.add(layers.Dense(64, activation='relu'))
    model.add(layers.Dense(10, activation='softmax'))
    model.compile(optimizer='adam',
                  loss='sparse_categorical_crossentropy',
                  metrics=['accuracy'])
    return model
```

```
# ⚙️ Оцінка хромосоми
def evaluate(individual):
    model = build_model(individual)
    history = model.fit(x_train[:2000], y_train[:2000], epochs=3, batch_size=64, verbose=0)
    _, acc = model.evaluate(x_test[:1000], y_test[:1000], verbose=0)
    return (acc,)
```

```
# 🚀 Налаштування генетичного алгоритму
creator.create("FitnessMax", base.Fitness, weights=(1.0,))
creator.create("Individual", list, fitness=creator.FitnessMax)
toolbox = base.Toolbox()
toolbox.register("attr_filter", random.choice, [16, 32, 64, 128])
toolbox.register("individual", tools.initRepeat, creator.Individual, toolbox.attr_filter, n=3)
```

Рисунок 3.7 – Реалізація

Для адаптації даних до вимог тривимірного глибокого навчання, початкове тетраедричне сіткове представлення було перетворено у формат регулярної воксельної сітки. Цей процес передбачав побудову регулярної просторової сітки на основі обмежувального паралелепіпеда з наступним визначенням розміщення кожного вокселя відносно елементів тетраедричної сітки. За допомогою барицентричних координат було визначено, чи належить центр вокселя відповідному тетраедру, після чого здійснювалася інтерполяція значень фізичних полів, таких як щільність і енергія деформації, від вузлів елемента до центру вокселя. У середньому цей процес займав від 5 до 15 хвилин на один зразок, залежно від складності геометрії й щільності сітки, а завдяки використанню інструмента GNU Parallel вдалося значно

скоротити загальний час вокселізації шляхом паралельної обробки.

Обчислення елементарної податливості ґрунтувалося на класичному підході SIMP, де відповідне значення отримувалося шляхом піднесення щільності до третього степеня та множення на енергію деформації ($C = \rho^3 \cdot SE$). Після завершення етапу вокселізації проводилася додаткова попередня обробка: зокрема, нормалізація відповідності за максимальним значенням і логарифмічне перетворення відповідних величин задля покращення збіжності під час навчання. Крім того, для підвищення генералізаційної здатності моделі було реалізовано доповнення даних у реальному часі шляхом обертання кожного зразка в усіх шести можливих орієнтаціях. У результаті цих процедур було сформовано повноцінний, збалансований та репрезентативний набір тривимірних навчальних даних для подальшої обробки за допомогою нейронних мереж.

3.3 Аналіз результатів

Вхідними даними для мережі є кольорові зображення 64 на 64 пікселі, що розділені на чотири класи. Програма генерує структури згорткових нейронних мереж за допомогою методу повного перебору. Програма генерує структури згорткових нейронних мереж для всіх комбінацій, потім, кожна отримана мережа навчається протягом двадцяти епох на даних.

Кожна з моделей оцінюється на тестових даних. Для кожної моделі виводиться точність та ROC-крива. Дані комбінації передбачають шість структур згорткових нейронних мереж Після завершення навчання всіх моделей, користувачу виводиться комбінація параметрів, що показали найкращі результати.

Для кількісного порівняння ефективності запропонованих методів з базовою стратегією DOD першим кроком стало дослідження дотримання обмеження на об’ємну частку (VF). Прогнозована об’ємна частка для кожної остаточної топології була отримана шляхом усереднення значень щільності по всій проєктній області. Далі для кожного методу було обчислено середньоквадратичну похибку (MSE) між цими прогнозованими значеннями та відповідними еталонними значеннями VF, отриманими з методу SIMP.

З метою оцінки узгодженості між прогнозованими та фактичними значеннями об’ємної частки було також побудовано графік кореляції, а коефіцієнт кореляції Пірсона обчислено для даних двовимірного тестового набору. Отримані результати вказують на перевагу методу CDCS у здатності задовольняти обмеження на VF, що підтверджується більш високою стабільністю та точністю відповідно до гістограм на рисунку 3.8. Хоча методи DOD та CDCS демонструють порівнювану кореляцію між прогнозованими та фактичними значеннями (відповідно R = 0,8986 і R = 0,8945), метод DS показує значно нижчу здатність до точного відтворення об’ємної частки (R = 0,6883).

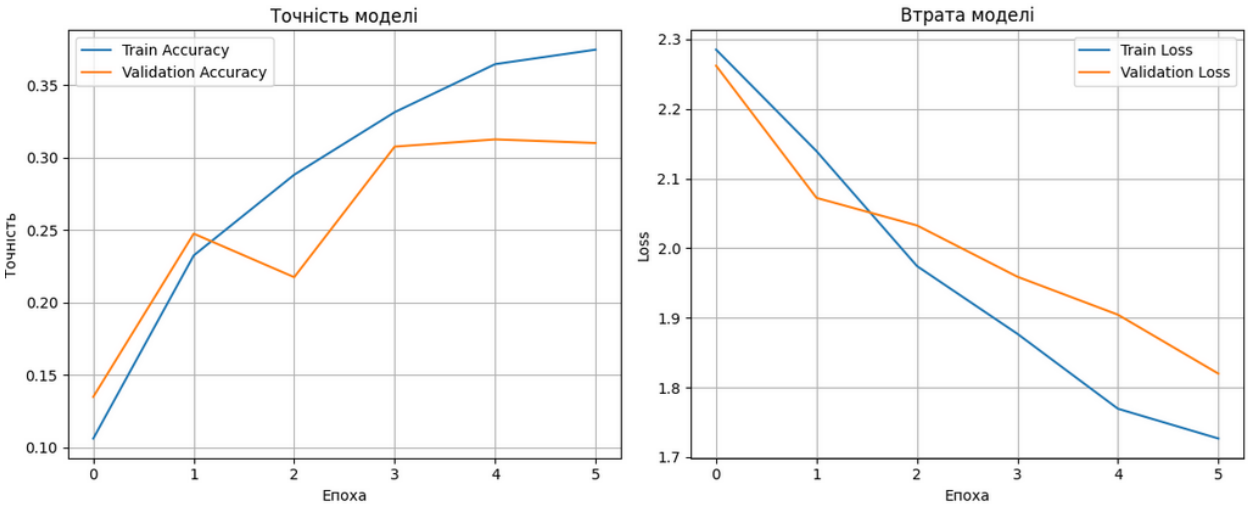

Рисунок 3.8 – Обговорення результатів

Після цього ефективність моделей було додатково оцінено з погляду фізичної відповідності – показника, відомого як загальна відповідність (TC). Цей показник відображає цільову функцію в задачі оптимізації топології і характеризує здатність моделі мінімізувати комплаєнс при збереженні встановлених обмежень. Для оцінювання TC було використано мережу прогнозування відповідності, яка є складовою частиною структури CDCS, з подальшим сумуванням елементарних значень по всій області. Це дозволяє зіставити прогнозовану відповідність з еталонним значенням TC, обчисленим методом SIMP.

Загальна відповідність (TC), що відіграє роль цільової функції в алгоритмі SIMP, слугує ключовим критерієм ефективності оптимізації, оскільки вона відображає загальну податливість структури при дотриманні обмежень об’ємної частки. Для порівняння результатів було обчислено середньоквадратичну похибку (MSE) між прогнозованими та фактичними значеннями TC для кожного з розглянутих підходів. Прогнозування TC здійснювалося через підсумовування елементарних значень відповідності, отриманих за допомогою мережі прогнозування відповідності (CPN), що входить до складу структури CDCS. Як орієнтир використовувалося мінімальне значення TC, досягнуте за результатами ітерацій методу SIMP.

Аналіз результатів засвідчив, що методи DS і CDCS демонструють значно нижчу похибку MSE порівняно з базовим методом DOD – щонайменше на порядок. Це спостереження було підтверджено побудовою гістограми MSE для 2D тестових даних (див. рисунок 6.2(b)). Також виконано кореляційний аналіз між прогнозованими та фактичними значеннями ТС, який показав сильну узгодженість для підходів DS (коефіцієнт Пірсона R = 0,9551) та CDCS (R = 0,8926), тоді як для DOD кореляція виявилася нижчою (R = 0,8403).

Задля всебічного порівняння проведено статистичний аналіз похибок MSE як щодо топологічного обмеження VF, так і фізичної відповідності ТС. Результати цього аналізу показали, що методи CDCS та DOD мають схожі мінімальні, середні та максимальні значення MSE для VF, при цьому вони дещо переважають метод DS. Водночас для ТС методи DS і CDCS суттєво переважають DOD за всіма трьома статистичними показниками, демонструючи подібні середні та максимальні втрати, при цьому DS має найнижче мінімальне значення.

Крім того, для щільнісних характеристик було здійснено порівняння за точністю та трьома основними метриками втрат – двійковою перехресною ентропією (BCE), середньою абсолютною похибкою (MAE) та MSE – між прогнозованими значеннями щільності та базовими еталонними конфігураціями. У цьому порівнянні CDCS знову підтвердив свою перевагу над DOD, забезпечуючи більш точне наближення до оптимальних топологічних рішень.

ВИСНОВКИ

В ході виконання кваліфікаційної роботи розроблена комп’ютерна обробки зображень з використанням методів машинного навчання, зокрема глибокого навчання, та генетичних алгоритмів. Проведено аналіз алгоритмів навчання глибоких нейронних мереж, запропоновано алгоритм структурного синтезу згорткових нейронних мереж з використанням генетичних алгоритмів; обрано технології та програмні засоби для реалізації програмного забезпечення.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Abueidda, D.W., Koric, S., Sobh, N.A., 2020. Topology optimization of 2D structures with nonlinearities using deep learning. Computers & Structures 237, 106283.

2. Banga, S., Gehani, H., Bhilare, S., Patel, S., Kara, L., 2018. 3D topology optimization using convolutional neural networks. arXiv preprint arXiv:1808.07440 .

3. Bendspe, M.P., 1989. Optimal shape design as a material distribution problem. Structural optimization 1, 193-202.

4. Bendspe, M.P., Kikuchi, N., 1988. Generating optimal topologies in structural design using a homogenization method. Computer Methods in Applied Mechanics and Engineering .

5. Bhatnagar, S., Afshar, Y., Pan, S., Duraisamy, K., Kaushik, S., 2019. Prediction of aerodynamic flow fields using convolutional neural networks. Computational Mechanics , 1-21.

6. Bujny, M., Aulig, N., Olhofer, M., Duddeck, F., 2018. Learning-based topology variation in evolutionary level set topology optimization, in: Proceedings of the Genetic and Evolutionary Computation Conference, pp. 825-832.

7. Chandrasekhar, A., Suresh, K., 2021. Tounn: Topology optimization using neural networks. Structural and Multidisciplinary Optimization 63, 1-15. doi:10.1007/s00158-020-02748-4.

8. (Cicek, O., Abdulkadir, A., Lienkamp, S.S., Brox, T., Ronneberger, O., 2016. 3D U-Net: learning dense volumetric segmentation from sparse annotation, in: International conference on medical image computing and computer-assisted intervention, Springer. pp. 424-432.

9. Das, R., Jones, R., Xie, Y.M., 2011. Optimal topology design of industrial structures using an evolutionary algorithm. Optimization and Engineering 12, 681-717.

10. De, S., Hampton, J., Maute, K., Doostan, A., 2019. Topology optimization under uncertainty using a stochastic gradient-based approach. arXiv:1902.04562.

11. Doi, S., Sasaki, H., Igarashi, H., 2019. Multi-Objective Topology Optimization of Rotating Machines Using Deep Learning. IEEE Transactions on Magnetics 55, 1-5. doi:10.1109/TMAG. 2019.2899934.

12. Eschenauer, H.A., Olhoff, N., 2001. Topology optimization of continuum structures: a review. Appl. Mech. Rev. 54, 331-390.

13. Guo, T., Lohan, D.J., Cang, R., Ren, M.Y., Allison, J.T., 2018. An indirect design representation for topology optimization using variational autoencoder and style transfer, in: 2018 AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, p. 0804.

14. Hamdia, K.M., Ghasemi, H., Bazi, Y., AlHichri, H., Alajlan, N., Rabczuk, T., 2019. A novel deep learning based method for the computational material design of flexoelectric nanostructures with topology optimization. Finite Elements in Analysis and Design 165, 2130. URL: https://www.sciencedirect.com/science/article/pii/S0168874X1930023X, doi:https://doi.org/10.1016/j.finel.2019.07.001.

15. Hochreiter, S., Schmidhuber, J., 1997. Long Short-Term Memory. Neural Comput. 9, 1735-1780. URL: https://doi.org/10.1162/neco.1997.9.8.1735, doi:10.1162/neco. 1997.9.8.1735.

16. Hu, J., Shen, L., Sun, G., 2018. Squeeze-and-Excitation Networks, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7132-7141. doi:10.1109/CVPR. 2018.00745.

17. Ioffe, S., Szegedy, C., 2015. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift, in: Bach, F., Blei, D. (Eds.), Proceedings of the 32nd International Conference on Machine Learning, PMLR, Lille, France. pp. 448-456. URL: http://proceedings.mlr.press/v37/ioffe15.html.

18. Jagtap, A.D., Karniadakis, G.E., 2019. Adaptive activation functions

accelerate convergence in deep and physics-informed neural networks. arXiv preprint arXiv:1906.01170 .

19. Jang, S., Kang, N., 2020. Generative design by reinforcement learning: Maximizing diversity of topology optimized designs. arXiv:2008.07119.

20. Kingma, D.P., Ba, J., 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 .

21. Kollmann, H.T., Abueidda, D.W., Koric, S., Guleryuz, E., Sobh, N.A., 2020. Deep learning for topology optimization of 2D metamaterials. Materials & Design 196, 109098. doi:https: //doi.org/10.1016/j.matdes.2020.109098.

22. Krizhevsky, A., Sutskever, I., Hinton, G.E., 2017. ImageNet Classification with Deep Convolutional Neural Networks. Commun. ACM 60, 84-90. URL: https://doi.org/10.1145/ 3065386, doi:10.1145/3065386.

23. Kukacka, J., Golkov, V., Cremers, D., 2017. Regularization for Deep Learning: A Taxonomy. arXiv:1710.10686.

24. Lagaros, N., Kallioras, N., Kazakis, G., 2020. Accelerated topology optimization by means of deep learning. Structural and Multidisciplinary Optimization 62. doi:10.1007/ s00158-020-02545-z.

25. Lecun, Y., Bottou, L., Bengio, Y., Haffner, P., 1998. Gradient-based learning applied to document recognition. Proceedings of the IEEE 86, 2278-2324. doi:10.1109/5.726791.

26. Lee, S., Kim, H., Lieu, Q.X., Lee, J., 2020. CNN-based image recognition for topology optimization. Knowledge-Based Systems 198, 105887. URL: https: //www.sciencedirect.com/science/article/pii/S0950705120302379, doi:https://doi. org/10.1016/j.knosys.2020.105887.

27. Lin, Q., Hong, J., Liu, Z., Li, B., Wang, J., 2018. Investigation into the topology optimization for conductive heat transfer based on deep learning approach. International Communications in Heat and Mass Transfer 97. doi:https://doi.org/10.1016/j.icheatmasstransfer.2018. 07.001.