

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Західноукраїнський національний університет
Факультет комп'ютерних інформаційних технологій
Кафедра комп'ютерної інженерії

Приходько Олександр Валерійович

**Алгоритми реконструкції зображень на основі
штучних нейронних мереж / Algorithms for image
reconstruction based on artificial neural networks**

спеціальність: 123 - Комп'ютерна інженерія
освітньо-професійна програма - Комп'ютерна інженерія
Кваліфікаційна робота

Виконав студент групи КІм-21
О.В. Приходько

Науковий керівник:
к.т.н., доц. Н.Я. Савка

Кваліфікаційну роботу допущено
до захисту:

" ___ " _____ 20__ р.

Завідувач кафедри
_____ Л. О. Дубчак

Тернопіль – 2025

АНОТАЦІЯ

Приходько О.В. Алгоритми реконструкції зображень на основі штучних нейронних мереж. – Рукопис.

Кваліфікаційна робота на здобуття освітнього ступеня «магістр» за спеціальністю 123 «Комп'ютерна інженерія», освітньо-професійна програма. Західноукраїнський національний університет, Тернопіль, 2025.

Робота написана обсягом 76 сторінок і містить 30 рисунків, 1 таблицю, 1 додаток та 41 джерело за переліком посилань.

Метою роботи є розробка алгоритмів реконструкції зображень на основі нейронних мереж.

Проведено аналіз задачі реконструкції зображень, виявлено основні труднощі. Досліджено методи реконструкції зображень й обґрунтовано доцільність використання нейромережових технологій для відновлення зображень з артефактами. Охарактеризовано існуючі архітектури штучних нейронних мереж, виокремлено генеративну модель нейронних мереж як перспективну для реконструкції зображень. Описано особливості функціонування генеративних нейромереж та алгоритм їх навчання. Розроблено алгоритм підготовки даних для навчання генеративної нейромережі. Розроблено архітектуру генеративної мережі для реконструкції зображень. Розроблено програмну систему для реконструкції зображення з користувацьким інтерфейсом. Виконано дослідження адекватності розробленої архітектури для задач реконструкції зображень.

Практичне значення – програмна система для реконструкції зображень з артефактами.

Ключові слова: ЗГОРТКОВІ НЕЙРОННІ МЕРЕЖІ, МАШИННЕ НАВЧАННЯ, GAN-МЕРЕЖІ, КЛАСИФІКАЦІЯ ЗОБРАЖЕНЬ, РЕКОНСТРУКЦІЯ ЗОБРАЖЕНЬ.

ANNOTATION

Prykhodko O.V. Algorithms for image reconstruction based on artificial neural networks. – Manuscript.

Qualification work for the Bachelor degree in specialty 123 “Computer Engineering”, educational and professional program. Western Ukrainian National University, Ternopil, 2025.

The work is written in 76 pages and contains 30 figures, 1 table, 1 appendice and 41 references.

The purpose of this work is to development of image reconstruction algorithms based on neural networks. An analysis of the image reconstruction problem was carried out, and the main challenges were identified. Methods of image reconstruction were investigated, and the feasibility of using neural network technologies for restoring images with artifacts was justified. Existing artificial neural network architectures were characterized, and a generative neural network model was highlighted as a promising approach for image reconstruction. The features of generative neural network functioning and their training algorithm were described. An algorithm for data preparation and training the generative neural network was developed. The architecture of the generative network for image reconstruction was designed. Software for image reconstruction with user interface was implemented. The adequacy of developed architecture for image reconstruction tasks was evaluated.

Practical significance: software for reconstructing images with artifacts.

Keywords: CONVOLUTIONAL NEURAL NETWORKS, MACHINE LEARNING, GAN NETWORKS, IMAGE CLASSIFICATION, IMAGE RECONSTRUCTION

ЗМІСТ

Вступ.....	7
1 Методи опрацювання зображень.....	11
1.1 Особливості задачі реконструкції зображень	11
1.2 Аналіз методів реконструкції зображень	14
1.3 Характеристика засобів реконструкції зображень	18
1.4 Постановка задачі кваліфікаційної роботи.....	22
1.5 Висновки до розділу.....	23
2 Алгоритм навчання нейромереж для реконструкції зображень	24
2.1 Штучні нейромережі як універсальні засоби обробки інформації	24
2.2 Модель нейромережі для реконструкції зображень.....	30
2.3 Алгоритм формування вибірки даних для навчання нейромережі	37
2.4 Модель архітектури генеративної нейромережі для реконструкції зображень.....	41
2.5 Висновки до розділу	44
3 Програмна система реконструкції зображень	46
3.1 Структура програмної системи	46
3.2 Архітектура GAN-мережі для реконструкції зображень.....	49
3.3 Експериментальні дослідження GAN-моделі при реконструкції зображень.....	56
3.4 Висновки до розділу	61
Висновки	62
Список використаних джерел	64
Додаток А Світлокопії публікацій.....	68

ВСТУП

Реконструкція зображень полягає у відновленні повноцінного та якісного зображення на основі неповних, спотворених або зашумлених даних. Це необхідно у тих ситуаціях, коли первинна інформація про зображення недоступна у цілісному вигляді через технічні обмеження обладнання, умови збирання даних або випадкові втрати. Реконструкція розглядається як інверсна задача: необхідно знайти таке зображення, яке найкраще пояснює отримані вимірювання, враховуючи модель формування сигналу та можливі спотворення.

Така задача поширена у медичній діагностиці для відновлення томографічних зображень, де дані збираються у вигляді множини проєкцій або сигналів, в супутниковій зйомці допомагає компенсувати втрати даних, зумовлені атмосферними перешкодами або обмеженнями сенсорів, у цифровій фотографії дає змогу підвищити роздільну здатність, усунути шум або відновити зображення після часткового пошкодження.

Складність задачі реконструкції у тому, що незначні похибки вхідних даних можуть призводити до суттєвих відхилень результативних. Тому для її розв'язання застосовуються різні підходи – від класичних математичних методів, які формалізують задачу через оптимізацію та регуляризацію, до сучасних нейромережових моделей, здатних навчатися на великих вибірках зображень і відтворювати складні просторові залежності.

Традиційні математичні методи реконструкції зображень ґрунтуються на строгих аналітичних моделях, властивостях сигналів та чисельних алгоритмах, які дозволяють відновлювати зображення на основі частково спотворених або неповних даних. Проте досить часто вони використовують лінійні математичні моделі зображень та шумів, через що не здатні адекватно описувати складні текстури, нерегулярні структури або високорівневі патерни. Методи регуляризації потребують точного підбору параметрів, що здійснює суттєвий

вплив на кінцевий результат. В той же час, класичні алгоритми чутливі до зашумлених даних й часто вимагають значних обчислювальних ресурсів.

За цих умов, досить популярними є нейромережеві моделі, які здатні автоматично навчатися на закономірностях у зображеннях і відновлювати їх із вищою точністю та стійкістю до шумів. Такі архітектури дають змогу відновлювати, покращувати або доповнювати зображення навіть у випадках, коли вихідні дані є частково втраченими, спотвореними або мають низьку роздільну здатність. На відміну від класичних методів, нейромережеві не вимагають вилучення ознак із зображень вручну й навчаються на великих обсягах неоднорідних даних. В той же час, варто зазначити, що нейромережеві моделі можуть стикатися із проблемою перенавчання, що впливає на точність розпізнавання.

Використання згорткових та глибоких нейронних мереж забезпечує автоматичне виявлення складних структур, текстур і просторових залежностей, що робить процес реконструкції суттєво точнішим. Завдяки здатності до навчання на великих вибірках даних такі моделі можуть відтворювати складні деталі та адаптуватися до різних типів даних – медичних знімків, супутникових фотографій, технічної візуалізації чи побутових фотографій.

Розвиток генеративних архітектур, зокрема автокодерів, GAN та дифузійних моделей уможливив суттєво підвищити якість реконструкції, забезпечуючи зменшення шумів, підвищення чіткості, відновлення пропущених фрагментів і навіть створення нових елементів, які органічно узгоджуються з контекстом. Це відкриває широкі можливості для медицини, криміналістики, промислової діагностики, а також систем відеоспостереження та мультимедійних технологій.

Постійний розвиток апаратного забезпечення та оптимізаційних алгоритмів сприяє зростанню точності нейромережевих моделей, зменшенню часу реконструкції й підвищенню ефективності їхнього застосування у реальних системах. Таким чином, розвиток алгоритмів реконструкції зображень на основі штучних нейронних мереж формують новий стандарт якості обробки

візуальних даних і створюють підґрунтя для подальших досліджень у сфері інтелектуальної обробки зображень. Звідси випливає мета кваліфікаційної роботи, а саме, розробка та удосконалення алгоритмів реконструкції зображень на основі згорткових та глибоких нейронних мереж. Досягнення зазначеної мети можливе завдяки виконанню таких задач:

- дослідити особливості задачі реконструкції зображень;
- проаналізувати методи та програмні засоби реконструкції зображень;
- проаналізувати існуючі архітектури штучних нейромереж;
- дослідити особливості функціонування згорткових та глибоких нейромереж;
- охарактеризувати алгоритм навчання генеративних моделей нейромереж;
- розробити алгоритм підготовки тренувальних даних;
- розробити архітектуру генеративної нейромережі для реконструкції зображень;
- провести експериментальні дослідження розробленої архітектури для задачі реконструкції зображень з артефактами.

Об’єкт дослідження – процес реконструкції зображень.

Предмет дослідження – алгоритми навчання штучних нейронних мереж для реконструкції зображень.

Методи дослідження: методи аналізу при дослідженні існуючих архітектур штучних нейромереж, синтезу при розробці архітектури нейронних мереж, методи фільтрації при реконструкції зображення, алгоритми навчання нейронних мереж, комп’ютерного моделювання при дослідженні розробленої архітектури нейронної мережі, машинного навчання для класифікації зображень.

Наукова новизна – генеративно-змагальна архітектура нейронної мережі для реконструкції зображень.

Практичне значення дослідження полягає у розробці програмної системи на основі штучних нейронних мереж для реконструкції зображень.

Кваліфікаційну роботу виконано, зважаючи на вимоги, які викладено у [33, 34].

Публікація та апробація результатів. Вагомі результати, отримані в процесі дослідження опубліковано на XVIII Міжнародній науково-практичній конференції «Інформаційні технології і автоматизація – 2025» та III Всеукраїнській науково-практичній конференції студентів, аспірантів та молодих вчених «Інтелектуальні комп'ютерні системи та мережі» [30, 35]. З копіями публікацій можна ознайомитися у додатку А.

У першому розділі проаналізовано особливості задачі реконструкції зображень. Досліджено найбільш поширені архітектури штучних нейронних мереж для розпізнавання зображень. Охарактеризовано алгоритми навчання штучних нейромереж. Здійснено постановку задачі кваліфікаційної роботи.

У другому розділі проаналізовано архітектури загорткових та глибоких нейронних мереж. Описано процедуру підготовки тренувальних даних. Розроблено алгоритм навчання нейромереж для задач реконструкції зображень.

У третьому розділі розроблено архітектуру програмної системи для реконструкції зображень. Обґрунтовано вибір середовища для розробки програмної системи. Розв'язано задачу реконструкції зображень з артефактами й результатами експериментів підтверджено перспективність застосування генеративно-змагальних нейромереж для задач такого типу.

У додатках наведено копії публікацій результатів дослідження.

РОЗДІЛ 1 МЕТОДИ ОПРАЦЮВАННЯ ЗОБРАЖЕНЬ

1.1 Особливості задачі реконструкції зображень

Процес обробки зображень охоплює виконання певних задач, зокрема сегментацію, видалення шумів і відокремлення об'єктів від фону, визначення контурів об'єктів, їх реконструкцію та подальше розпізнавання. Задача реконструкції зображень є однією із найскладніших задач комп'ютерного зору та цифрової обробки сигналів, оскільки спрямована на відновлення повного, якісного та інформативного зображення з даних, що містять спотворення, пропуски або шум.

У найзагальнішому вигляді вона виникає тоді, коли процес формування зображення порушується через технічні обмеження сенсорів, зовнішні фактори або передавання інформації, а тому вихідний візуальний сигнал втрачає частину просторової або інтенсивнісної інформації. Реконструкція полягає у знаходженні такого зображення, яке найбільш правдоподібно відповідає доступним даним, дотримуючись при цьому фізичних, математичних або статистичних моделей процесу формування зображення.

Процес реконструкції пов'язаний із необхідністю розв'язання інверсної задачі, де відомими є результати вимірювань або проєкцій, а невідомим – бажане зображення [40]:

$$y = Ax + n,$$

де x – невідоме істинне зображення, яке необхідно відновити,

A – оператор прямої задачі, що моделює перетворення (проєкцію, розмиття, згортку, субдискретизацію чи комбіновані ефекти),

y – спостережувані дані,

n – шум або випадкові перешкоди.

Задача реконструкції полягає у знаходженні оцінки \hat{x} , яка є найбільш узгодженою з даними y і властивостями системи. Тому реконструкцію формулюють як задачу оптимізації, де розв'язок визначають через мінімізацію функції невідповідності з регуляризацією.

У прямому розуміння таку задачу слід вважати некоректною, оскільки невеликі зміни у вхідних даних можуть призводити до суттєвих порушень у кінцевому результаті, тому оператор A часто є виродженим або погано обумовленим.

Зважаючи на це, реконструкція вимагає застосування спеціальних стабілізуючих методів, зокрема, методів регуляризації, що накладають певні обмеження на шукане зображення – гладкість, однорідність чи збереження контурів. Такі обмеження забезпечують отримання фізично зрозумілих та стійких до шуму розв'язків. Таким чином, реконструкцію можна сформулювати, як оптимізаційну задачу, де розв'язок визначають через мінімізацію функціоналу невідповідності з регуляризацією [40]:

$$\hat{x} = \arg \min_x (\|Ax - y\|^2 + \lambda R(x)),$$

де $\|Ax - y\|^2$ – міра узгодженості між зображеннями та вимірними даними,

$R(x)$ – регуляризаційний член, який вводить додаткові припущення про структуру зображення – гладкість, розрідженість, збереження контурів,

λ – параметр, що визначає компроміс між точністю наближення та стійкістю розв'язку.

Якщо оператор A є згорткою, то задачу можна представити у частотній області, що дозволяє застосовувати методи, засновані на Фур'є- або вейвлет-перетвореннях.

Реконструкція зображень включає такі підзадачі, що спрямовані на отримання максимально правдоподібного зображення, яке відтворює структури, деталі та текстури реального об'єкта:

- відновлення після зашумлення з метою вилучення випадкових перешкод без втрати інформативних деталей;
- суперрезолюція, яка передбачає отримання зображення з вищою роздільною здатністю;
- інпейнтинг – заповнення відсутніх фрагментів зображення унаслідок пошкоджень, маскування або неповних вимірювань. Параметр A є маскою, що позначає доступні пікселі, а реконструкція спрямована на відновлення відсутніх значень із урахуванням локальних або глобальних залежностей;
- томографічна реконструкція, що базується на аналізі множини проєкцій, з яких відновлюється об'ємна структура об'єкта. Оператор A відповідає інтегральним перетворенням і задача реконструкції зводиться до знаходження його інверсії.

Важливою складовою задачі реконструкції є формування математичної моделі процесу, що пов'язує невідоме зображення з доступними даними. Ця модель може бути лінійною або нелінійною, стохастичною чи детермінованою, залежно від природи вимірювальної системи. У практичних системах часто використовуються апроксимації, зокрема згорткові моделі, які описують неточності. Від того, наскільки точно модель відображає реальний процес, залежить якість реконструкції.

Із вищезазначеного можна сформувати схему розв'язування задачі реконструкції зображень, яку зображено на рисунку 1.1.

Таким чином, математичний опис задачі реконструкції поєднує операторний аналіз, статистичні моделі та методи оптимізації, формуючи універсальну основу для побудови як традиційних алгоритмів, так і сучасних нейромережевих підходів. Однак незалежно від методології, задача реконструкції залишається однією з найскладніших у комп'ютерному зорі, оскільки вимагає одночасного урахування математичної коректності, стійкості

до шуму, збереження деталей і відповідності реальним фізичним характеристикам об'єкта.

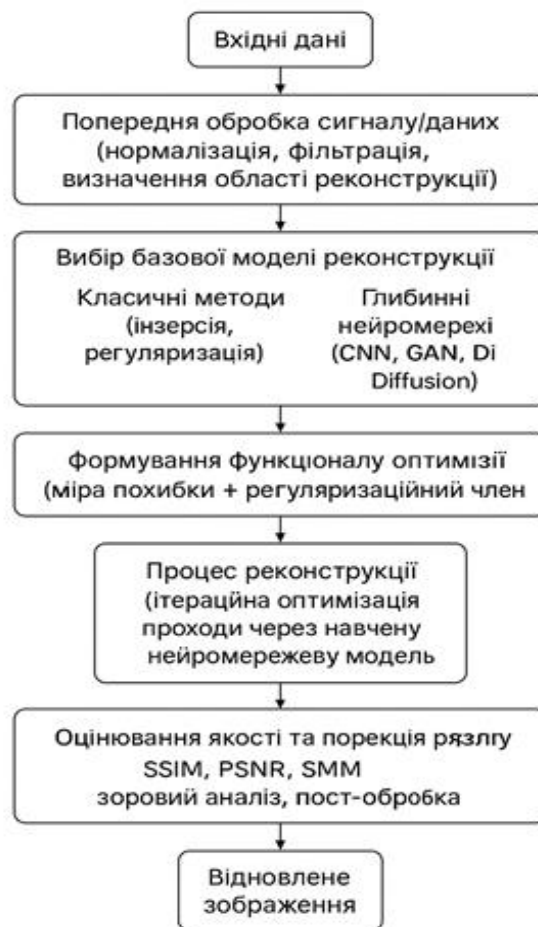


Рисунок 1.1 – Структурна схема розв’язування задачі реконструкції зображень

У наступному підрозділі детально розглянемо сучасні методи реконструкції зображень, що уможливають отримати якісний результат на основі неповних вихідних характеристик.

1.2 Аналіз алгоритмів реконструкції зображень

Алгоритми реконструкції зображень охоплюють методи, які відновлюють зображення з неповних, зашумлених або проміжних даних, оцінюючи їх

точність, стійкість, обчислювальну ефективність та здатність відтворювати дрібні структури. Класичні математичні алгоритми базуються на методах обернених задач, лінійній алгебрі та варіаційному аналізі. Вони включають фільтраційні техніки, регуляризаційні моделі та оптимізаційні процедури, які націлені на знаходження балансу між точністю та стійкістю результату. Перевагою таких алгоритмів є формальна строгість та передбачуваність результатів, однак вони обмежені у випадках складних, нелінійних або сильно зашумлених даних, де звичайні припущення про гладкість чи простоту структури сигналу не працюють.

До традиційних математичних методів реконструкції зображень відносять методи аналізу часових рядів, статистичні, оптимізаційні методи. Методи Фур'є-реконструкції ґрунтуються на тому, що зображення можна відновити за його частотним спектром. Застосування фільтраційних функцій, наприклад, Ram-Lak або Shepp–Logan дає змогу компенсувати спотворення, але чутливість до шуму залишається високою [28].

Варіаційні методи формулюються як задача мінімізації функціоналу, який включає терм узгодженості з даними та регуляризатор, що задає вимоги до гладкості або структурності результату. Найпоширенішими регуляризаторами є Tikhonov, Total Variation та TV-L1. Метод Тихонова забезпечує просту гладку реконструкцію, але надмірно згладжує деталі. Метод Total Variation краще зберігає контури, проте може породжувати «східцеві» артефакти (staircasing effect) [40].

Методи на основі ітеративних оптимізацій включають алгоритми ART Algebraic Reconstruction Technique (ART), SIRT Simultaneous Iterative Reconstruction Technique (SIRT) та Multiplicative ART (MART). Ці підходи покроково покращують оцінку зображення, звіряючи її з доступними проєкціями або вимірюваннями. ART забезпечує швидку збіжність, але часто є нестійким, SIRT характеризується повільною збіжністю.

Статистичні методи використовують максимізацію правдоподібності або апостеріорної ймовірності. Метод максимального правдоподібності та його

модифікації враховують статистичні властивості шуму, що дозволяє отримувати якісніші реконструкції, однак вимагають обчислювальних затрат та часто призводять до посилення шуму без регуляризації.

Алгоритми реконструкції, що спираються на штучні нейронні мережі, формують інший клас методів. Вони здатні вивчати складні закономірності з великих обсягів даних, моделювати нелінійні залежності та реконструювати зображення навіть у дуже спотворених умовах. Аналіз таких алгоритмів охоплює оцінку архітектури мережі, здатність до узагальнення, стабільність роботи, чутливість до вибірки навчальних даних та ризику появи артефактів. Порівняно з традиційними методами вони забезпечують вищу якість реконструкції, проте можуть вимагати великих обчислювальних ресурсів.

Конволюційні (згорткові) нейронні мережі (CNN) для суперрезолуції застосовуються для підвищення чіткості та якості зображення, виходячи з низькороздільних даних. Мережі типу SRCNN, FSRCNN або EDSR навчаються знаходити складні просторові залежності, які не враховуються класичними методами. CNN добре відновлюють текстури, проте схильні до появи штучних «галюцинованих» деталей [40].

Автокодові архітектури (Autoencoder, Denoising Autoencoder) ефективно використовуються для задач шумоприглушення та часткової реконструкції. Автокодер вчиться відтворювати чисте зображення із зашумленого, що дозволяє використовувати його як регуляризатор у задачах з неповними даними. Однак якість результату залежить від репрезентативності навчальної вибірки.

Глибокі генеративні моделі на кшталт Variational Autoencoder (VAE) та Generative Adversarial Networks (GAN) дозволяють реконструювати зображення при сильній деградації. GAN-моделі типу pix2pix, CycleGAN або StyleGAN здатні відновлювати деталі, яких взагалі немає у вихідних даних, спираючись на статистику з навчальної вибірки [28]. Цей клас методів забезпечує природний вигляд зображень, але водночас може створювати артефакти та продукувати дані, які не мають фізичного обґрунтування.

Методи гібридної реконструкції поєднують математичні моделі та нейромережі. До них належать підходи типу Deep Unrolling, у яких алгоритм оптимізації перетворюється на багат шарову мережу з навчальними параметрами регуляризації [10, 12]. Такий підхід використовує основи традиційних методів і гнучкість нейромереж, забезпечуючи високу якість та стабільність відновлення. Недоліком таких методів є складність навчання і необхідність великих вибірок даних.

Для задачі реконструкції зображень часто виникає необхідність підвищення роздільної здатності. Основним підходом у цьому випадку є методи інтерполяції, що дозволяють обчислювати проміжні значення на основі наявних даних. Фактично, збільшення роздільної здатності полягає у визначенні кольорових значень нових пікселів, що з'являються при масштабуванні зображення. Інструменти інтерполяції застосовуються у більшості сучасних редакторів растрових зображень. Проте використання інтерполяції може призводити до виникнення артефактів, серед яких:

- хвильові спотворення в областях різких змін яскравості (так званий Ringing);
- ступінчастість діагональних кордонів (Aliasing);
- зниження чіткості зображення (Unsharpening).

Додатково, залежно від типу інтерполяції, можливі субпіксельні зрушення (Subpixel shift), які хоч і малопомітні візуально, проте можуть суттєво впливати на формальні метрики оцінки якості.

Інтерполяційні алгоритми умовно поділяють на дві категорії:

- неадаптивні, які обробляють усі пікселі однаково;
- адаптивні, що підлаштовуються під конкретну задачу інтерполяції.

Адаптивні алгоритми, зазвичай, дозволяють уникнути таких спотворень, як overshooting чи aliasing, і зберегти різкість контурів. Водночас вони можуть продукувати окремі пікселі або текстурі, невластиві вихідному зображенню.

До прикладів неадаптивних лінійних алгоритмів належать:

- метод найближчого сусіда, який визначає колір пікселів на основі значень сусідніх пікселів за певною метрикою відстані;
- білінійна інтерполяція, що є розширенням лінійної інтерполяції для двовимірних функцій;
- бікубічна інтерполяція, яка забезпечує гладку поверхню на межах суміжних квадратів у двовимірній сітці;
- сплайн-інтерполяція, де область визначення функції ділиться на відрізки, на кожному з яких функція апроксимується багаточленом.

З огляду на вищезазначене, ефективне вирішення задачі реконструкції зображень потребує комплексного підходу, що враховує характеристики вихідних даних, вимоги до точності та швидкодії, стійкості до шуму і спотворень, а також інтерпретованості результатів. Такий підхід дозволяє досягати високої якості відновлення зображень, що досить важливо у таких сферах, як медицина, відеографія, промислова візуалізація.

У наступному підрозділі детально проаналізуємо існуючі програмні системи для реконструкції зображень із зазначенням їхніх переваг та недоліків.

1.3 Характеристика засобів реконструкції зображень

У традиційних програмних комплексах реконструкції зображень використовують алгоритми фільтрації, регуляризації, інтерполяції та оптимізації, вбудовані в системи обробки сигналів. Такі інструменти забезпечують відновлення зображень у томографії, мікроскопії, радіолокаційних системах та цифровій фотографії. Їх робота часто базується на чітко визначених математичних моделях та аналітичних розв'язках, що робить результати передбачуваними й інтерпретованими, але менш ефективними при складних або надто деградованих даних.

Серед найпоширеніших інструментів реконструкції зображень у науковій та інженерній практиці важливе місце займає MATLAB із пакетом Image Processing Toolbox [17]. Цей комплекс забезпечує широкий вибір алгоритмів регуляризації, фільтрації, інтерполяції та оптимізації. Image Processing Toolbox містить засоби для суперрезолюції, інпейнтингу, роботи з частотними перетвореннями та реалізації методів томографічної реконструкції, що уможливорює моделювання та тестування реконструкційних алгоритмів різної складності. Основною перевагою пакету MATLAB є високий рівень точності реалізації математичних методів та зручність роботи з великими масивами даних. Однак недоліком є висока вартість ліцензії та залежність від закритої екосистеми, що обмежує можливості інтеграції з власними програмними системами.

Бібліотека OpenCV широко використовується в задачах комп'ютерного зору. Її особливістю полягає у відкритому вихідному коді, великій спільноті та високій швидкодії обробки зображень у реальному часі [36]. Попри це, OpenCV зосереджується переважно на методах класичної обробки, тому не завжди забезпечує максимальну якість реконструкції у високоточних задачах, де потрібні сучасні алгоритми машинного навчання.

Особливе значення в медичній сфері займає середовище ІТК (Insight Segmentation and Registration Toolkit), що розроблено спеціально для задач тривимірної реконструкції, сегментації та реєстрації зображень. Універсальність, наукова достовірність та підтримка стандартів підтверджують його придатність у томографії [13]. Недоліком такої системи є складність засвоєння та необхідність глибоких знань програмування.

Для роботи з великими вибірками даних часто використовують 3D Slicer, який поєднує інструменти реконструкції, візуалізації та аналізу. Його перевагою є доступність, наявність засобів для роботи з 3D-моделями та підтримка модулів на Python. Разом із тим, продуктивність 3D Slicer може бути недостатньою для задач, що потребують обробки зображень в режимі реального часу [1].

У сфері сучасних нейромережевих методів реконструкції зображень досить поширеною є бібліотека TensorFlow, яка дозволяє реалізовувати складні моделі суперрезолюції, шумозаглушення чи томографічної реконструкції. Перевагою TensorFlow є масштабованість і можливість використання апаратного прискорення, особливо на GPU [22, 23]. Недоліком бібліотеки виступає складність розробки моделей та висока вимога до обчислювальних ресурсів.

Бібліотеку PyTorch вважають гнучкою і зручною, що забезпечує простоту експериментів і широкий вибір готових моделей для реконструкції зображень [25]. PyTorch забезпечує середовище для побудови та навчання глибоких нейромереж, що дозволяє реалізовувати як класичні реконструкційні методи, так і найновіші генеративні моделі.

PyTorch підтримує автоматичне диференціювання, що дає змогу легко формулювати та оптимізувати функціонали реконструкції, налаштовувати архітектури згорткових мереж, автоенкодерів, GAN-моделей або дифузійних підходів. Це уможливорює будувати адаптивні методи реконструкції, які відновлюють структуру зображень навіть у складних умовах шуму, неповних вимірювань або низької роздільності.

У поєднанні з бібліотеками TorchVision та PyTorch Lightning середовище PyTorch дозволяє формувати повні програмні прототипи реконструкційних алгоритмів, виконувати навчання на великих наборах даних та порівнювати різні архітектури, що виділяє його як один із найпотужніших інструментів у галузі відновлення зображень. Однак така гнучкість може ускладнювати розгортання систем, які вимагають стабільності і контролю версій.

Спеціалізованим інструментом у сфері глибоких методів є MONAI, розроблений для задач реконструкції томографічних зображень. MONAI надає попередньо налаштовані архітектури, стандартизовані пайплайни та інструменти оптимізації для медичних форматів даних [40]. Недоліком такої системи є складність використання для не медичних задач та вимога потужної апаратної бази.

Платформа NVIDIA Clara поєднує апаратні та програмні засоби для високошвидкісної реконструкції зображень, включаючи алгоритми на основі штучного інтелекту [18]. Вона забезпечує надзвичайно високу продуктивність, проте її використання обмежене потребою у специфічному обладнанні та ліцензіях.

До інноваційних систем також належить Deep Image Prior – метод, який використовує структуру нейромережі як регуляризатор навіть без традиційного навчання на великих вибірках [8-10]. Система здатна відновлювати зображення, спираючись лише на внутрішні властивості мережі, що відкриває можливість застосування у випадках, коли відсутні великі вибірки навчальних даних.

HuggingFace Diffusers – це спеціалізована бібліотека для розробки, навчання та застосування дифузійних моделей (diffusion models), які сьогодні є одним із найпотужніших інструментів реконструкції, інпейнтингу та суперрезолуції зображень. Це не просто набір моделей, а повноцінна екосистема для роботи з сучасними генеративними алгоритмами. Недоліками такої бібліотеки є суттєве споживання пам'яті, висока обчислювальна складність, відсутність універсальної стабільності, високі вимоги до знань дослідника.

Помітне місце займають дифузійні моделі, інтегровані у такі програмні системи, як Stable Diffusion. Вони демонструють здатність відновлювати зображення зі великим рівнем шуму або спотворення, працюючи через ітераційне уточнення структури з використанням стохастичних процесів.

Таким чином, із проведеного аналізу випливає, що в точності реконструкції зображень виграють системи, які ґрунтуються на нейромережевих підходах, оскільки часто вихідні зображення є нечіткими, зашумленими й досить спотвореними, що свідчить про неоднорідність вибірки даних.

Зважаючи на це, перейдемо до постановки задачі кваліфікаційної роботи й у наступному розділі розглянемо детально особливості функціонування штучних нейромереж, типові архітектури та алгоритми їх навчання. Це

уможливить обґрунтований вибір архітектури мережі для розробки моделі реконструкції зображень.

1.4 Постановка задачі кваліфікаційної роботи

Реконструкція зображень є однією з фундаментальних проблем сучасної комп'ютерної графіки, медичної візуалізації та цифрової обробки сигналів. Вона виникає у випадках, коли зображення спотворене шумом, має знижену роздільну здатність або містить пропуски, спричинені недосконалістю сенсорів, обмеженнями каналів передачі даних чи особливостями вимірювальних систем. Традиційні методи реконструкції, засновані на математичній інтерполяції, частотному аналізі або стохастичних моделях, є непридатними у випадках, коли зображення має досить зашумлене, має складну структуру або неоднорідні артефакти. При цьому класичні методи не здатні відновити дрібні деталі та коректно реконструювати текстури.

Зазначені обмеження враховано у методах, що ґрунтуються на нейромережевому підході й здатні відтворювати складні структури та локальні закономірності, притаманні реальним об'єктам. Такі алгоритми здатні до відновлення інформації з неповних або сильно зашумлених даних, враховуючи статистичні залежності у великих вибірках зображень. Нейромережеві архітектури можуть самостійно навчатися, є адаптивними, що уможлиблює отримати результат із мінімальною похибкою.

Звідси можна сформулювати мету роботи, а саме – розробка архітектури штучної нейронної мережі для реконструкції зображень. Реалізація цієї мети вимагає виконання таких задач:

- проаналізувати існуючі архітектури штучних нейромереж;
- охарактеризувати алгоритми навчання штучних нейронних мереж;

- дослідити особливості функціонування згорткових та глибоких нейромереж;
- описати процедуру підготовки тренувальних даних;
- розробити архітектуру генеративно-змагальної нейромережі для реконструкції зображень;
- провести експериментальні дослідження розробленої архітектури мережі для реконструкції зображень з артефактами.

Виходячи з цього, задача розробки алгоритмів реконструкції полягає у розробці нейромережевої моделі, здатної приймати спотворене зображення і відтворювати його відновлену версію з максимальною якістю та точністю.

1.5 Висновки до розділу

Проаналізовано задачу реконструкції зображень й зазначено, що формально – це оптимізаційна задача на основі мінімізації функції невідповідності із регуляризацією. Досліджено існуючі методи реконструкції зображень, що забезпечують передбачуваність та теоретичну обґрунтованість, але суттєво обмежені за умов дуже спотворених та зачумлених даних. Програмні засоби на основі нейромереж показують вищу якість реконструкції, здатність до відновлення тонких структур і адаптацію до різних типів даних, проте інколи потребують великих вибірок даних та значних обчислювальних ресурсів. Досягнення максимальної точності та стійкості реконструкції зображень можливе за умов застосування комплексної моделі.

РОЗДІЛ 2 АЛГОРИТМ НАВЧАННЯ НЕЙРОМЕРЕЖ ДЛЯ РЕКОНСТРУКЦІЇ ЗОБРАЖЕНЬ

2.1 Штучні нейромережі як універсальні засоби обробки інформації

Нейронні мережі представляють математичні моделі обробки інформації, які імітують структуру та функції нервової системи живих організмів. Вони складаються з нейронів, взаємопов'язаних між собою та здатних виконувати обчислення над вхідними даними. Нейрони отримують набір вхідних сигналів, кожен із яких коригується на відповідний ваговий коефіцієнт, який визначає його значущість для формування вихідного сигналу. Сумарне значення зважених входів подається на функцію активації, яка визначає кінцевий вихід нейрона.

Функції активації різних типів застосовуються для обмеження значень вихідного сигналу нейрона у визначеному діапазоні та зменшення впливу аномальних значень у вхідних даних.

Нейрони об'єднуються у шари – вхідний, прихований та вихідний. Приховані шари, розміщені між вхідним та вихідним, виконують обробку даних, що й є особливістю нейронних мереж. Кількість прихованих шарів та нейронів у них визначається специфікою розв'язуваної задачі. Узагальнену архітектуру нейронної мережі наведено на рисунку 2.1.

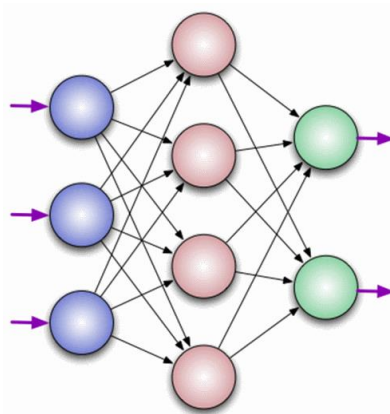


Рисунок 2.1 – Узагальнена архітектура нейромережі

Навчання нейромережі відбувається шляхом оптимізації вагових коефіцієнтів на основі наявних вхідних даних та очікуваних вихідних результатів. Вихідні значення мережі порівнюються з еталонними, і на основі обчисленої помилки коригуються вагові коефіцієнти із застосуванням здебільшого алгоритму зворотного поширення помилки, що дозволяє поступово мінімізувати різницю між фактичними та очікуваними результатами (див. рисунок 2.2).

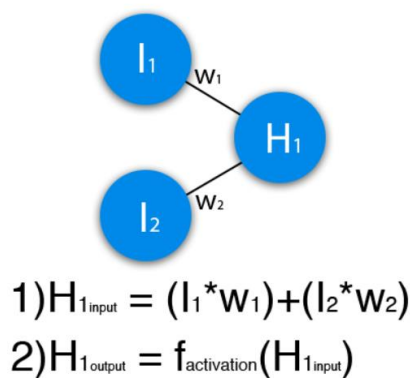


Рисунок 2.2 – Розрахунок виходу нейрона

Існує чимало класів нейромереж, серед яких багатошарові перцептрони, згорткові, рекурентні та глибокі нейромережі, кожен з яких застосовується у різних сферах, від розпізнавання зображень до опрацювання природної мови. Багатошаровий перцептрон є класичним типом нейронної мережі, його архітектуру наведено на рисунку 2.3.

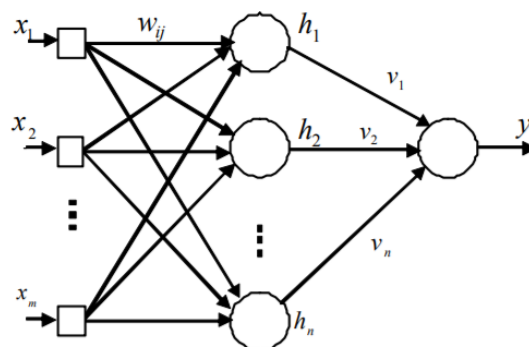


Рисунок 2.3– Архітектура перцептрона

Рекурентні неймережі (Recurrent Neural Networks, RNN) застосовуються для аналізу послідовних даних, оскільки здатні зберігати інформацію про попередні значення вхідних сигналів (див. рисунок 2.4). Водночас базові RNN мають певні обмеження, такі як проблема зниклого градієнта та складнощі у моделюванні довгострокових залежностей. Для подолання цих проблем розроблено вдосконалені архітектури, які забезпечують ефективнішу обробку послідовностей даних і дозволяють вирішувати більш складні задачі [37].

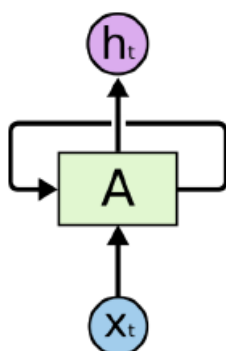


Рисунок 2.4 – Рекурентна мережа

«Глибокі неймережі (Deep Neural Networks, DNN) – це архітектури з декількома прихованими шарами, що мають велику обчислювальну потужність і здатність до вилучення ознак об’єкта, що дозволяє їм ефективно вирішувати складні задачі, зокрема, розпізнавання мови чи обробка зображень» [38].

Згорткові неймережі (Convolutional Neural Networks, CNN) застосовуються для аналізу зображень і складаються з шарів згортки, що виконують фільтрацію вхідних даних, та шарів пулінга, які зменшують розмірність даних. Така архітектура дозволяє CNN автоматично виділяти ознаки зображення та здійснювати його класифікацію [29].

Моделі згорткових неймереж, що використовуються для задач підвищення роздільної здатності зображень, здобули широке поширення завдяки здатності навчатися на інформативних ознаках та відновлювати зображення на основі обмеженої кількості вихідних даних.

«Модель SRCNN (Super-Resolution Convolutional Neural Network) є однією з перших, яка застосувала підхід глибокого навчання для задач підвищення роздільної здатності зображень. Вона працює з попередньо збільшеними на основі інтерполяції та зменшеними за роздільністю зображеннями, навчаючись відновлювати деталі та покращувати якість вихідного зображення» [29].

Подальші модифікації, такі як «VDSR (Very Deep Convolutional Network), FSRCNN (Fast Super-Resolution Convolutional Neural Network) та ESPCN (Efficient SubPixel Convolutional Neural Network), удосконалили цей підхід, забезпечуючи можливість збільшення роздільності зображень у режимі реального часу» [29].

Застосування згорткових нейромереж у задачах Super-Resolution має суттєву перевагу над класичними методами, оскільки мережі здатні навчатися на інформативних ознаках та враховувати взаємозв'язки між пікселями зображення, що дозволяє досягати суттєво кращих результатів. Одними з перших моделей, які продемонстрували ефективність у задачі підвищення роздільної здатності, є SRCNN та її модифікація FSRCNN (див. рисунок 2.5).

Згорткові нейронні мережі працюють шляхом формування карт ознак із вхідного зображення та подальшого підвищення їх дискретизації для отримання зображення з більшою роздільною здатністю. Такі мережі суттєво зменшують розмірність вхідних даних, водночас зберігаючи ключові ознаки, що дозволяє досягти максимально точного результату.

Для цього вхідне зображення проходить через згорткові шари, шари об'єднання та шар класифікації (див. рисунок 2.6). На відміну від традиційних нейромереж, нейрони в CNN мають однакові ваги, і, як правило, не всі нейрони між шарами зв'язані, що забезпечує ефективне виділення локальних ознак.

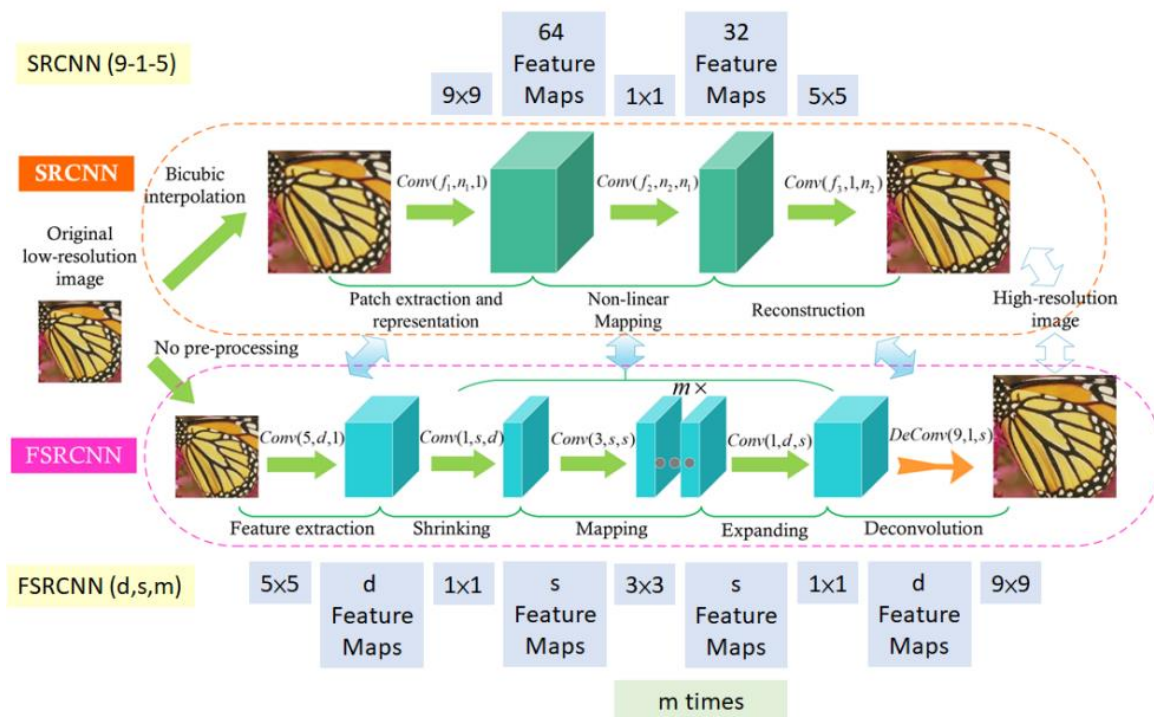


Рисунок 2.5 – Архітектури SRCNN та FSRCNN

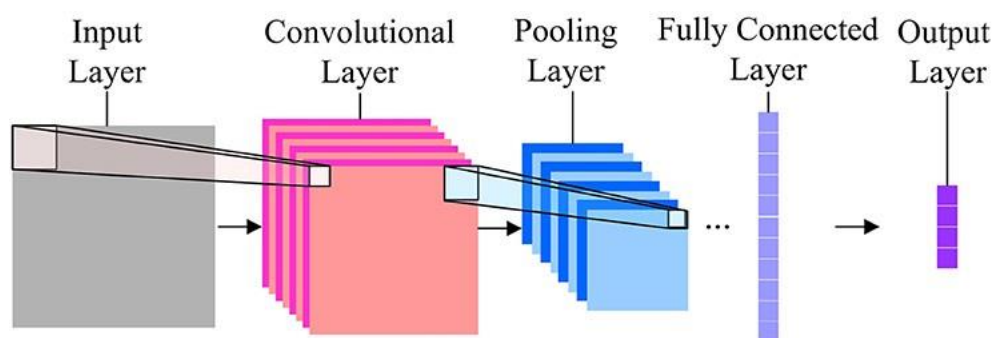


Рисунок 2.6 – Архітектура CNN

Для зменшення розмірності вхідних даних до зображення застосовуються спеціальні фільтри, які називають ядрами. Зазвичай, їх розмір становить 3×3 або 5×5 , й при цьому вони дозволяють виділяти високорівневі ознаки або виконувати перетворення. Застосування таких операцій може призводити до збільшення, або зменшення розмірів об'єкта, залежно від використаного типу заповнення (padding). Ядро послідовно переміщується по всьому зображенню за певним шаблоном, як показано на рисунку 2.7.

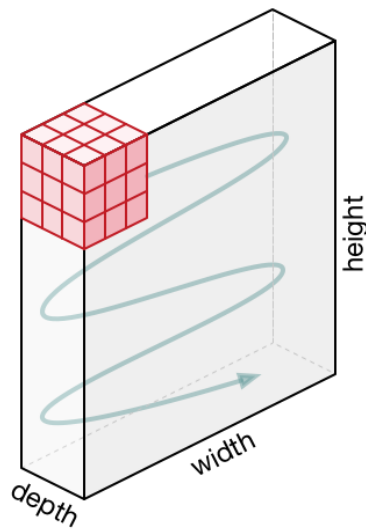


Рисунок 2.7 – Переміщення ядра по шаблону

Згорткові нейромережі також використовують набір фільтрів (див. рисунок 2.8) для виділення ознак із вхідного зображення, які слугують інформативними параметрами для класифікації. Початкові шари нейромережі відповідають за виявлення простих та дрібних ознак, таких як контури та краї об'єктів на зображенні.

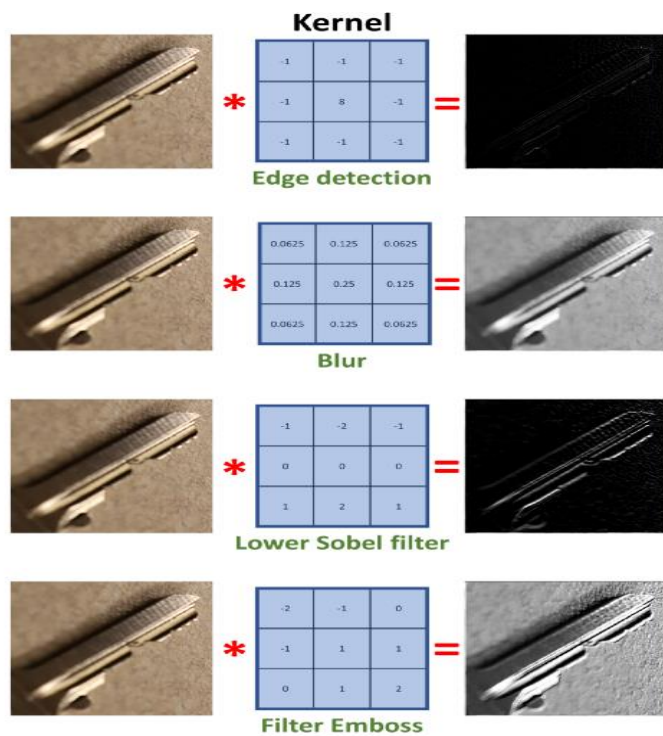


Рисунок 2.8 – Фільтрація зображення

Кожне ядро виявляє певну ознаку у вхідному зображенні, формуючи карту ознак, що будується нейронами з використанням однакового ядра. Під час навчання мережі значення в цій карті змінюються відповідно до функції мінімізації втрат. Одержані карти ознак проходять через активаційну функцію ReLU, де шари та карти об'єктів об'єднуються, а при зворотному поширенні помилки значення в матрицях фільтрів коригуються для оптимізації роботи мережі.

Шар пулу застосовується для зменшення розмірів вихідних даних зі згорткового шару, що дозволяє зменшити обсяг обчислень, необхідних для подальшої обробки, та виділити домінуючі ознаки, незалежні від позиції, при збереженні процесу навчання мережі. Цей шар обробляє вихідне зображення з попереднього шару, виконуючи операції максимального або середнього об'єднання (max pooling або average pooling), що дозволяє скоротити розмірність даних і зменшити ризик перенавчання шляхом акцентування на найбільш значущих ознаках.

У наступному розділі перейдемо до детального аналізу нейромереж, які найбільш ефективні у задачах реконструкції зображень.

2.2 Модель нейромережі для реконструкції зображень

Нейромережа для реконструкції зображень складається з кількох шарів, кожен із яких виконує певні обчислювальні операції та впливає на поведінку всієї моделі. Кожен шар характеризується параметрами, які визначають його функціонування та здатність відновлювати деталі зображень.

Розмір ядра (Kernel Size) визначає просторові розміри фільтра, що застосовується до вхідних даних, і впливає на форму та розмір шару виходу. Кількість фільтрів (Number of Filters) визначає кількість незалежних ядер, які здатні виявляти різні ознаки на зображенні, що підвищує здатність мережі

розпізнавати складні залежності. Крок зсуву (Stride) відстежує переміщення фільтра по зображенні, впливаючи на розмір вихідних даних та деталізацію результату. Заповнення (Padding) дозволяє додавати пікселі навколо вхідного зображення, щоб зберегти розмір виходу під час обробки.

Активаційна функція (Activation Function) надає моделі нелінійність, визначаючи, які сигнали передавати до наступного шару. Вибір функції, наприклад ReLU, sigmoid чи tanh, залежить від специфіки задачі та параметрів даних. Складність шару (Layer Complexity) обчислює кількість параметрів, що навчаються, і впливає на обчислювальні витрати та час тренування. Усі ці параметри визначають поведінку мережі та якість відновлення зображень.

При застосуванні генеративних нейромереж архітектуру розробляють таким чином, щоб під час тренування отримувати зображення із високою роздільною здатністю $F(Y)$ на основі вхідного зображення із низькою роздільною здатністю. Процес включає три основні етапи:

- виділення ділянок зображення та розробка прогнозу, де фрагменти низькороздільного зображення перетворюються у багатовимірні вектори, які утворюють карту ознак;
- нелінійне відображення, під час якого кожному вектору зіставляється спрогнозований вектор високої якості зображень;
- відновлення, що полягає в об'єднанні окремих ділянок у єдине зображення.

Реалізація цього підходу забезпечує ефективне відновлення деталей і підвищення якості реконструкції. На рисунку 2.8 показано схему підвищення якості зображення на основі CNN.

На вхід нейромережі поступає зображення низької якості Y . Перший шар згортки виконує виокремлення інформативної карти ознак із отриманого зображення. Другий шар виконує нелінійне перетворення отриманих ознак у зображення високої якості. Останній шар здійснює об'єднання прогнозних

результатів для окремих об'єктів, формуючи при цьому результуюче зображення високої якості $F(Y)$.

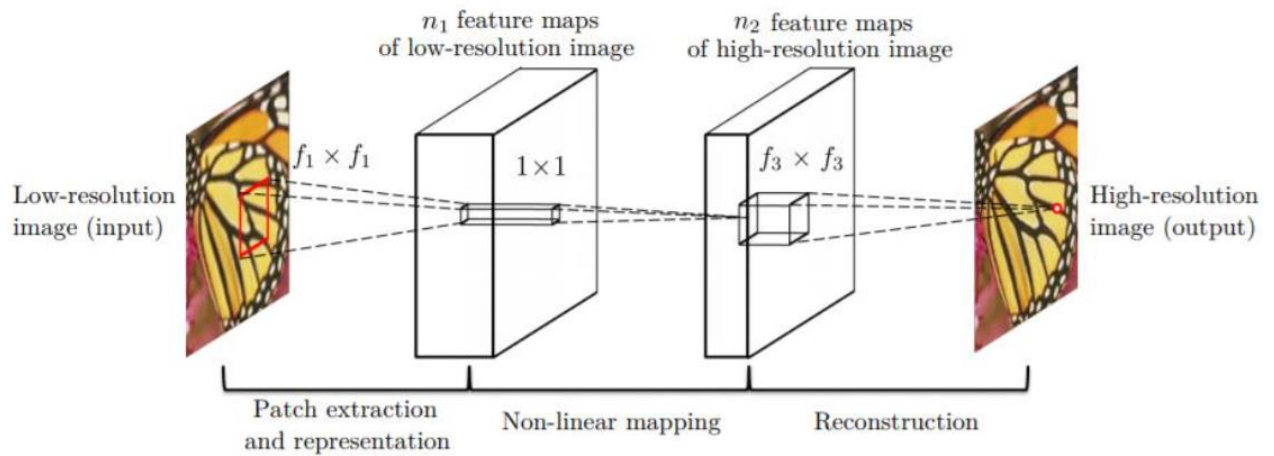


Рисунок 2.8 – Схема підвищення роздільної здатності

«Всі CNN володіють трьома вимірами – ширина, глибина та роздільна здатність» [26]. «Глибина поозначає кількість шарів, ширина – кількість каналів а роздільна здатність – кількість пікселів у зображенні, тобто його якість. Кожен із цих вимірів підлягає масштабуванню, і кожен при цьому певним чином підвищує точність CNN. Масштабування ширини мережі збільшує кількість каналів у зображенні (або нейронів у певному шарі). Така процедура дозволяє шарам виявляти деталізовані ознаки, однак при збільшенні ширини зростає складність задачі виявлення комплексних ознак» [26]. Приклад масштабування шарів нейромережі показано на рисунку 2.9.

Масштабування глибини відповідає за збільшення кількості шарів CNN-мережі, що уможлиблює виявляти складніші ознаки. В той же час існує проблема зникнення градієнтів, яка ускладнює процес навчання нейромереж. Хоч методи пакетної нормалізації та обхідні зв'язки суттєво зменшують цей ефект, емпіричні дослідження показали, що приріст точності з часом сповільнюється.

Масштабування роздільної здатності полягає у збільшенні кількості пікселів зображення, що дозволяє моделі виявляти дрібніші структури завдяки додатковим деталям.

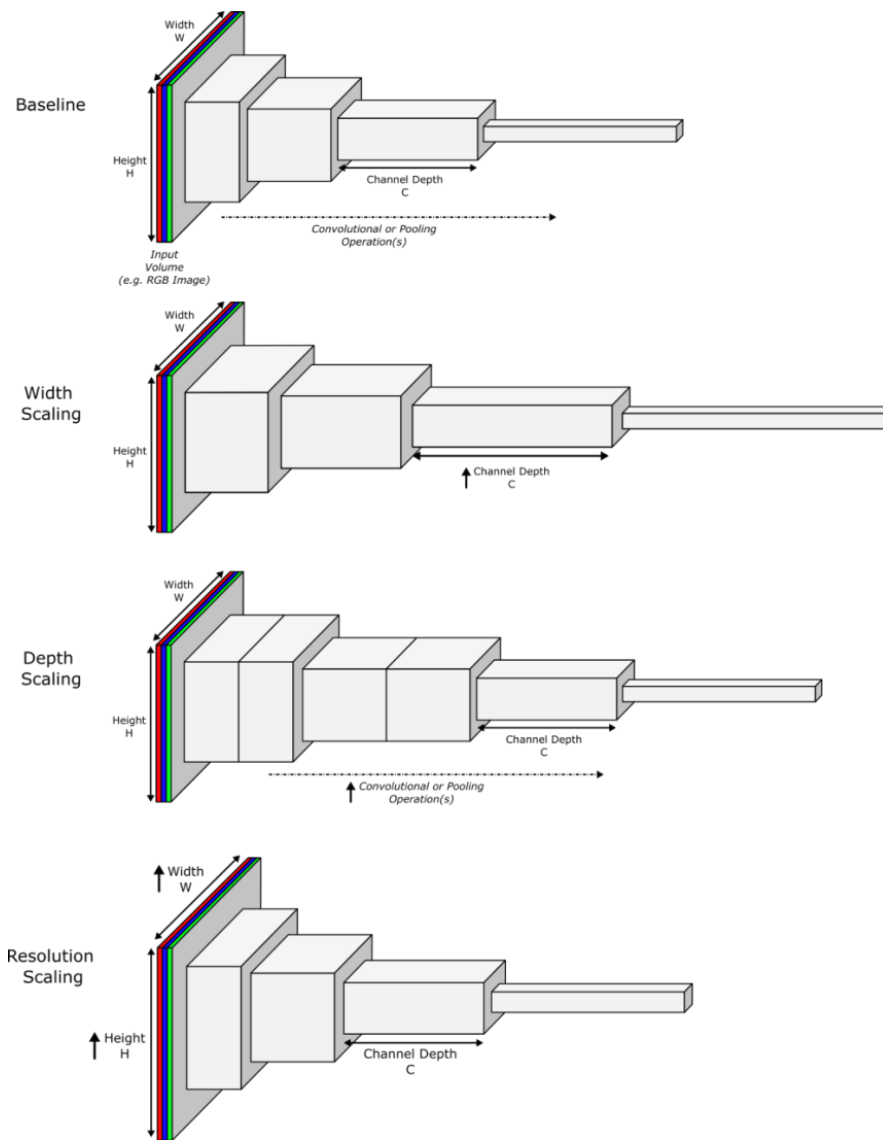


Рисунок 2.9 – Методи масштабування неймереж

Дослідження показують, що оптимальний приріст точності досягається при збалансованому масштабуванні всіх трьох аспектів мережі – ширини, глибини та роздільної здатності, при цьому обчислювальна складність зростає мінімально.

«Генеративно-змагальна неймережа (Generative Adversarial Network, GAN) – це архітектура глибокого навчання, яка складається з двох нейронних

мереж, що навчаються одночасно у режимі змагання» [27, 28, 38]. Її суть полягає у тому, що одна мережа створює дані, максимально подібні до реальних, а інша – намагається відрізнити штучно створені дані від справжніх. У результаті цього протистояння модель поступово вчиться генерувати реалістичні зображення. Узагальнену архітектуру генеративної мережі наведено на рисунку 2.10.



Рисунок 2.10 – Структура генеративної нейромережі

Генератор (Generator) – це мережа, яка вхід якої поступає випадковий шум й у процесі навчання намагається згенерувати дані (зображення), що виглядають максимально реалістично. Його мета обманути дискримінатор, змусити його вважати згенероване зображення справжнім. Генератор побудований на основі так званих діляційних або транспонованих згорткових шарів, які виконують протилежну дію до класичних згорток – не стискають, а розширюють простір зображення.

Дискримінатор (Discriminator) – це класифікатор, який розрізняє реальні дані та згенеровані. Його мета правильно відрізнити справжні приклади від підроблених. Дискримінатор працює як CNN, що виконує послідовність згорток й аналізує патерни текстур, взаємне розташування об’єктів, локальні та глобальні структури.

Навчання GAN відбувається у вигляді двостороннього змагання – генератор покращує здатність створювати реалістичні дані, а дискримінатор

покращує здатність виявляти підробки. Обидві мережі постійно удосконалюються, доки генератор не навчиться створювати такі зображення, які дискримінатор уже не може відрізнити від справжніх. Таке змагальне навчання формує мінімакс-гру, де кожна мережа прагне мінімізувати свою помилку й максимізувати хибність протилежної.

Охарактеризуємо процедуру навчання GAN-мереж. «Нехай $D(x)$ – вихід дискримінатора, який є ймовірністю того, що x є реальним зображенням. $G(z)$ – вихід генератора (дані, синтезовані генератором), де h – шум, поданий на вхід генератору. Мета дискримінатора – максимізувати логарифм ймовірності реальних даних та логарифм зворотної ймовірності синтетичних даних» [11]:

$$\max_D \left\{ \log(D(x)) + \log(1 - D(G(z))) \right\}.$$

Генератор у цьому випадку прагне мінімізувати логарифм зворотної ймовірності, який оцінює дискримінатор для синтетичних даних. Його навчання спрямоване на те, щоб продуковані ним дані були максимально реалістичними та з високою ймовірністю не розпізнавалися дискримінатором як штучно згенеровані:

$$\min_D \left\{ \log(1 - D(G(z))) \right\}.$$

Таким чином, кінцева функція втрат визначається як мінімаксна гра між генератором і дискримінатором, яку можна формалізувати у вигляді математичної моделі:

$$\min_G \max_D \left\{ \log(D(x)) + \log(1 - D(G(z))) \right\}$$

й яка зводиться до дискримінатора, що здатен прогнозувати усе з імовірністю 0,5.

Проте на практиці мінімаксна гра часто призводить до проблем зі збіжністю мережі, тому важливим є ретельне налаштування параметрів навчання.

Позначимо результуюче зображення нейромережі, як I_{SR} . Зображення із низькою роздільною здатністю, яке потребує збільшення в n разів, як I_{LR} . Зображення високої роздільної здатності (без опрацювання) позначимо як I_{HR} . Таким чином, I_{LR} зображення отримуємо в результаті зменшення зображення I_{HR} в n разів.

У результаті навчання нейромережі потрібно одержати функцію генерування. При цьому генератор навчається як згорткова нейромережа прямого поширення з параметрами θ_G :

$$\theta_G = \{W_{1:L}; b_{1:L}\}, \quad (2.1)$$

де $W_{1:L}$ і $b_{1:L}$ – це ваги і зміщення нейромережі із глибиною в L шарів відповідно. Такий параметр обчислюють на сонові методу оптимізації функції втрат l_{SR} . У результаті навчання нейромережі обчислюємо значення функції:

$$\theta_G = \min \frac{1}{N} l_{SR} \left(G(I_n^{LR}), I_n^{HR} \right), \quad (2.2)$$

Функція втрат l_{SR} складається із таких параметрів:

$$l_{SR} = l_{MSE} + 10^{-3} l_{GEN}, \quad (2.3)$$

де l_{MSE} – середньоквадратична похибка відхилення згенерованого зображення I_{SR} від оригіналу I_{HR} ;

l_{GEN} – функція втрат, яку описують на основі формули:

$$l_{\text{Gen}} = \sum_{n=1}^N -\log(D(G(I^{\text{LR}}))), \quad (2.4)$$

де $D(G(I^{\text{LR}}))$ – це імовірність розпізнавання дискримінатором того, що зображення $G(I_{\text{LR}})$, створене генератором, високої роздільної здатності I_{HR} .

Таким чином, GAN-мережі уможливають отримати реалістичні, текстурнобагаті реконструкції й можуть відновлювати відсутні деталі, що не забезпечують традиційні методи. Ризики при цьому включають «галюцинації», за яких генератор вигадує деталі, що не відповідають істині.

2.3 Алгоритм формування вибірки даних для навчання нейромережі

Для побудови моделі реконструкції зображень із застосуванням нейромереж вибірку даних, зазвичай, розподіляють на три підмножини: навчальна, валідаційна та тестова. Як правило, близько 70 % даних відводять на навчання моделі, 20 % – на валідацію для налаштування параметрів і перевірки проміжної точності, а решту 10 % – на тестування для оцінки продуктивності моделі.

Алгоритм формування вибірки зображень для навчання GAN включає такі кроки:

- крок 1. Визначення типу даних і характеристик майбутніх зображень, які мають бути відтворені генератором;
- крок 2. Після обраної цільової області, здійснюється збір вихідних даних із доступних відкритих чи локальних джерел.

– крок 3. Приведення даних до єдиних технічних параметрів, включаючи роздільність, формат та кольоровий простір, що забезпечує однорідність набору та спрощує навчання моделі.

– крок 4. Перевірка якості даних – видаляють зіпсовані, розмиті або дубльовані зображення, оскільки їх наявність призводить до деградації моделі або неправильного формування розподілу даних.

– крок 5. Стандартизація зображень – нормалізація інтенсивностей, вирівнювання яскравості та контрасту.

Одним із поширених способів отримання зображень низької роздільної здатності (*LR*) є погіршення вихідних зображень високої якості (*HR*) шляхом застосування розмиття, додавання шуму або врахування особливостей формату JPEG та ефектів квантування.

Існує низка відкритих ресурсів із готовими базами зображень, таких як CIFAR-10, MNIST, Labelme, більшість із яких зберігається у форматах архівів .gz або .zip. Для ефективного навчання нейромережі переважно використовують зображення з великою кількістю деталей у колірному просторі RGB.

Для організації навчання нейромережі використовують бібліотеку Keras, яка дозволяє не лише тренувати глибокі нейромережі, а й має доступні попередньо навчені моделі. Це уможлиблює заощадити час та обчислювальні ресурси, не моделюючи архітектуру мережі з нуля. У Keras попередньо навчені мережі містяться в модулі applications і здебільшого призначені для розпізнавання об'єктів на зображеннях із набору ImageNet, який є масштабною базою анотованих зображень для тестування методів комп'ютерного зору. Серед доступних моделей можна виділити VGG16, VGG19, Inception v3, ResNet50, Xception та CRNN, що застосовуються для класифікації зображень та ідентифікації ознак.

Набір даних ImageNet включає більше 14 мільйонів зображень із зазначенням об'єкта на кожному з них. У цьому наборі передбачено близько 21

тисячі класів об'єктів, які утворюють ієрархічну структуру, що дозволяє, наприклад, не лише розпізнати “собаку”, а й конкретний його вид (вівчарка, дворняга, тощо), що робить ImageNet придатним для навчання з учителем.

Найбільша роздільна здатність монохроматичних знімків становить 3360×3360 пікселів. Такий формат обрано з певних міркувань, зокрема з урахуванням використання модифікованої мережі UNET, яка розрахована на обробку зображень із вхідною розмірністю $20 \times 224 \times 224$, при цьому $3360 = 224 \times 15$. Число 224 часто застосовується як сторона квадрата у багатьох відомих згорткових нейромережах, таких як VGG16 та ResNet, для яких існують попередньо підготовлені ваги на великому наборі даних ImageNet. Це дозволяє використовувати підготовлені зображення для експериментів із зазначеними архітектурами мереж.

Усі доступні зображення приведено до формату 3360×3360 та об'єднано в одну тривимірну матрицю розміром $20 \times 3360 \times 3360$. Полігони проектуються на маску зображення аналогічного розміру, де значення 1 означає наявність об'єкта за координатами відповідного пікселя, а значення 0 – його відсутність.

Для підготовки тренувального набору дані розділено окремо для кожного з десяти класів сегментації. Матриця та маска розбиваються на 225 неперетинних фрагментів розміром $20 \times 224 \times 224$ та 224×224 відповідно. У масках ділянки, що містять хоча б один полігон класу, позначено червоним кольором, а число в ділянці відображає відсоток площі зображення, зайнятої полігонами поточного класу.

Розподіл даних виявився нерівномірним, оскільки деякі класи відсутні на окремих зображеннях, а інші мають дуже малу площу полігонів відносно загальної площі. Тому тренувальні зображення варто готувати окремо для кожного класу. Після початкового етапу, на якому обираються неперетинні зображення розміром $20 \times 224 \times 224$, до кожного класу додаються додаткові фрагменти з ненульовою маскою, отримані методом плаваючого вікна навколо ненульових областей.

Приклад окремого тестового зображення, що складається з 20 шарів, подано на рисунку 2.11, а відповідну маску наведено на рисунку 2.12.

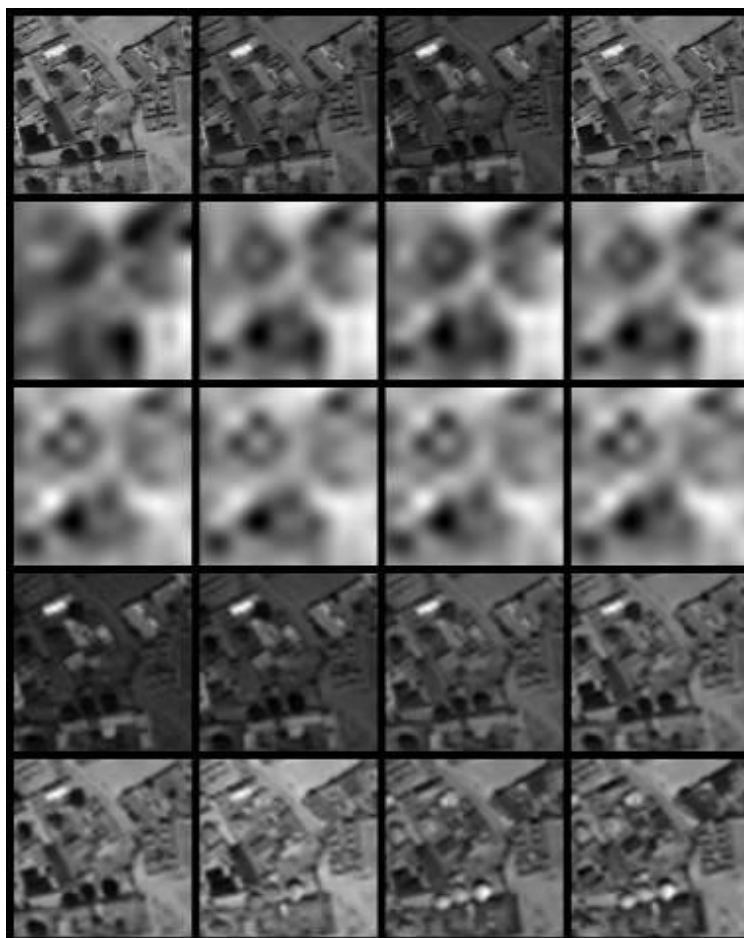


Рисунок 2.11 – Тестовий приклад зображення



Рисунок 2.12 – Маска зображення

Результатом описаного алгоритму є структурована, збалансована й узгоджена вибірка, яка коректно відображає розподіл реальних даних та забезпечує стабільне навчання GAN-мережі й високу якість реконструкції зображень.

2.4 Модель архітектури генеративної мережі для реконструкції зображень

Архітектура генеративної нейромережі побудована таким чином, щоб забезпечити поступове відновлення просторових деталей зображення та стабільність процесу навчання. Мережа складається з двох згорткових шарів із ядром 3×3 та 64 картами ознак у кожному. Після кожного шару згортки застосовується нормалізаційний шар, що сприяє стабілізації градієнтів і пришвидшує збіжність моделі.

Подальше масштабування просторової роздільної здатності здійснюється за допомогою двох субпіксельних згорткових шарів (PixelShuffle), які забезпечують ефективне підвищення роздільності зображення без суттєвого підвищення обчислювальної складності.

Як функцію активації обрано Parametric ReLU, що уможливорює моделлю адаптивно коригувати коефіцієнт для негативних значень, завдяки чому покращується здатність мережі відтворювати дрібні структури та запобігати проблемі «загасання» градієнтів:

$$f(x) = \begin{cases} x, & x > 0 \\ ax, & x \leq 0 \end{cases} \quad (2.7)$$

де x – сигнал вхідний,

a – коефіцієнт, що налаштовується у результаті навчання моделі.

«Для пришвидшення роботи нейромережі застосовано метод батчнормалізації (Batch Normalization), який уможливує додати додатковий шар, що здійснює нормалізацію з метою зниження рівня спотворення даних. Зазначений метод на сонові застосування вищого значення параметра швидкості дозволяє зменшити час навчання (learning rate) й при цьому знизити ризик перенавчання» [24].

Архітектуру генератора нейромережі наведено на рисунку 2.13. Вона включає такі компоненти:

- Input I_{LR} – зображення низької якості;
- Conv – шар згортки із n картами ознак (feature maps) та кроком зсуву (stride) – s ;
- ReLU – функція активації, що забезпечує адаптивну обробку негативних значень і підвищує здатність моделі відновлювати дрібні деталі зображення.

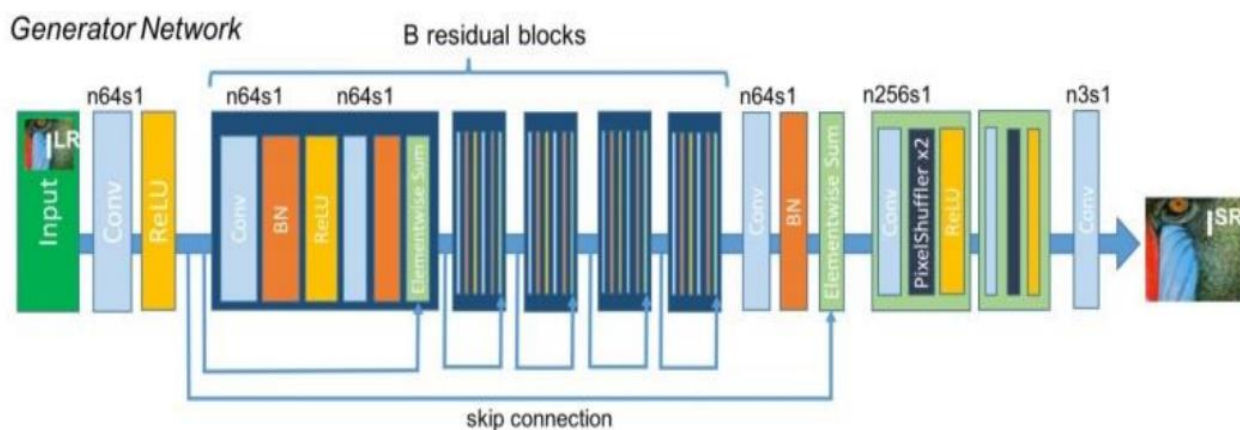


Рисунок 2.13 – Архітектура генератора

- BN – Batch Normalization;
- skip connection – зв'язки додаткові;
- B residual blocks – блоки залишкові;
- Elementwise sum – сума поелементна;
- PixelShuffler x2 – субпіксельні шари згортки;

– I_{SR} – результат виходу нейромережі – високоякісне зображення.

Для розпізнавання зображень високої якості I_{HR} від згенерованих I_{SR} проводиться навчання дискримінатора. На його вхід випадковим чином подаються як I_{HR} , так і I_{SR} зображення. На виході дискримінатор видає 0, якщо зображення визначено як I_{SR} та 1 – якщо воно віднесене до I_{HR} . Архітектуру дискримінатора наведено на рисунку 2.14.

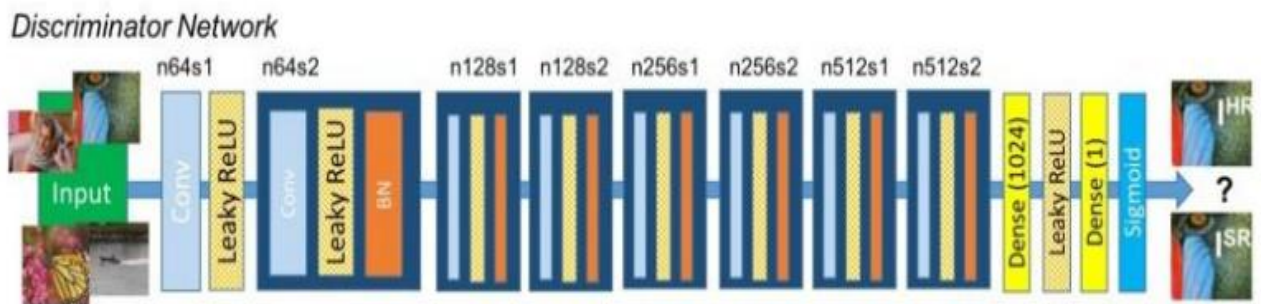


Рисунок 2 – Архітектура дискримінатора

Елементи архітектури дискримінатора:

- Input – вхідні зображення високої якості та зображення, згенеровані генератором;
- Conv – шар згортки з картами ознак (feature maps) n і кроком (stride) s ;
- Leaky ReLU – активаційна функція;
- BN – Batch Normalization;
- Dense – шар повнозв'язний;
- Sigmoid – функція активації сигмоїда.
- I_{HR} або I_{SR} -зображення.

Для об'єктивної оцінки якості реконструкції зображення на основі генеративних нейромереж використовують метрики, які оцінюють наскільки реконструйоване зображення подібне до оригінального [40].

Метрика IS (Inception Score) – це показник якості згенерованих зображень у генеративних моделях. Inception Score якість зображень, тобто наскільки

класифікатор упевнений у передбаченні класу для згенерованого зображення та різноманітність зображень – наскільки отримані результати відрізняються між собою. IS високий, якщо кожне зображення має чіткі та однозначні ознаки об'єкта (класифікатор впевнений) або різні зображення належать до різних класів.

Метрика Mean Squared Error (MSE) дозволяє оцінити середньоквадратичну похибку відхилення між оригінальним та відновленим зображенням. Низьке значення MSE свідчить про точніше відновлення.

Peak Signal-to-Noise Ratio (PSNR) уможливорює визначити відношення максимально можливого сигналу до рівня шуму і застосовується для порівняння якості відновлених зображень. Вищі значення PSNR відповідають кращій якості відновлення.

Structural Similarity Index (SSIM) враховує структурні та текстурні властивості зображень і вимірює ступінь подібності структури між оригіналом та відновленим зображенням. Значення SSIM, зазвичай, коливаються від -1 до 1, причому вищі показники відображають більшу подібність за текстурою, контрастом і яскравістю.

Mean Absolute Error (MAE) оцінює середнє абсолютне відхилення між оригінальним та відновленим зображенням.

Розглянуті метрики дозволяють об'єктивно оцінити, наскільки ефективний метод відновлення якості зображень, у зазначеному випадку архітектура генеративної нейромережі.

2.5 Висновки до розділу

Проаналізовано поширені архітектури штучних нейромереж, як універсальних засобів опрацювання даних. Обґрунтовано вибір архітектури генеративної нейромережі для задачі реконструкції зображень. Розроблено

алгоритм формування вибірки даних (зображень) для побудови архітектури нейромережі й зазначено, що зображення повинні бути однакового формату роздільної здатності та кольорового простору для забезпечення ефективної процедури навчання. Розроблено модель генеративної нейромережі для реконструкції зображень, описано елементи генератора та дискримінатора.

РОЗДІЛ 3 ПРОГРАМНА СИСТЕМА РЕКОНСТРУКЦІЇ ЗОБРАЖЕНЬ

3.1 Структура програмної системи

Програмна система реконструкції зображень призначена для автоматизованого відновлення якісних зображень з частково пошкоджених, зашумлених або неповних вхідних даних шляхом застосування методів глибокого навчання. Система поєднує модулі підготовки вибірки даних, навчання нейромережових моделей, прогнозування та валідації якості, забезпечуючи ітеративний процес покращення результатів та інтеграцію з користувацьким інтерфейсом. Діаграму варіантів використання системи реконструкції зображено представлено на рисунку 3.1.

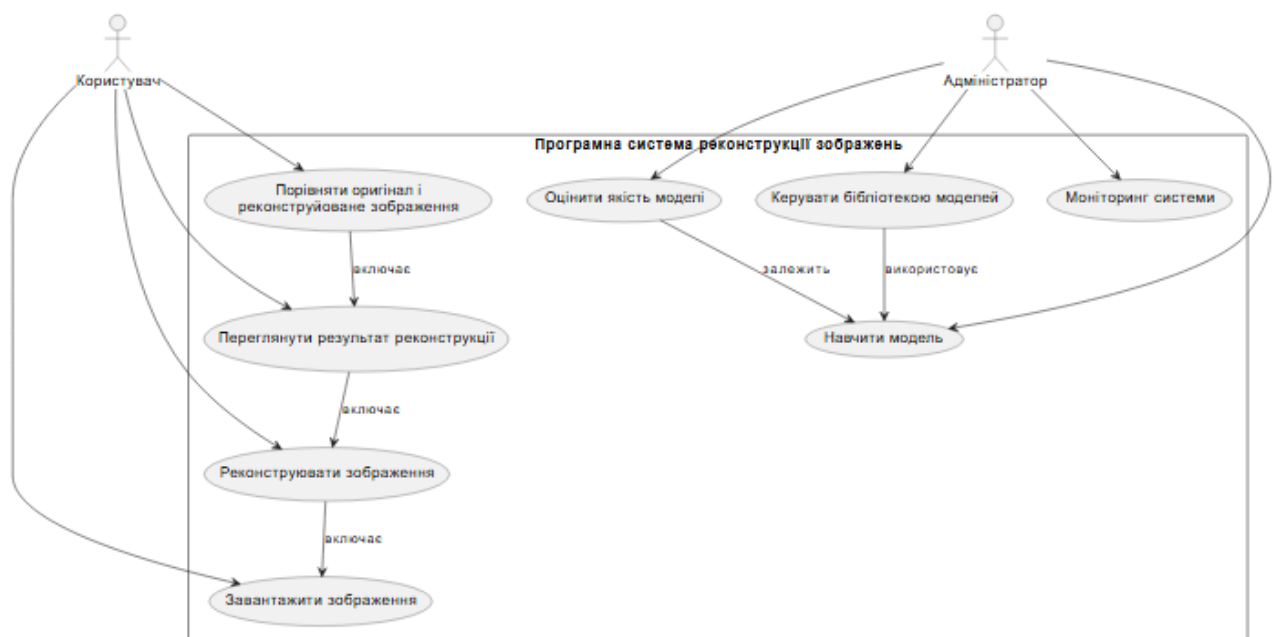


Рисунок 3.1 – Діаграма варіантів використання

Діаграма варіантів використання програмної системи реконструкції зображень відображає основних учасників системи та їхню взаємодію з функціями. Актори – Користувач та Адміністратор виконують свої задачі.

Користувач взаємодіє із системою переважно для реконструкції зображень. Він має можливість завантажити оригінальне зображення, ініціює процес його реконструкції, після чого може переглянути отриманий результат та порівняти його з вихідним зображенням. Ці варіанти використання логічно пов'язані між собою, адже реконструкція виконується після завантаження, перегляд – після реконструкції, а порівняння – після перегляду результату.

Адміністратор виконує розширені системні функції, пов'язані з підготовкою та оптимізацією моделі. Він може запускати процес навчання нейромережі, оцінювати якість отриманої моделі, керувати бібліотекою моделей (додавати нові, видаляти застарілі, обирати активну модель для роботи системи) та здійснювати моніторинг роботи системи. Частина варіантів використання адміністратора взаємопов'язана – оцінювання якості можливе лише після навчання моделі мережі, а керування моделями залежить від результатів навчання.

Структура програмної системи реконструкції зображень складається з логічних взаємопов'язаних модулів (підсистем), кожен з яких виконує окрему функцію.

1. Підсистема підготовки даних відповідає за імпорт, фільтрацію, нормалізацію та сигментацію зображень, а також за формування навчальних, тестових і валідаційних вибірок.

2. Підсистема моделювання та навчання архітектури нейромережі – містить архітектури моделей GAN-мереж, механізми навчання, функції втрат, оптимізатори та модуль параметричного налаштування.

3. Підсистема оцінки якості реконструкції реалізує обчислення метрик (PSNR, SSIM), візуалізацію результатів і порівняння оригінал-реконструкція.

4. Підсистема прогнозування забезпечує застосування натренованої моделі нейромережі для відновлення нових зображень у реальному часі або пакетному режимі.

5. Підсистема зберігання даних і моделей містить сховище датасетів, збережених версій моделей, проміжних результатів та журналів експериментів.

6. Підсистема сервісів та API надає зовнішній інтерфейс для інтеграції з іншими системами.

7. Підсистема управління та моніторингу відповідає за логування, моніторинг продуктивності, аналітику роботи моделі та управління конфігураціями системи.

Зважаючи на зазначене, можна побудувати діаграму послідовності виконання реконструкції зображення, яку продемонстровано на рисунку 3.2.

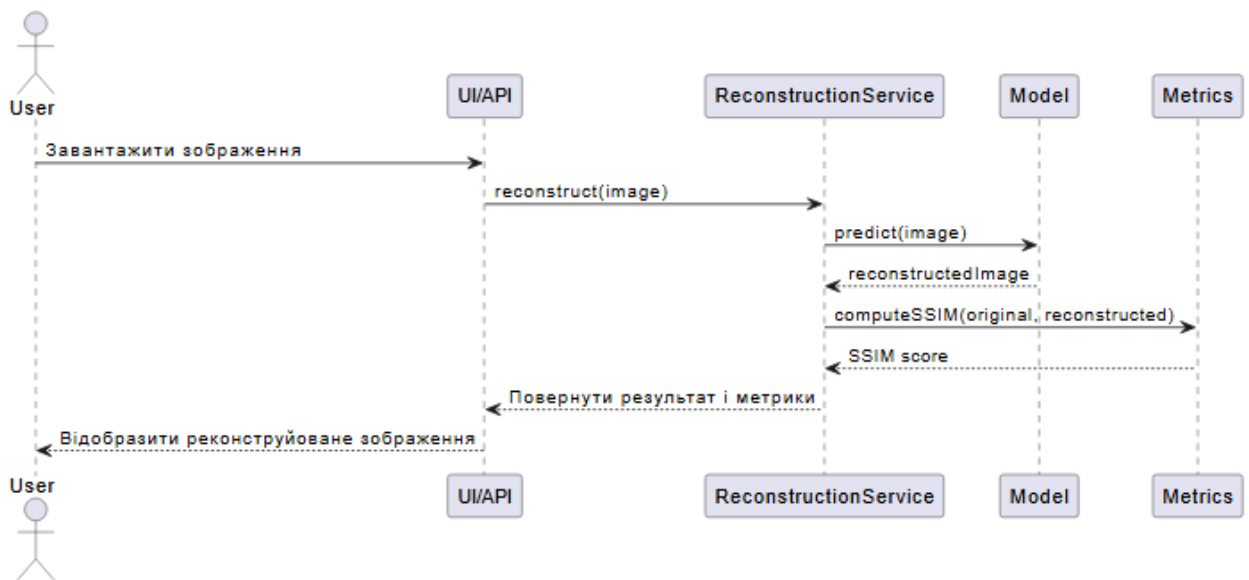


Рисунок 3.2 – Діаграма послідовності

Спочатку Користувач завантажує зображення через інтерфейс, після чого запит передається до модуля реконструкції. Модуль надсилає зображення до моделі нейромережі, яка здійснює прогноз і повертає реконструйовану версію.

Отримавши результат, система додатково передає оригінал і реконструйоване зображення до модуля оцінки метрик, який обчислює показники якості. Після цього сервіс формує відповідь, що містить реконструйоване зображення та обчислені метрики, і повертає її назад до API. Інтерфейс відображає результат користувачу.

3.2 Архітектура GAN-мережі для реконструкції зображень

Розробку програми, яка здатна покращити якість зображень на основі нейромережі використаємо бібліотеку мови Python [20, 25, 36], TensorFlow. TensorFlow є потужним інструментом для розробки архітектур генеративних нейромереж завдяки своїй гнучкості, оптимізації та розвиненій екосистемі. Він дозволяє розробляти як прості автоенкодерів, так і складні GAN-архітектури. Фреймворк підтримує інструменти розробки моделей як у високорівневому стилі через Keras [14], так і в низькорівневому форматі, що уможливорює точно контролювати архітектуру, шари та процес навчання [22, 23].

TensorFlow забезпечує ефективне виконання генеративних моделей на GPU і TPU, що важливо для обчислювально складних процесів, характерних для генерації зображень або сигналів. Використання бібліотек TensorFlow Addons та TensorFlow Probability дає змогу реалізовувати вдосконалені типи шарів, стохастичних компонентів чи спеціалізованих функцій втрат, які необхідні в генеративних задачах. Завдяки екосистемі TensorFlow легко організувати тренувальні цикли, логування, візуалізацію та налагодження моделі.

Структуру коду для розробки моделі архітектури генеративної нейромережі за допомогою бібліотеки TensorFlow наведено нижче:

```
import tensorflow as tf
from tensorflow.keras import layers, losses, models, optimizers
# Завантаження даних та підготовка їх до навчання
# (input_images, target_images) – пари низькоякісних та високоякісних
зображень
#Розробка моделі: model = models.Sequential()
# Додання шарів CNN: model.add(layers.Conv2D(64, (3, 3), activation='relu',
padding='same', input_shape=(image_height, image_width, 3)))
```

```

#Компіляція моделі:
model.compile(optimizer=optimizers.Adam(learning_rate=0.001),
loss=losses.MeanSquaredError())

# Тренування моделі: model.fit(input_images, target_images,
batch_size=batch_size, epochs=num_epochs)

# Збереження навченої моделі: model.save('image_enhancement_model.h5')

Після завершення навчання модель можна застосовувати до нових
зображень для підвищення їх якості та відновлення деталей.

# Завантаження моделі, що вже навчилась: model =
tf.keras.models.load_model('image_enhancement_model.h5')

# Завантаження низькоякісного зображення, яке необхідно відновити:
input_image = load_input_image()

# Покращення зображення на основі розробленої моделі: enhanced_image
= model.predict(input_image)

# Збереження реконструйованого зображення:
save_enhanced_image(enhanced_image)

# Компілюємо модель та задаємо функцію втрат:
input_shape = (29, 29, 1)
model = create_model(input_shape)
model.compile(optimizer='adam', loss='binary_crossentropy')

# Тренуємо модель на даних без дефектів (вхідні та вихідні дані
однакові): model.fit(x_train, x_train, epochs=10, batch_size=128, shuffle=True,
validation_data=(x_test, x_test))

# Зберігаємо модель: model.save('autoencoder_model.h5')

# Перевіряємо модель на основі тестових даних (зображень з дефектами):
decoded_images = model.predict(x_test)

```

Слід зауважити, що існує проблема, пов'язана з нерівномірністю навчання, коли дискримінатор навчається надто ефективно, його вихідні

значення наближаються до 0 або 1, що ускладнює генератору отримання корисного градієнта для навчання.

Якщо ж генератор дуже добре навчений, то він використовуватиме неточності дискримінатора, які призводитимуть до помилкового опрацювання.

У такому випадку дискримінатор навчається значно частіше, ніж генератор, і вже не оцінює ймовірність належності події до конкретного класу, як у бінарній класифікації, а визначає відповідність події реальним міткам класу. Інакше кажучи, вихід дискримінатора не проходить через функцію активації і сприймається не як ймовірність, а як кількісна оцінка вхідних даних. Функція втрат у цьому випадку визначається як середнє відхилення прогнозованих значень від реальних для справжніх та згенерованих даних.

На рисунку 3.3 показано структурну схему роботи генеративної неромережі.

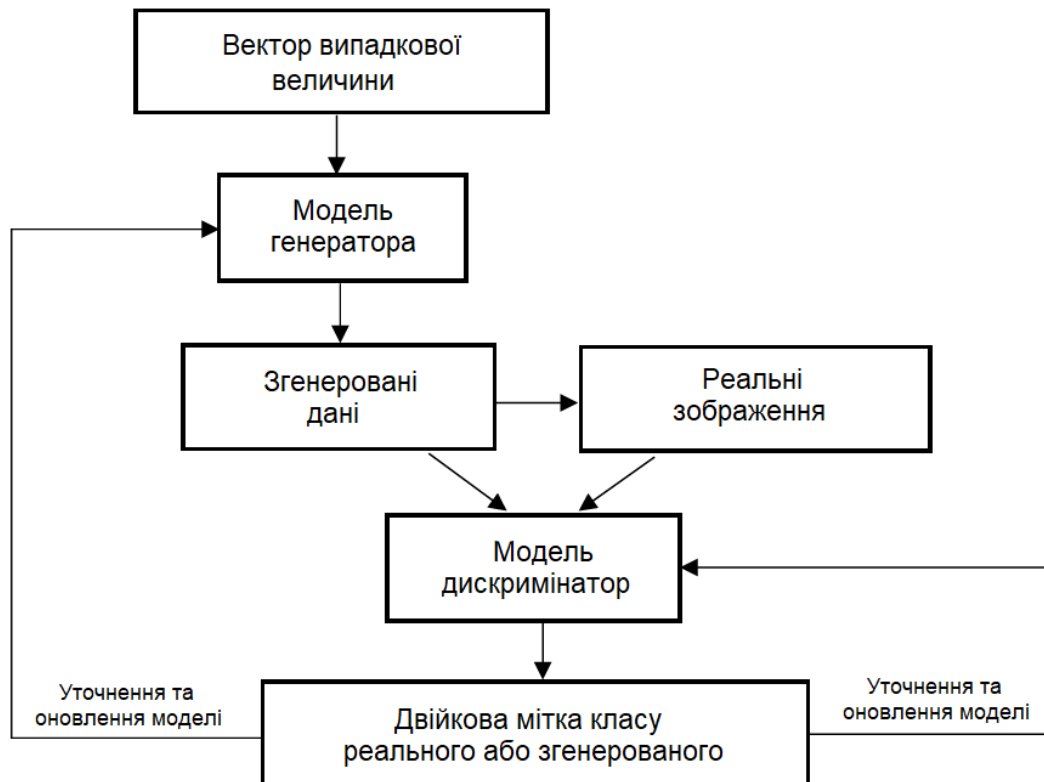


Рисунок 3.3 – Структурна схема процедури реконструкції зображення на основі генеративних нейромереж

На рис. 3.4 представлено графік залежності значення функції втрат від кількості епох навчання. Як видно із графіка, чим більше епох навчання нейромережі, тим менша похибка відхилення прогнозного значення від реального, тобто функція втрат близька до 1.

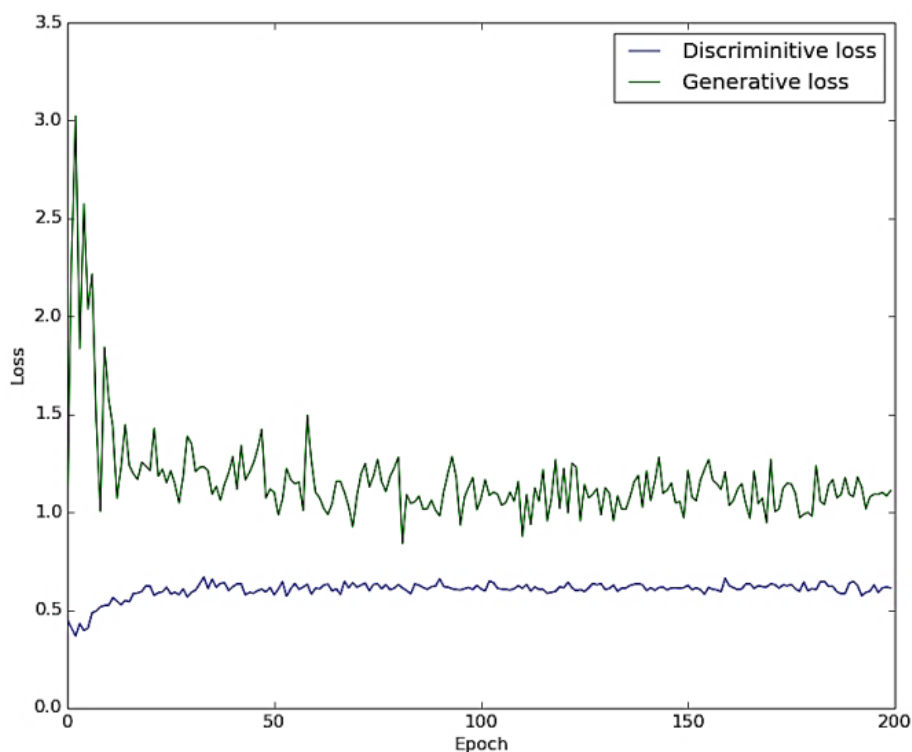


Рисунок 3.4 – Графік функції втрат

Згенеровані зазначеним чином зображення можуть бути використані для розширення навчальних та тестових вибірок при розв'язанні задач реконструкції зображень.

На рисунку 3.5 наведено результати навчання дискримінатора нейромережі, де по осі x відображено кількість епох навчання, а по осі y – значення функції втрат. Як видно із графіка, на 49 епосі навчання функція втрат різко почала знижуватися і на 50 епосі досягла свого мінімуму.

На рисунку 3.6 зображені результати навчання генератора, тобто залежність функції втрат від кількості епох навчання.



Рисунок 3.5 – Графік навчання дискримінатора



Рисунок 3.6 – Результат навчання генератора

Як видно із рисунка 3.6, функція втрат генератора, починаючи із 16 епохи, є стабільною. Найвищого значення вона досягає на перших епохах навчання.

Отже, на основі функції втрат оцінюємо помилки відхилення між реальними зображеннями та результативними, отриманими у результаті роботи генеративної нейромережі, а основною метою навчання, як зазначено у розділі 2, є мінімізація цієї функції помилки.

У математичному сенсі процес навчання GAN-мережі розглядають як мінімаксну гру двох гравців – дискримінатор прагне мінімізувати помилку при розпізнаванні реальних і згенерованих екземплярів, тоді як генератор

навчається максимізувати ймовірність того, що дискримінатор допустить помилку.

На рисунку 3.7 наведено модель точності та функцій втрат для процедури навчання та валідації розробленої моделі генеративної нейромережі на реальних зображеннях.

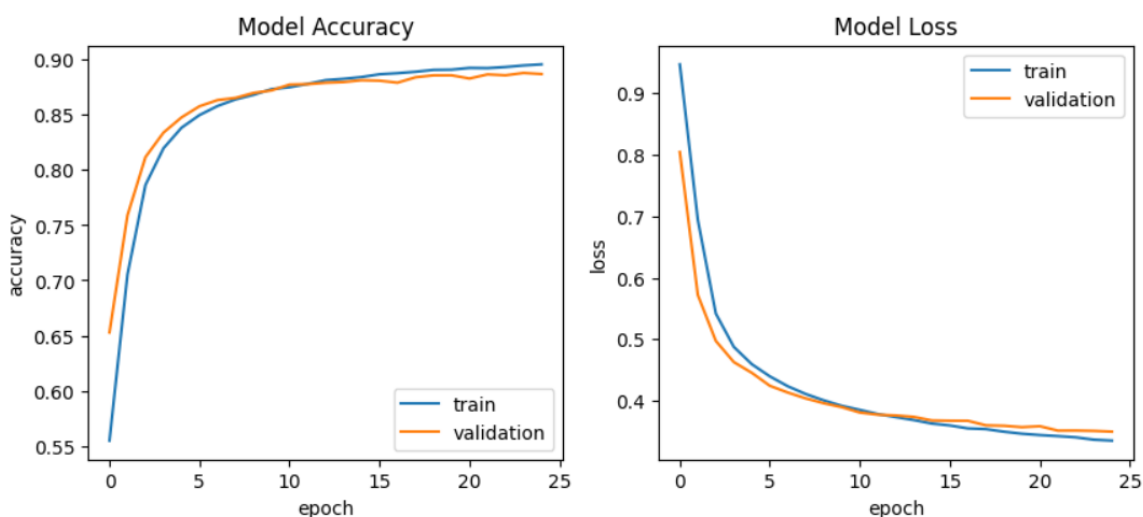


Рисунок 3.7 – Результати навчання та перевірки архітектури GAN-мережі для реконструкції зображень

Під час експерименту реалізовано багатошарову GAN-мережу. Для набору даних MNIST, який включає 10 класів (параметр $N = 10$). Навчання мережі проводилося протягом 8 епох на 50 000 тренувальних зображеннях. На рисунку 3.8 показано результати навчання за метрикою IS , де зліва представлено початкові нечіткі, спотворені зображення із шумом ($IS = 5,13$), а праворуч – кінцевий, якісніший результат ($IS = 7,12$). Це свідчить про прогрес у навчанні нейромережі.

Другий експеримент стосувався генерації числових даних із нерозмічених датасетів. Згідно із запропонованим методом, перед генерацією необхідно кластеризувати вихідні дані за N категоріями. Спочатку проведено навчання та оцінку GAN-мережі на датасеті HAR. Експерименти показали високі значення метрики IS для генерації нерозмічених даних різної розмірності. Для

дослідження IS виконано низку експериментів із різною кількістю категорій N – від 10 до 1500 (див. рисунок 3.9). Навчання мережі тривало 1000 епох із замірами IS кожних 10 епох для різних значень N .

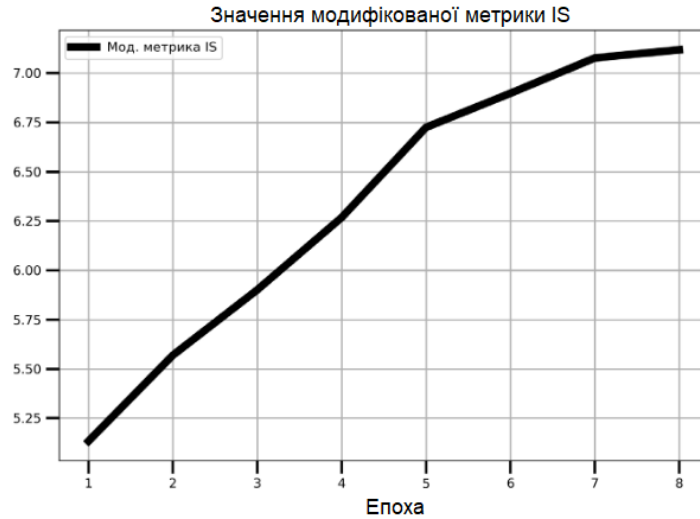


Рисунок 3.8 – Процес навчання GAN-мережі на наборі даних MNIST

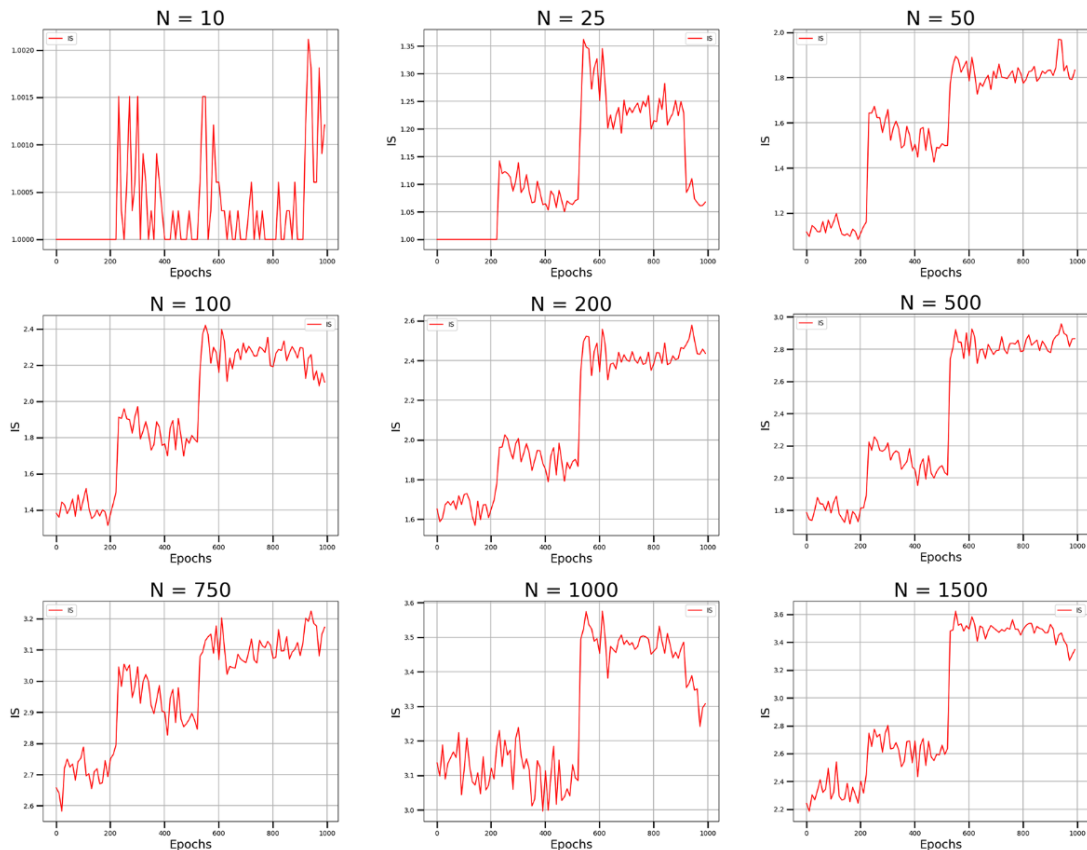


Рисунок 3.9 – Дослідження якості зображень при навчанні GAN-мережі на наборі даних HAR

Одержані результати уможливають зробити певні висновки, які стосуються оцінки роботи генеративної нейромережі. Для $N \leq 25$ процес навчання коректно оцінити неможливо – значення IS у цьому діапазоні або коливається в межах однієї величини, або знижується. Таким чином, оцінити продуктивність GAN-мережі неможливо. Натомість при $N \geq 25$ оцінка навчання відображається коректно – зі збільшенням кількості ітерацій значення IS зростає, а помилки дискримінатора та генератора зменшуються.

Важливо зазначити, що коректність оцінки на основі метрики IS зберігається навіть для $N > 1500$. Водночас при $N > 500$ спостерігається незначне підвищення значення показника IS , що пояснюється надмірністю кластерів із невеликими відмінностями, що також призводить до зростання обчислювального навантаження. Отже, з метою отримання оптимальних результатів варто застосовувати $N \approx 500$.

У майбутніх дослідженнях можна застосувати методи, які виключають дуже подібні кластери, що зменшить обчислювальні витрати й призведе до отримання коректніших результатів.

3.3 Експериментальні дослідження GAN-моделі при реконструкції зображень

Задачу реконструкції зображень із використанням GAN-нейромережі представляємо як модель, що складається з двох компонентів – генератора та дискримінатора. Дискримінатор навчається розпізнавати реальні зображення від згенерованих генератором, тоді як генератор у процесі навчання поступово вдосконалюється, намагаючись продукувати більш реалістичні зображення. Після завершення навчання генератор здатний відтворювати зображення, які неможливо відрізнити від оригінальних за критеріями, встановленими дискримінатором.

Далі представлено прототип додатку для реконструкції зображень, реалізований на мові Python. Інтерфейс додатку забезпечує відображення початкового та кінцевого зображень після обробки, включно з початковою фільтрацією. На рисунку 3.10 показано приклад відфільтрованого зображення.

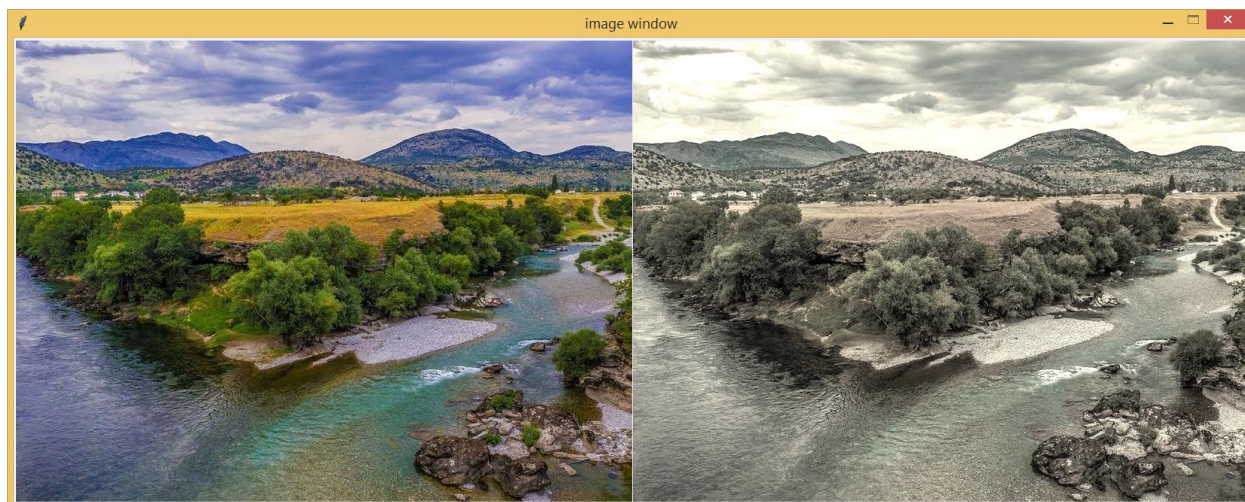


Рисунок 3 – Результат фільтрації зображення

При запуску додатку в середовищі Python результати реконструкції відображаються в головному вікні, з розрахунком параметра найбільших співпадінь, який визначається під час розпізнавання та зіставлення ознак зображення. Для відкриття зображення необхідно скористатися відповідним пунктом меню, що наведено на рисунку 3.11.

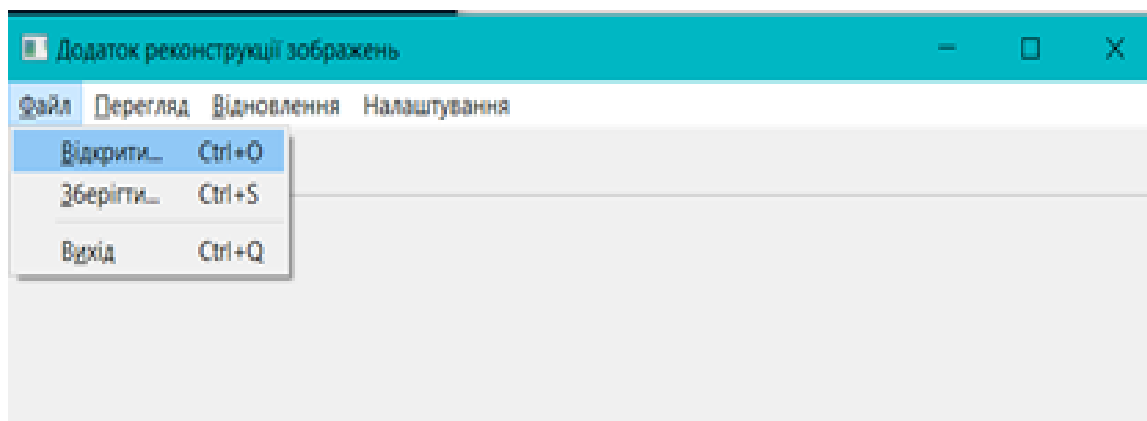


Рисунок 3.11 – Екранна форма вікна інтерфейсу користувача системи реконструкції зображень

Після відкриття системи користувач має можливість налаштувати параметри процесу відновлення зображення. Усі зміни можна виконати у діалоговому вікні, яке зображено на рисунку 3.12. Тобто, можна змінити формат зображення, розмити його, зблизити.

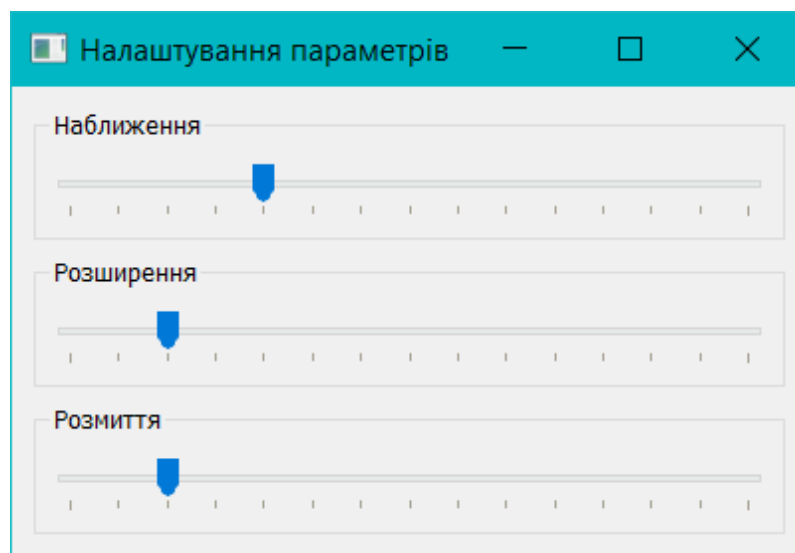


Рисунок 3.12 – Екранна форма вікна інтерфейсу користувача для налаштування параметрів

Початок процедури реконструкції відбувається після того, як вибрано відповідні пункти меню із випадаючого вікна, як зазначено на рисунку 3.13.

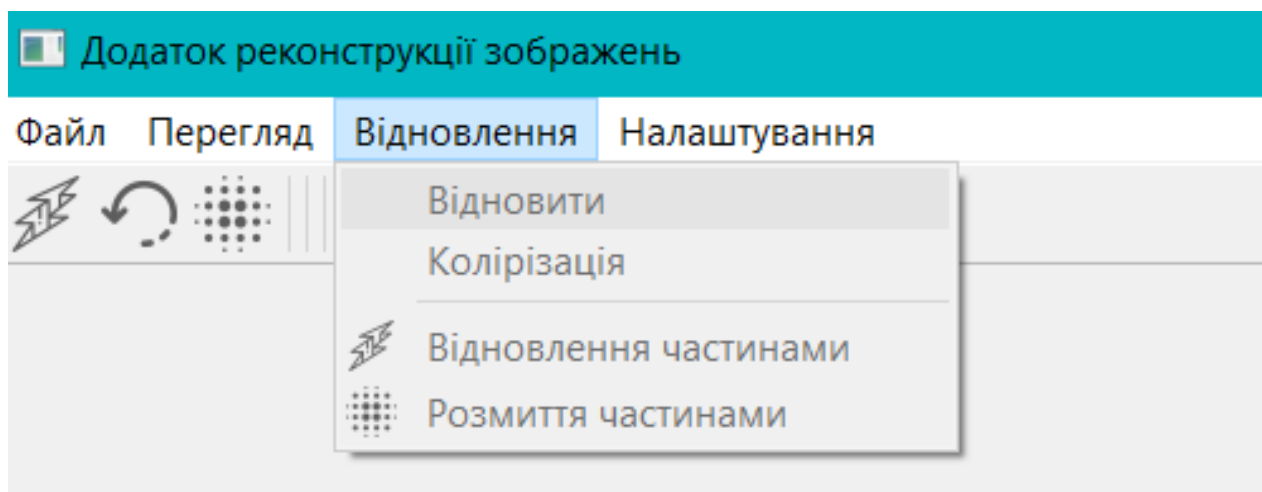


Рисунок 3.13 – Вікно вибору режимів обробки зображення

Проведемо реконструкцію зображення, яке містить пошкодження. Для цього спочатку завантажимо зображення-оригінал та його відфільтровану версію (див. рисунок 3.14).

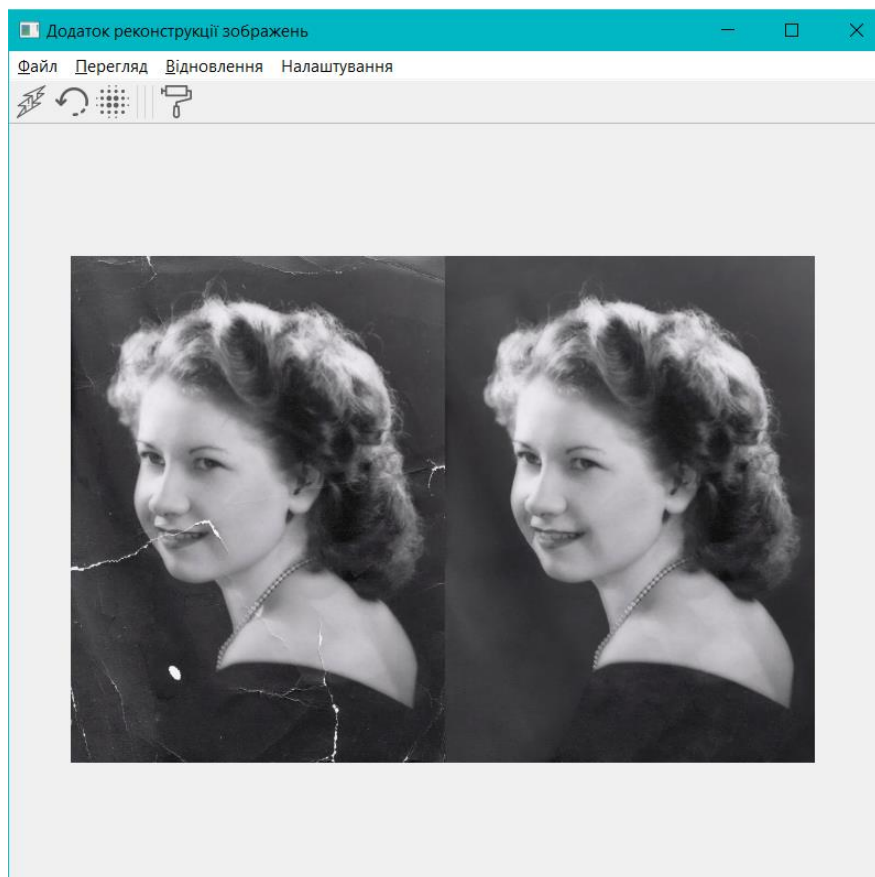


Рисунок 3.14 – Режим відкриття зображення

Результат реконструкції зображення на основі генеративної нейронної мережі показано на рисунку 3.15.

Тепер перейдемо до оцінки роботи розробленої моделі архітектури мережі в порівнянні із класичними архітектурами нейромереж, зокрема багат шаровим перцептроном. Тестування проведено на вибірці даних MNIST, яка містить понад 50000 зображень. У таблиці 3.1 наведено результати порівняльного аналізу параметрів роботи нейромереж при 5 епохах навчання.

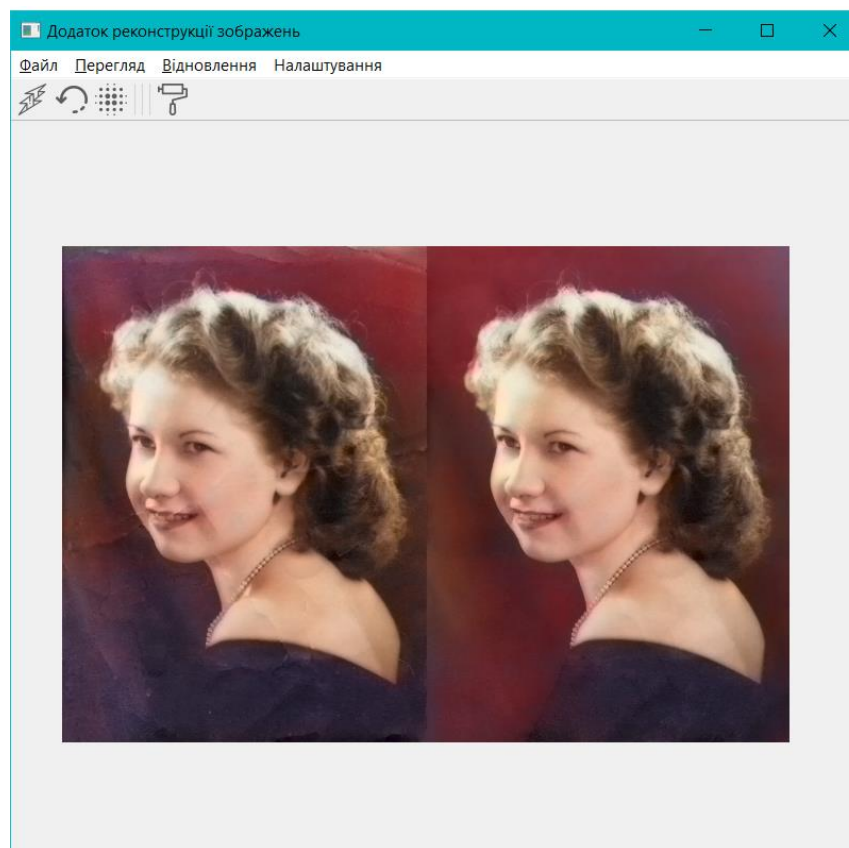


Рисунок 3.15 – Результат реконструкції та колоризації зображення

Таблиця 3.1 – Порівняння показників роботи неймереж при реконструкції зображень

Неймережа	Точність		Функція помилки		Час тренування
	Навчальна	Тестова	Навчальна	Тестова	
Багатошарвий перцептрон	99,4	98	0,016	0.068	112
GAN	99,5	98,9	0.015	0.034	122

Із аналізу результатів таблиці 3.1 можна зробити висновок, що модель генеративної неймережі точніше відтворює зображення (значення точності на тестовій вибірці даних становить близько 99%). Хоч по часу навчання розроблена модель неймережі програє багатошаровому перцептроні, проте це не є ключовим параметром оцінки адекватності.

Такі результати підтверджують перспективність застосування генеративних архітектур неймереж в задачах реконструкції зображень. В той

же час, варто зауважити, що такі архітектури здатні відновлювати зображення із суттєвими артефактами.

3.4 Висновки до розділу

Розроблено програмний прототип моделі, що включає генератор, дискримінатор та механізм їхнього змагального навчання. Архітектура була реалізована на основі фреймворку TensorFlow, що забезпечило гнучкість у побудові мереж, налаштуванні функцій втрат, оптимізаторів і обробленні тренувальних даних. Генератор сконструйований за принципами глибоких згорткових мереж із використанням, що дозволило формувати більш чіткі та структурно коректні реконструкції. Дискримінатор реалізовано як згорткову мережу, здатну розрізняти справжні та реконструйовані зразки, що підвищило якість навчання завдяки змагальному процесу. Здійснені експериментальні дослідження підтвердили, що реалізована архітектура здатна ефективно реконструювати зображення з артефактами, зменшувати шум та відновлювати втрачені області. Отримані результати демонструють практичну придатність моделі для задач відновлення цифрових зображень та створюють основу для подальшого вдосконалення архітектури й алгоритмів навчання.

ВИСНОВКИ

У результаті виконання кваліфікаційної роботи було розв'язано науково-прикладну задачу розробки алгоритмів реконструкції зображень на основі нейромереж. При цьому отримано вагомі результати.

1. Проаналізовано специфіку задач реконструкції зображень, включаючи відновлення пошкоджених фрагментів, усунення артефактів та підвищення якості й визначено основні труднощі, зокрема, втрата інформації, неоднозначність відновлення та залежність від структури зображення.

2. Проаналізовано методи та програмні засоби реконструкції зображень й обґрунтовано доцільність застосування нейромережевого підходу, що базується на здатності до навчання та розробки прогнозних моделей на основі здобутих знань й програмних засобів TensorFlow, Keras, як готові бібліотеки інструментів для обробки зображень.

3. Проаналізовано існуючі архітектури штучних нейромереж, виділено як оптимальну генеративну архітектуру для реконструкції зображень завдяки високій точності відновлення зображень навіть з артефактами.

4. Проаналізовано принципи роботи згорткових та глибоких моделей та механізми виділення ознак й встановлено, що згорткові шари забезпечують локальне відновлення структури, а глибокі генеративні моделі дозволяють узгоджувати глобальний контекст зображення. Це підтвердило доцільність об'єднання цих підходів у розробленій архітектурі.

5. Охарактеризовано алгоритм навчання генеративних нейромереж, що ґрунтується на змагальному підході генератора та дискримінатора з метою мінімізації функції втрат.

6. Розроблено алгоритм підготовки вибірки даних для навчання нейромереж, що включає процедури нормалізації, масштабування, штучного додавання артефактів, поділ на навчальні та тесту вальні, що забезпечило стабільність навчання та підвищило узагальнювальну здатність мережі.

7. Розробити архітектуру генеративної нейромережі для реконструкції зображень, що уможлиблює реконструкцію зображень із високою точністю.

8. Проведено навчання моделі на підготовленому даних даних, виконано її оцінювання за метрикою IS й результатами дослідження підтверджено перспективність застосування таких архітектур для реконструкції зображень з різними типами артефактів.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. 3D Slicer image computing platform: веб-сайт. URL: <https://www.slicer.org> (дата звернення 10.10.2015 р.).
2. AI vs. Machine Learning vs. Deep Learning vs. Neural Networks: What's the difference?: веб-сайт. URL: <https://www.ibm.com/blog/ai-vs-machine-learning-vs-deep-learning-vs-neural-networks> (дата звернення 09.11.2023).
3. Beazley D. Python_Cookbook_3rd_Edition_2015. Jones: O'Reilly Media. 2016. 706 p.
4. Breiman, L. Bagging predictors. Machine Learning. 1996. Vol. 24 (2). Pp. 123- 140/
5. Buduma N. Fundamentals of Deep Learning. Designing Next-Generation Machine Intelligence Algorithms. O'Reilly Media. 2017. 298 p.
6. Chollet F. Deep Learning with Python. Manning Publications. 2017. – 386 p.57.
7. Chollet, F. Xception: Deep Learning with Depthwise Separable Convolutions, Google, Inc. Retrieved from https://openaccess.thecvf.com/content_cvpr_2017/papers/Chollet_Xception_Deep_Learning_CVPR_2017_paper.pdf (дата звернення 11.11.2023).
8. Deep Learning in Medical Image Analysis: веб-сайт. URL: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5479722/> (дата звернення 15.11.2025).
9. Geron, A. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow. 2nd Edition. O'Reilly Media, Inc. Retrieved from: <https://www.oreilly.com/library/view/hands-on-machinelearning/9781492032632/> (дата звернення 05.11.2025).
10. Goodfellow I., Bengio Y., Courville A. Deep Learning. MIT Press, 2016: веб-сайт. URL: <http://www.deeplearningbook.org> (дата звернення 20.09.2025).

11. Goodfellow I., Pouget-Abadie J., Mirza M., Xu B., Warde-Farley D., Ozair S., Courville A., Bengio Y. "Generative adversarial nets" // Advances in neural information processing systems, 2014. P. 2672–2680.
12. Gordienko Yu., Gang P., Hui J. et al. Deep Learning with Lung Segmentation and Bone Shadow Exclusion Techniques for Chest X-Ray Analysis of Lung Cancer // The First International Conference on Computer Science, Engineering and Education Applications. Springer, 2018. Vol 754. P. 638-647.
13. ITK Advanced Course: веб-сайт. URL: <https://www.kitware.com/courses/itk-advanced> (дата звернення 10.10.2015 р.).
14. Keras Documentation. keras.io: веб-сайт. URL: <https://keras.io/applications/> (дата звернення 12.11.2025).
15. Keras: веб-сайт. URL: http://cloud-5.bitp.kiev.ua/?page_id=600 (дата звернення 05.11.2025).
16. Kuhlman Dave. A Python Book: Beginning Python, Advanced Python, and Python Exercises. Archived from the original, 2012. 673 p.
17. MATLAB: веб-сайт. URL: <https://www.mathworks.com/products/matlab.html> (дата звернення 10.10.2015 р.).
18. NVIDIA Clara: веб-сайт. URL: <https://docs.nvidia.com/clara/index.html> (дата звернення 10.10.2015 р.).
19. Oksana Shkurat, Yevgeniya Sulema, Viktoriya Suschuk-Sliusarenko, Andrii Dychka, Image Segmentation Method Based on Statistical Parameters of Homogeneous Data Set // Advances in Intelligent Systems and Computing. Springer, 2019. Vol. 902. P. 271-281.
20. Python Documentation: веб-сайт. URL: <https://www.python.org/doc/> (дата звернення 13.11.2025).
21. Shukla N. Manning Early Access Program Machine Learning with TensorFlow. Manning Publications, 2017. 244 p.
22. Tensorflow Tutorial 2: image classifier using convolutional neural network: веб-сайт. URL: <https://cvtricks.com/tensorflow-tutorial/training-convolutional-neural-network-for-imageclassification/> (дата звернення 11.11.2025).

23. TensorFlow: веб-сайт. URL: <https://www.tensorflow.org/datasets/catalog/mnist> (дата звернення 14.10.2015 р.).
24. What is Batch Normalization In Deep Learning? веб-сайт. URL: <https://www.geeksforgeeks.org/deep-learning/what-is-batch-normalization-in-deep-learning> (дата звернення 14.10.2015 р.).
25. Бенгфорт Б. Прикладний аналіз текстових даних на Python. Машинне навчання та створення додатків обробки природної мови, 2019. 368 с.
26. Білашенко С.В., Шаповалова Н.Н., Рибальченко О.Г. Розпізнавання зображень за допомогою згорткових нейронних мереж з використанням бібліотеки Keras // Гірничий вісник, 2018. Вип. 103. С. 148-154.
27. Біляєва Є.І. Реконструкція зображень з використанням глибинного навчання: веб-сайт. URL: <https://openarchive.nure.ua/server/api/core/bitstreams/78da4599-4222-4bde-8d45-2c8f8b477d78/content> (дата звернення 16.09.2025 р.).
28. Вовк Л. Б., Мальчиков В. В., Юр'єва К. Г. Підхід до реставрації фотографій на основі глибокого навчання. Прикладна математика та комп'ютинг. ПМК-2023: шістнадцята науково-практична конференція магістрантів та аспірантів. Київ, 2023 р. С. 23-27.
29. Згорткова нейронна мережа – просте пояснення CNN та її застосування: веб-сайт. URL: <https://evergreens.com.ua/ua/articles/cnn.html> (дата звернення 12.11.2023).
30. Зубрович Р.Е., Приходько алгоритм реконструкції зображень на основі глибоких нейромереж // Матеріали III всеукраїнської науково-практичної конференції студентів, аспірантів та молодих вчених «Інтелектуальні комп'ютерні системи та мережі. Тернопіль:ЗУНУ, 2025. С. 135-135.
31. Каллан, Р. Нейронні мережі: короткий довідник. К.: Вільямс І.Д., 2017. 288 с.
32. Матвієнко В., Криницький В. Штучні нейромережі в задачах розпізнавання зображень // Матеріали VIII науково-практичної конференції

молодих вчених та студентів «Інтелектуальні комп'ютерні системи та мережі». Тернопіль, 2023. С. 17.

33. Методичні вказівки до оформлення курсових, звітів про проходження практики, випускних кваліфікаційних робіт для студентів спеціальності «Комп'ютерна інженерія» / І.В. Гураль, Л.О.Дубчак / під ред. О.М. Березького. Тернопіль: ТНЕУ, 2019. 34 с.

34. Методичні рекомендації до виконання кваліфікаційної роботи з освітнього ступеня “Магістр” спеціальності 123 «Комп'ютерна інженерія» Магістерська програма - Комп'ютерна інженерія" / О.М. Березький, Г.М. Мельник / Під ред. Л.О.Дубчак. Тернопіль: ЗУНУ, 2024. 32 с.

35. Мірецький В.В., Приходько О.В. Системи контролю показників здоров'я людини // Матеріали XVIII міжнародної науково-практичної конференції «Інформаційні технології та автоматизація – 2025». Одеса, 30-31 жовтня 2025. С.

36. Плас Д. Python для складних завдань. Наука про дані і машинне навчання. Керівництво. К.: ВМС, 2018. 759 с.

37. Редько В.Г. Еволюція, нейронні мережі, інтелект: Моделі і концепції еволюційної кібернетики. М.: Ленанд, 2019. 224 с.

38. Синєглазов В., Чумаченко О. Глибокі нейронні мережі для вирішення завдань розпізнавання і класифікації зображення: веб-сайт. URL: <http://itcm.comp-sc.if.ua/2017/Sineglazov.pdf> (дата звернення 11.11.2025).

39. Чорний, С., Брендель, О., Гратіашвілі, Д. Автентифікація зображень на основі їх семантичної сегментації у нейронних мережах глибокого навчання з їх попереднім обробленням за методами фільтрації. Теорія та практика судової експертизи і криміналістики. 2022. Вип. 1 (26). С. 125-137.

40. Юр'єва К.Г. Математичне та програмне забезпечення системи видалення артефактів з фотографій: магістерська дисертація. Київ:КПІ імені І.Сікорського, 2024. 100 с.

41. Як працює VGG16 — нейромережа для виділення ознак зображень: веб-сайт. URL: <https://robotdreams.cc/uk/blog/319-yak-pracyuye-vgg16> (дата звернення 09.11.2025).