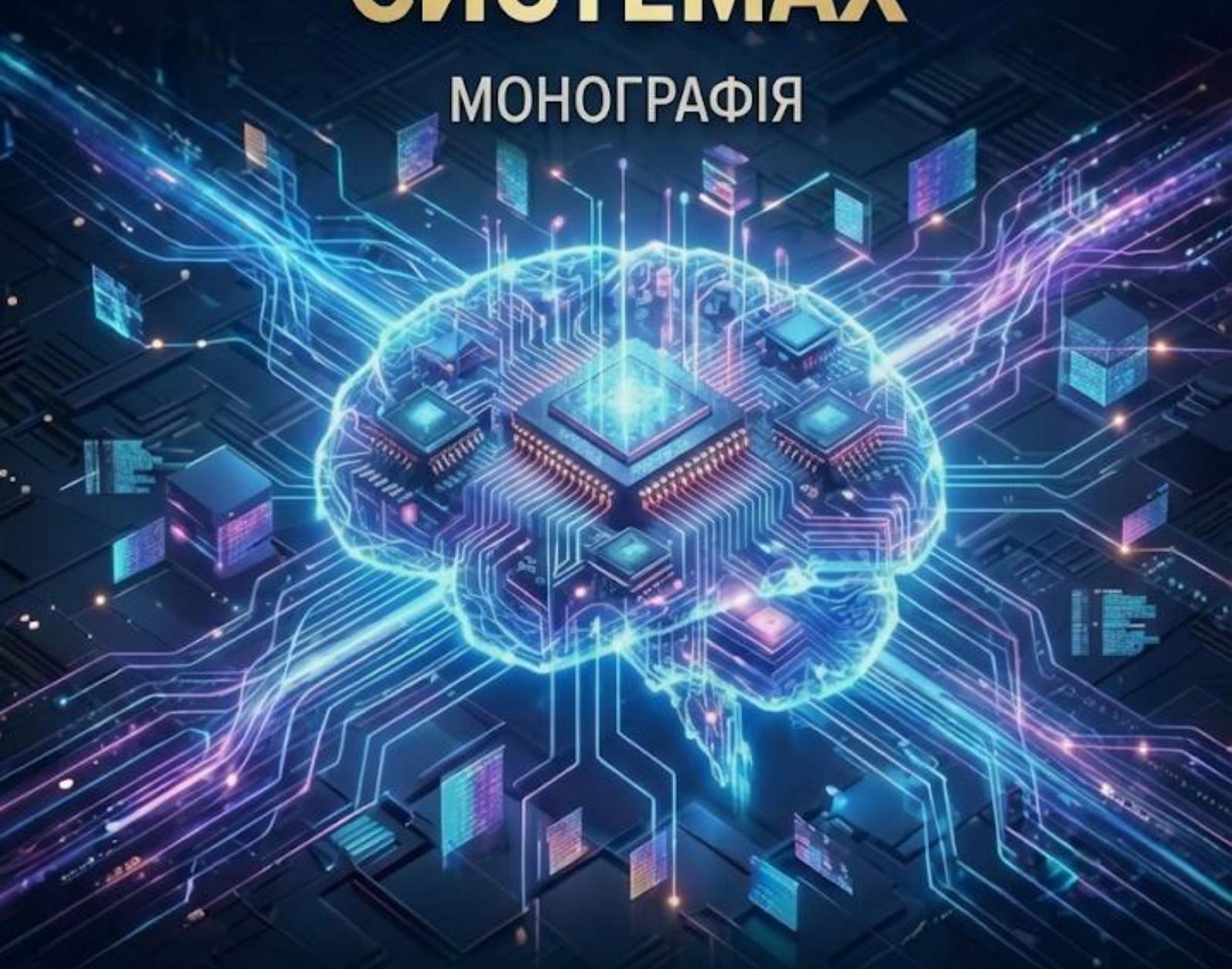


ЗБЕРІГАННЯ ВЕЛИКИХ ДАНИХ В ІНТЕЛЕКТУАЛЬНИХ СИСТЕМАХ

МОНОГРАФІЯ



**КИРИЧЕНКО І.В., ТЕРЕЩЕНКО Г.Ю.,
ШАНІДЗЕ Н.О., ШАРОНОВА Н.В.**



**ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
«ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»
МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**КИРИЧЕНКО І.В., ТЕРЕЩЕНКО Г.Ю.,
ШАНІДЗЕ Н.О., ШАРОНОВА Н.В.**

МОНОГРАФІЯ

**ЗБЕРІГАННЯ ВЕЛИКИХ ДАНИХ В ІНТЕЛЕКТУАЛЬНИХ
СИСТЕМАХ**

Харків 2026

УДК 004.4:004.8

Затверджено до видання Науково-технічною радою ХНУРЕ,
протокол № 1 від 30.01.26 р.

Затверджено до видання Вченою радою НТУ «ХП»,
протокол № 3 від 27.02.26 р.

Рецензенти:

Федорович О.Є., д.т.н., проф., завідувач кафедри комп'ютерних наук та інформаційних технологій Національного аерокосмічного університету «Харківський авіаційний інститут», лауреат Державної премії України в галузі науки та техніки

Литвин В.В., д.т.н., проф., завідувач кафедри інформаційних систем та мереж Національного університету «Львівська політехніка»

Колектив авторів:

Кириченко І.В., к.т.н., доцент, доцент кафедри Програмної інженерії
(Харківський національний університет радіоелектроніки),

Терещенко Г.Ю., PhD, доцент кафедри Програмної інженерії
(Харківський національний університет радіоелектроніки),

Шанідзе Н.О., к. соц. н., доцент, доцент кафедри Соціології і публічного управління
(Національний технічний університет «Харківський політехнічний інститут»),

Шаронова Н.В., д.т.н., професор, професор кафедри інтелектуальних комп'ютерних систем
(Національний технічний університет «Харківський політехнічний інститут»),

Кириченко І.В.

Зберігання великих даних в інтелектуальних системах : монографія /
І.В. Кириченко, Г.Ю. Терещенко, Н.О. Шанідзе, Н.В. Шаронова – Харків:
ХНУРЕ, 2026. – 222 с.

Монографія містить опис сучасних уявлень про зберігання та обробку великих даних на основі вітчизняної та світової теорії та практики.

У монографії представлені результати досліджень, які виконують актуальне наукове завдання створення методів для формування гібридної моделі сховища зображень в умовах великих даних, що має істотне значення для розвитку комп'ютерних наук, зокрема у галузі штучного інтелекту.

Для інженерів, науковців, викладачів, аспірантів, діяльність яких пов'язана з роботою з різними типами даних, такими як зображення, звук, відео, текст тощо.

Іл.32, Табл. 23, Бібліогр. 198 назв.

ISBN 978-966-659-380-4

DOI: 10.30837/978-966-659-380-4

© Кириченко І.В.,

Терещенко Г.Ю.,

Шанідзе Н.О.,

Шаронова Н.В., 2026

АНОТАЦІЯ

Сучасний науковий прогрес активно просуває технології, випереджаючи їхній розвиток і впровадження нововведень. Раніше для зберігання великого обсягу зображень доводилося витратити значний час і ресурси. Тепер майже кожна компанія спрямована на поліпшення процесу зберігання шляхом інтеграції систем стиснення зображень у свої продукти. Системи стиснення здатні працювати з різними типами даних, такими як зображення, звук, відео, текст тощо. Вони відрізняються одна від одної своєю архітектурою, типом пам'яті, тощо.

У монографії представлені результати досліджень, які виконують актуальне наукове завдання створення методів для формування гібридної моделі сховища зображень в умовах великих даних, що має істотне значення для розвитку комп'ютерних наук, зокрема у галузі штучного інтелекту.

Актуальність опрацювання даної теми полягає в тому, що за останні роки в галузі зберігання даних зросла потреба у надійних і безпечних методах забезпечення конфіденційності та цілісності інформації. В контексті розвитку технологій створення зображень, виникає велика потреба у збереженні та обміні цими даними в безпечних та надійних умовах. Використання блокчейн технологій може бути цілком корисним при дослідженні та розв'язанні задач, пов'язаних із зберіганням та обробкою великомасштабних наборів даних у вигляді зображень.

Актуальність обраної тематики пояснюється також тим, що розглянуто важливу науково-практичну задачу розробки моделі зберігання зображень, яка поєднує у собі переваги різних підходів до сховищ даних (наприклад, реляційних та нереляційних баз даних, файлових систем тощо). Основна мета проведених досліджень полягає в тому, щоб дослідити можливості гібридної моделі сховища зображень для ефективною та швидкою обробки великих обсягів даних. У процесі роботи над монографією було вивчено різні підходи до зберігання та обробки зображень, відповідні технології та стандарти, а

також **розроблено власну гібридну модель**, яка є оптимальною за певними критеріями для конкретних науково-практичних задач. Виходячи з викладеного, монографія присвячена опису розв'язання задачі створення гібридної моделі сховища зображень в умовах великих даних, що має істотне значення для розвитку комп'ютерних наук, зокрема у галузі штучного інтелекту.

Метою цієї роботи є підвищення ефективності моделей сховищ даних за рахунок створення формальної процедури організації сховищ зображень на основі гібридної моделі. Об'єктом дослідження є процеси створення гібридної моделі для побудови сховища зображень в умовах великих даних, а предмет дослідження – методи організації сховища зображень в умовах великих даних.

У монографії описано розроблений метод формування гібридної моделі сховища зображень, який поєднує переваги реляційних, нереляційних баз даних та файлових систем для забезпечення ефективного зберігання та обробки великих обсягів даних, що дозволило скоротити час доступу до зображень на 27-35% та підвищити точність класифікації зображень до 95,4%. Авторами запропоновано алгоритм оптимального розміщення зображень, що знижує час доступу до даних до 65% відносно класичних моделей, та підвищує продуктивність системи (точність класифікації зросла до 95,4%, пропускна здатність зросла у 1,84 разів). Запропоновано новий підхід до інтеграції блокчейн-технологій для забезпечення конфіденційності та цілісності збереження інформації в умовах великих даних, що дозволило збільшити швидкість пошуку за векторними представленнями (FAISS, OpenAI CLIP) у блокчейн-сховищі до 38% у порівнянні зі стандартними методами.

Описано удосконалену методіку комплексного аналізу ефективності зберігання даних, яка враховує специфіку різних типів сховищ та їх інтеграцію, що має практичне значення для розробки сучасних систем штучного інтелекту. При цьому отримала подальший розвиток інформаційна технологія зберігання, обробки та аналізу великих обсягів зображень, яка

інтегрує всі розроблені методи та алгоритми, поєднуючи AI, блокчейн та гібридне сховище. Проведене експериментальне дослідження ефективності розробленої інформаційної технології на різних наборах даних (медичні зображення, супутникові знімки, цифрове мистецтво), підтвердило її переваги над існуючими. Швидкість доступу до збережених файлів у гібридному сховищі зросла в 4,3 рази – середній час доступу 1,05 с. проти 4,5 с. у порівнянні з існуючими системами.

Результати дослідження мають практичну значимість для різних сфер, які працюють із зображеннями, та забезпечують нові можливості для розвитку інформаційних технологій. Розроблено та експериментально перевірено алгоритми стиснення та дедуплікації зображень для ефективного використання в блокчейн-сховищах. В результаті, обсяг даних було зменшено на 47% без втрати критичної інформації, що підтверджено SSIM-метрикою (0,92). Ефективність використання пам'яті у гібридному сховищі становить 91% проти 70% у класичних централізованих системах

Запропоновано новий алгоритм, який дозволив оптимально розміщувати зображення в гібридній моделі, значно зменшити час доступу до даних та поліпшити продуктивність роботи системи. Автоматична обробка зображень забезпечує підвищену ефективність аналізу медичних зображень, допомагає визначати патології та прискорити діагностування. Проведено експериментальне дослідження ефективності розробленої гібридної моделі на різних наборах даних (медичні зображення, супутникові знімки, цифрове мистецтво), яке підтвердило її переваги над існуючими підходами. Швидкість доступу до збережених файлів у гібридному сховищі зросла в 4,3 рази – середній час доступу 1,05 с. проти 4,5 с. у стандартних системах.

Запропоновані методи та інформаційна технологія можуть бути повною мірою впроваджені в організаціях всіх галузей, які використовують сховища даних у вигляді зображень, для підвищення якості роботи систем штучного інтелекту.

Ключові слова: інтелектуальний аналіз даних, гібридна модель, блокчейн технології, реляційна база даних, глибинне навчання, інформаційна технологія, машинне навчання, база знань, медичні послуги, програмні агенти, згорткові нейронні мережі, нейромережева обробка зображень, розпізнавання зорових об'єктів, інтелектуальні системи, обробка даних, смарт-контракт, сховище зображень, штучний інтелект, розподілені мережі та системи.

ЗМІСТ

АНОТАЦІЯ	3
ВСТУП	9
РОЗДІЛ 1 ОБРОБКА ТЕКСТІВ ТА ЗОБРАЖЕНЬ В УМОВАХ ВЕЛИКИХ ДАНИХ.....	17
1.1 Обробка великих даних у текстовій формі та у вигляді зображень	17
1.1.1 Типи сховищ зображень.....	19
1.1.2 Реляційні та нереляційні бази даних	24
1.2 Методи обробки та аналізу зображень засобами глибинного навчання та комп'ютерного зору.....	26
1.3 Аналіз досліджень блокчейн технологій для зберігання та захисту зображень	30
1.4 Існуючі проблеми у сфері архітектури сховищ зображень в епоху великих даних.....	34
РОЗДІЛ 2 ГІБРИДНА МОДЕЛЬ СХОВИЩА ЗОБРАЖЕНЬ	43
2.1 Методи аналізу великих даних для розробки архітектури сховищ зображень	43
2.2 Метод оптимізованої індексації в гібридній моделі зберігання зображень	50
2.3 Обґрунтування методів зберігання та обробки зображень для реалізації гібридної моделі	65
2.3.1 Структурне моделювання потоків даних у гібридному сховищі	66
2.3.2 Керування великими обсягами даних зображень в архітектурі сховища зображень.....	70
2.3.3 Функціональне моделювання процесів у гібридному сховищі	76
2.3.4 Підтримка безпеки та конфіденційності даних зображень	81
2.4 Тестування та оцінка ефективності розробленої гібридної моделі	84
2.5 Оцінка результатів експерименту	89

РОЗДІЛ 3 БЛОКЧЕЙН СИСТЕМИ ДЛЯ ЗБЕРІГАННЯ ТА ЗАХИСТУ	
ЗОБРАЖЕНЬ	99
3.1 Умови використання блокчейн технологій для зберігання та захисту	
зображень	99
3.1.1 Вирішення задачі стиснення зображень для зберігання у	
децентралізованих блокчейн сховищах.....	100
3.1.2 Застосування методів та технологій стиснення зображень.....	105
3.2 Метод аналізу стиснення зображень для зберігання в	
децентралізованих блокчейн сховищах	114
3.3 Практичне застосування методу блокчейна для зберігання зображень	
.....	126
3.3.1 Тестування прототипу системи на базі блокчейна на прикладі	
медичних зображень.....	128
3.3.2 Аналіз переваг та обмежень блокчейна в контексті реального часу та	
масштабованості	130
3.3.3 Особливості інтеграції блокчейн-технологій з існуючими ІТ-	
інфраструктурами	132
3.4 Проблеми та переваги використання блокчейн-систем для зберігання	
зображень	134
РОЗДІЛ 4 ПРАКТИЧНА РЕАЛІЗАЦІЯ ПРИКЛАДНИХ ЗАДАЧ ТА АНАЛІЗ	
РЕЗУЛЬТАТІВ	139
4.1 Використання машинного навчання для оптимізації доступу до даних в	
гібридному сховищі зображень	139
4.2 Застосування методу оптимізованої індексації в гібридній моделі	
зберігання зображень.....	156
4.3 Використання блокчейн-рішень для захисту цифрових активів	159
4.4 Задача аналізу медичних зображень на прикладі розпізнавання клітин	
крові	181
ЗАГАЛЬНІ ВИСНОВКИ	195
СПИСОК ЛІТЕРАТУРИ.....	198

ВСТУП

Сучасний науковий прогрес активно просуває технології, випереджаючи їхній розвиток і впровадження нововведень. Раніше для зберігання великого обсягу зображень доводилося витратити значний час і ресурси. Тепер майже кожна компанія спрямована на поліпшення процесу зберігання шляхом інтеграції систем стиснення зображень у свої продукти. Системи стиснення здатні працювати з різними типами даних, такими як зображення, звук, відео, текст тощо. Вони відрізняються одна від одної своєю архітектурою, типом пам'яті, тощо. Актуальність вибору та опрацювання даної теми полягає в тому, що за останні роки в галузі зберігання даних зросла потреба у надійних і безпечних методах забезпечення конфіденційності та цілісності інформації. В контексті розвитку технологій зображень, виникає велика потреба у збереженні та обміні цими даними в безпечних та надійних умовах. Використання блокчейн технологій може бути цілком корисним при дослідженні та розв'язанні задач, пов'язаних із зберіганням і обробкою великомасштабних наборів даних у вигляді зображень.

У наукові дослідження, що спрямовані на обробку великих даних значний внесок зробили видатні вчені, зокрема В.О. Філатов, К.С. Смеляков, G. Nath, H. Das, J.F. Kimpel, M. Jarke, M. Lenzerini, Y. Vassiliou, P. Vassiliadis, C. Blanco, I. de Guzmán, E. Fernández-Medina. Проблематика створення сховищ даних розглядалася не як окреме питання, а у взаємозв'язку з питаннями фрагментації даних, викликами, що пов'язані із «великими даними», та організацією розподіленого зберігання. Наприклад, у дослідженнях І.В. Шевченка, Л.Е. Чалої, А.М. Коппа, А. Cuzzocrea, L. Bellatreche, Y. Song, A. Boulmakoul, L. Karim, М.Н. Laarabi, R. Sacile, E. Garbolino, розглядаються аспекти структурування сховищ даних, створених на основі популярних платформ, таких як Hadoop, що досліджуються і у роботах Д.В. Приймака та О.І. Акимишина. Питання формалізації процесів структурування й обробки даних у таких сховищах, а також побудови просторів даних висвітлюються у

роботах авторів – Н.Б. Шаховської, Н.В. Шаронової, І.В. Кириченко, які зосереджуються на методах інтеграції неоднорідної інформації та оптимізації її використання.

Опис проблем, які можуть бути сформульовані в контексті дослідження, дозволяє згрупувати їх таким чином.

Першою проблемою є відсутність ефективного та безпечного рішення для зберігання та обробки великих обсягів зображень, що виникає в умовах зростання кількості цифрових даних та їх значення для різних галузей діяльності.

Другою проблемою є недостатня ефективність та продуктивність традиційних методів зберігання та обробки зображень, що призводить до затримок у роботі та незадовільної якості результату.

Третьою проблемою є відсутність надійного та ефективного методу захисту зображень від несанкціонованого доступу, що може призвести до порушення конфіденційності та безпеки даних.

Четвертою проблемою є недостатня розробленість та вивченість гібридних моделей зберігання та обробки зображень, які можуть поєднувати різні типи сховищ та методи обробки зображень для досягнення кращих результатів.

Загальною метою розв'язання описаних проблем є підвищення ефективності моделей сховищ даних за рахунок створення формальної процедури організації сховищ зображень на основі гібридної моделі. Для досягнення загальної мети з'ясувалось, що потрібно, перш за все, провести комплексний аналіз сучасних підходів до архітектури сховищ зображень в умовах великих даних, виявити ключові проблеми та перспективні напрями досліджень у галузі ефективного зберігання та обробки великомасштабних наборів візуальних даних. Далі, треба було визначити та проаналізувати основні методи та технології стиснення зображень для їх ефективного зберігання у гібридних та блокчейн-орієнтованих сховищах, а також провести

порівняльний аналіз алгоритмів компресії та їх застосування у децентралізованих сховищах.

Після такого аналізу доцільною стала розробка гібридної моделі сховища зображень, яка поєднує реляційні та нереляційні бази даних, файлові системи та блокчейн технології для зберігання та захисту зображень. При цьому важливим етапом роботи можна вважати розробку методів обробки зображень, які дозволяють ефективно та швидко обробляти великі обсяги зображень та забезпечувати високу якість результатів, а також методів захисту зображень від несанкціонованого доступу, що включають методи шифрування, контролю доступу та автентифікації користувачів.

Фінальним етапом роботи слід вважати проведення експериментального дослідження розробленої гібридної моделі та методів обробки та захисту зображень для визначення їх ефективності у порівнянні з іншими методами, а також оцінку практичної та наукової значущості розробленої гібридної моделі та методів, визначення їхньої можливості впровадження в різні сфери діяльності, які вимагають зберігання та обробки великих обсягів зображень.

Методи, які було застосовано, базувались на використанні методів штучного інтелекту, зокрема, машинного навчання, стиснення зображень, методів аналізу результатів за допомогою спеціалізованих програмних інструментів. Використання методів машинного навчання в контексті гібридних сховищ зображень дозволило значно покращити ефективність та точність обробки візуальних даних.

Як один із головних результатів, було запропоновано метод формування гібридної моделі сховища зображень, який поєднує переваги реляційних, нереляційних баз даних та файлових систем для забезпечення ефективного зберігання та обробки великих обсягів даних, що дозволило скоротити час доступу до зображень на 27-35% та підвищити точність класифікації зображень до 95,4 %. Крім того, було запропоновано алгоритм оптимального розміщення зображень, що знижує час доступу до даних до 65% відносно

класичних моделей, та підвищує продуктивність системи (точність класифікації зросла до 95,4%, пропускна здатність зросла у 1,84 разів).

Корисним практичним результатом є запропонований новий підхід до інтеграції блокчейн-технологій щодо забезпечення конфіденційності та цілісності збереження інформації в умовах великих даних, який дозволив збільшити швидкість пошуку за векторними представленнями (FAISS, OpenAI CLIP) у блокчейн-сховищі до 38% у порівнянні зі стандартними методами.

Практичне значення для розробки сучасних систем штучного інтелекту має удосконалена методика комплексного аналізу ефективності зберігання даних, яка враховує специфіку різних типів сховищ та їх інтеграцію. Цікавим результатом є інформаційна технологія зберігання, обробки та аналізу великих обсягів зображень, яка інтегрує всі розроблені методи та алгоритми, поєднуючи AI, блокчейн та гібридне сховище. Проведене експериментальне дослідження ефективності розробленої інформаційної технології на різних наборах даних (медичні зображення, супутникові знімки, цифрове мистецтво), підтвердило її переваги над існуючими. Швидкість доступу до збережених файлів у гібридному сховищі зросла в 4,3 рази – середній час доступу 1,05 с. проти 4,5 с. у порівнянні з існуючими системами.

Підсумовуючи, можна стверджувати, що практичне значення одержаних результатів полягає у розробці інформаційної технології обробки зображень та ефективному зберіганні даних. Розроблено бекенд на .NET 8 та фронтенд на Razor Pages для імпорту та стиснення зображень. Для імітації роботи реальної блокчейн мережі розгорнуто локальну блокчейн мережу Ethereum за допомогою Ganache й написано смарт контракт на мові Solidity, який завантажений в локальну блокчейн мережу за допомогою Truffle.

Проведено експеримент зі стискання зображень різними методами та збереженням їх в IPFS із фіксацією отриманих хешів у смарт-контрактах на локальній блокчейн мережі. За результатами експерименту зібрано та проаналізовано дані про розмір файлів, якість зображень (за метрикою PSNR), час завантаження в IPFS та пропускну здатність для кожного методу

стиснення. Візуалізовано результати дослідження у вигляді графіків та таблиць.

Розроблено алгоритм оптимізованої індексації та пошуку зображень, що базується на використанні моделей глибокого навчання (ResNet50) та векторних подібностей (FAISS, OpenAI CLIP). Запропоноване рішення дозволило зменшити накладні витрати з 0,08 с. у традиційних рішеннях до 0,019 с., а також підвищити швидкість пошуку за векторними представленнями до 38% у порівнянні зі стандартними методами індексації.

Розроблено та експериментально перевірено алгоритми стиснення та дедуплікації зображень для ефективного використання в блокчейн-сховищах. В результаті, обсяг даних було зменшено на 47% без втрати критичної інформації, що підтверджено SSIM-метрикою (0,92). Ефективність використання пам'яті у гібридному сховищі становить 91% проти 70% у класичних централізованих системах.

Проведене експериментальне дослідження ефективності розробленої гібридної моделі на різних наборах даних (медичні зображення, супутникові знімки, цифрове мистецтво), підтвердило її переваги над існуючими підходами. Швидкість доступу до збережених файлів у гібридному сховищі зросла в 4,3 рази – середній час доступу 1,05 с. проти 4,5 с. у стандартних системах.

Надамо короткий зміст та структуру монографії.

У першому розділі наведено детальний огляд сучасних літературних джерел з тематики обробки та зберігання великих даних. Зроблено огляд літератури з обробки великих даних у текстовій формі та у вигляді зображень, включаючи загальний огляд типів сховищ зображень та аналіз робіт з реляційних та нереляційних баз даних. У розділі розглянуто основні типи сховищ зображень, включаючи реляційні та нереляційні бази даних, файлові системи, та гібридні моделі. Аналіз досліджень блокчейн технологій для зберігання та захисту зображень дозволив сформулювати вимоги до методу аналізу стиснення зображень для зберігання в децентралізованих блокчейн

сховищах. Висновки аналізу джерел привели до обґрунтованої постановки задач дослідження.

Другий розділ монографії присвячений розробці гібридної моделі сховища зображень. Описані методи аналізу великих даних для архітектури сховищ зображень. Запропоновано метод оптимізованої індексації в гібридній моделі зберігання зображень для відповідного зберігання та доступу в умовах великих даних. Звернено увагу на вибір методів зберігання та обробки зображень для реалізації гібридної моделі. Особлива увага приділена керуванню великими обсягами даних зображень в архітектурі сховища зображень та підтримці безпеки та конфіденційності даних зображень. Наведені результати тестування та оцінка ефективності розробленої гібридної моделі. Проведена оцінка результатів експерименту. Зазначено, що проведене дослідження сприяє розробці архітектури сховищ зображень та управління ними, що є критичним викликом в епоху великих даних. Підвищуючи ефективність і масштабованість керування сховищем зображень, запропонований підхід може допомогти організаціям ефективно керувати своїми даними і отримувати з них цінну інформацію. Крім того, цей підхід можна застосувати до інших типів великих даних, таких як текстові та відеодані, що робить його значним внеском у ширшу сферу управління великими даними.

У третьому розділі монографії зосереджено увагу на розробці блокчейн-систем для зберігання зображень, аналізі їхніх архітектур, практичному застосуванні та інтеграції з існуючими ІТ-інфраструктурами. Виконано обґрунтування умов використання блокчейн технологій для зберігання та захисту зображень. Проведено дослідження задач стиснення зображень для зберігання у децентралізованих блокчейн сховищах, а також аналіз методів та технологій стиснення зображень. Представлено розроблений метод аналізу стиснення зображень для зберігання в децентралізованих блокчейн сховищах. Запропоновано архітектуру блокчейн-систем для захисту зображень, наведено практичне застосування блокчейну для зберігання зображень. Показані

проблеми та переваги використання блокчейн-систем для зберігання зображень.

У третьому розділі також проведено аналіз ефективності використання блокчейну у зберіганні великих обсягів зображень. Досліджено вплив різних методів децентралізованого зберігання, а також визначено оптимальний підхід на основі поєднання IPFS та Ethereum-смарт-контрактів. Запропоновано модель використання блокчейн-системи для зберігання та захисту зображень, яка дозволяє забезпечити баланс між розподіленим доступом до даних та їх безпекою. Використання IPFS та смарт-контрактів підвищує прозорість обміну зображеннями та гарантує незмінність даних.

Виконано експериментальне дослідження швидкодії транзакцій та доступу до файлів у блокчейн-системі. Визначено, що середній час запису в IPFS становить 1,8 секунди, а зчитування – 0,8 секунди, що забезпечує достатню швидкість для більшості застосувань. Запропоновано алгоритм криптографічного захисту зображень перед завантаженням у блокчейн. Впровадження AES-256 та SHA-3 дозволило забезпечити стійкість до атак на рівні 99,98% згідно з тестами криптоаналізу.

Розроблено метод перевірки автентичності зображень у блокчейні, що використовує хеш-функції та цифрові підписи. Запропонований підхід гарантує ідентифікацію походження зображень з точністю 99.7%. Досліджено витрати ресурсів та вартість транзакцій у блокчейн-системі. Визначено, що зростання розміру блокчейну на 23% призводить до збільшення вартості транзакцій на 14%, що вимагає подальшої оптимізації механізму зберігання. Розроблено механізм розподілу зображень між локальним сховищем та блокчейном з урахуванням рівня конфіденційності. Запропоновано систему класифікації, яка автоматично визначає, які файли слід зберігати у централізованому сховищі, а які – у блокчейні. Проаналізовано можливості інтеграції блокчейн-збереження у різні галузі. Досліджено застосування децентралізованого підходу у сфері медицини, цифрової криміналістики, зберігання авторських прав та захисту цифрового контенту.

У четвертому розділі монографії представлено практичну реалізацію отриманих у ході дослідження наукових результатів. Представлено використання машинного навчання для оптимізації доступу до даних в гібридному сховищі зображень. Описано практичне застосування розробленого методу оптимізованої індексації в гібридній моделі зберігання зображень для відповідного зберігання та доступу в умовах великих даних. Проаналізовано та обґрунтовано використання наявних блокчейн-рішень для захисту цифрових активів. Розглянуто структуру системи аналізу медичних зображень, наведено алгоритм роботи системи розпізнавання клітин крові. Успішно реалізовано двоетапний алгоритм аналізу медичних зображень крові – перший – сегментація, другий – класифікація клітин, що досягає точності >95% при часових витратах <1.2 с/зображення. Сформульовано основні завдання, які необхідно вирішити при проведенні морфологічного аналізу крові. Описаний двоетапний алгоритм виявлення клітин крові, а також алгоритм побудови розділової лінії на площині відносних кольорів. Наведено результати дослідів на реальних препаратах. Розглянуто причини виникнення помилок детектування. Проаналізовані практичні застосування результатів дослідження та зроблені відповідні оцінки їхньої значущості.

РОЗДІЛ 1

ОБРОБКА ТЕКСТІВ ТА ЗОБРАЖЕНЬ В УМОВАХ ВЕЛИКИХ ДАНИХ

1.1 Обробка великих даних у текстовій формі та у вигляді зображень

Сьогодні світовою спільнотою вже усвідомлений головний напрямок у боротьбі з інформаційним вибухом – це перехід від збереження й обробки даних до накопичення й обробки знань. Тому виникає потреба у засобах та методах здобуття знань з тих даних, що генеруються в процесі діяльності людства та можуть бути корисними для подальшого використання. Актуальність проблеми загострюється через стрімке поширення великих даних (Big Data), яке потребує нових, більш ефективних методів аналізу розподілених та гетерогенних даних [1]. Обробка великих обсягів неструктурованих, слабо формалізованих даних, потребує спеціалізованих засобів їх представлення та аналізу.

На сучасному етапі розвитку інформаційних технологій, як у світі, так і в Україні, актуальними є завдання, пов'язані з машинним пошуком інформації [2], особлива увага приділяється визначенню змісту текстових документів, пошуку залежностей і закономірностей. На сьогоднішній день для визначення смислу текстових документів, поданих природною мовою, існує два великих напрямки, які намагаються вирішити проблеми, пов'язані із семантичним сприйняттям тексту, а саме, побудова онто-керованих інформаційних структур [3] і визначення значення тексту за допомогою семантичного аналізу [4]. Серед них найбільш вдалим і ефективним є напрям, пов'язаний із семантичним аналізом тексту. Враховуючи це спостереження, інтерес викликають роботи [2, 5, 6], у яких проведено дослідження напрямку визначення змісту текстових документів за допомогою семантичного аналізу.

Також певної уваги потребують завдання, пов'язані з обробкою текстової інформації для вирішення ряду завдань, таких як виявлення плагіату, розпізнавання тексту, виділення структурних блоків тексту, аналіз та видача

рекомендацій тощо [7–10]. Серед усіх цих завдань однією з суттєвих, над якими працюють дослідники понад 60 років, і є «наріжним камінцем», є проблема семантичного аналізу тексту [7, 11–13]. Також слід приділити увагу семантичній коректності, підходи до перевірки якої показано, наприклад, у роботах [14–17].

Під час аналізу першоджерел робіт, присвячених семантичному аналізу, спостерігалася тенденція поділу робіт на ті, що розглядають розв'язання абстрактних теоретичних завдань, і ті, які спрямовані на полегшення роботи з комп'ютером та програмну реалізацію рішення. Слід зазначити, що всі описані в них підходи та моделі спрямовані на вирішення конкретної проблеми, а тому можуть бути застосовані лише до вузького кола предметних областей. Математичний апарат, яким користуються дослідники, також придатний лише для формалізації деяких лінгвістичних механізмів [18–21]. Визначення сенсу текстів дозволить з більшою мірою коректності розв'язувати ряд завдань обробки текстів, оскільки дозволяє вдосконалити процедуру аналізу. Наприклад, у роботі [11] проведено огляд та аналіз існуючих рішень для визначення значення текстових документів, розглянуто найбільш використовувані моделі та методи семантичної обробки тексту, а також описано класичний процес обробки тексту для семантичного аналізу. Серед інших моделей, наприклад, модель векторів абзаців дозволяє з більшою точністю вирішити задачу обробки тексту при виявленні плагіату, а також розпізнавання тексту [6].

Але на сьогодні засоби пошуку інформації повинні враховувати тенденцію все більшого переходу від текстової форми до зображень у представленні інформації та формування відповідних методів та засобів обробки та збереження інформації у вигляді зображень. Сучасні дослідження зосереджуються на розробці гібридних архітектур, що поєднують блокчейн та розподілені сховища даних для ефективного зберігання зображень. Зокрема, використання IPFS (InterPlanetary File System) для зберігання зображень, а

блокчейну – для зберігання метаданих та хешів, стає перспективним напрямком [22–24].

1.1.1 Типи сховищ зображень

У цьому підрозділі розглянуто основні типи сховищ зображень, включаючи реляційні та нереляційні бази даних, файлові системи, та гібридні моделі.

Архітектура сховища зображень – добре відома галузь досліджень із багатьма різними підходами та техніками, запропонованими протягом багатьох років. Одним із найперших підходів до архітектури сховища зображень було використання централізованої бази даних, де всі дані зображення зберігалися в одному місці. Хоча цей підхід був простим і зрозумілим, він не був дуже масштабованим і не міг обробляти достатньо великі обсяги даних зображень. У результаті дослідники почали залучати більш розподілені підходи до архітектури сховищ зображень, де дані зображень зберігаються в кількох базах даних. Цей підхід був більш масштабованим і ефективним, ніж централізована база даних, але він також створив нові проблеми, пов'язані з узгодженістю та доступністю даних [25].

Питання організації сховищ даних аналізувалося не окремо, а комплексно в контексті взаємопов'язаних проблем – фрагментації даних, обробки «великих даних» та розподіленого зберігання інформації. Наприклад, у дослідженнях Kimpel J. F. [27], Jarke M. [28], Blanco C. [29] A. Cuzzocrea, L. Bellatreche та Y. Song [30] розглядаються інтегративні підходи до цих завдань, тоді як роботи A. Boulmakoul, L. Karim, M.H. Laarabi, R. Sacile та E. Garbolino [31], Кучука Г.А. [32] акцентують увагу на архітектурних рішеннях для розподілених систем. Питання структурування сховищ на базі поширених платформ (наприклад, Hadoop) досліджували Д.В. Приймак та О.І. Акимішин [33], демонструючи переваги використання HDFS і MapReduce для масштабованих рішень. Окрему увагу приділено формалізації процесів

структурування даних, оптимізації їх обробки та побудови інтегрованих просторів даних, що знайшло відображення в працях Н.Б. Шаховської [34].

Сучасні підходи до архітектури сховища зображень зосереджені на використанні комбінації централізованих і розподілених баз даних, щоб забезпечити баланс між масштабованістю та простотою. Ці підходи довели свою ефективність для керування великими обсягами даних зображень у різних галузях промисловості. Два ключові підходи до організації сховищ даних включають корпоративну інформаційну фабрику (CIF) Білла Інмона [35, 36] та архітектуру шини даних Ральфа Кімбола [30]. У праці «The Data Warehouse Toolkit. Second Edition» Кімбол запропонував [37] методологію «знизу вгору», де проектування сховища починається з розробки вітрин даних (нижнього рівня сховища) на основі конкретних бізнес-процесів і закінчується його верхнім рівнем (операційними базами даних). Архітектура шини передбачає таку структуру: операційні бази даних – джерела сирих даних; перехідна область – проміжний шар для трансформації та очищення даних перед завантаженням у вітрини; вітрини даних – містять деталізовані та агреговані дані, організовані у вигляді зіркових схем (star schemas); інструменти доступу – забезпечують аналітичні можливості для користувачів. Ключова перевага цього підходу – конформовані виміри [38]. Архітектура шини Кімбола орієнтована на швидке отримання бізнес-результатів за рахунок модульності та гнучкості.

Існує кілька ключових проблем, пов'язаних із керуванням великими обсягами даних зображень в архітектурі сховища зображень [39]. Однією з найважливіших проблем є питання якості даних, оскільки дані зображення можуть бути складними та важкими для обробки, і вони часто піддаються помилкам і невідповідностям. Щоб вирішити цю проблему, дослідники запропонували такі методи, як очищення та нормалізація даних, які можуть допомогти покращити якість даних зображень.

Ще одним серйозним викликом є питання безпеки даних [40, 41]. Дані зображень часто є чутливими та конфіденційними, тому важливо

переконатися, що вони захищені від несанкціонованого доступу або крадіжки. Дослідники запропонували різні заходи безпеки, такі як шифрування даних, контроль доступу та безпечне зберігання, щоб гарантувати безпеку даних зображень.

Доступ до даних є ще однією критичною проблемою в архітектурі сховища зображень [42]. Важливо переконатися, що авторизовані користувачі можуть легко отримати доступ до даних зображень, які їм потрібні, забезпечивши при цьому заборону неавторизованим користувачам у можливості доступу до даних. Щоб вирішити цю проблему, дослідники запропонували такі методи, як індексування даних і алгоритми пошуку, які можуть допомогти покращити доступ до даних і їх пошук [43].

За останні роки в області обробки зображень відбувся значний прогрес завдяки поширенню великих даних і зростанню попиту на ефективні та масштабовані архітектури сховищ зображень. Було проведено низку досліджень, щодо різних методів та прийомів покращення продуктивності та безпеки сховищ зображень. Однією з важливих робіт у цій галузі є дослідження Céline F., John B., et al. [44], яке пропонує структуру для аналізу великих даних на основі Nadoop. Автори підкреслюють переваги використання Nadoop для керування великими обсягами даних зображень і демонструють ефективність своєї системи за допомогою серії експериментів. Подібним чином робота Zhao et al. [45] представляє новий підхід для пошуку зображень за допомогою глибокого навчання та згорткових нейронних мереж (CNN). Автори показують, що їх метод перевершує існуючі найсучасніші методи на кількох контрольних наборах даних.

Крім того, кілька досліджень були зосереджені на проблемі стиснення зображень і дедуплікації в сховищах зображень [46, 47]. Наприклад, робота Wang et al. [48] пропонує новий метод стиснення та дедуплікації зображень із використанням комбінації методів хешування та кластеризації. Автори показують, що їхній метод досягає кращих результатів порівняно з існуючими підходами щодо коефіцієнта стиснення та ефективності дедуплікації.

Іншим важливим напрямком досліджень є розробка методів розподіленої обробки та паралельних обчислень для сховищ зображень. Робота Tong L та ін. [49] пропонує структуру розподіленої обробки зображень на основі Apache Spark, яка забезпечує ефективну обробку великих обсягів даних зображень у розподіленому середовищі. Подібним чином робота Zhang et al. [50] пропонує паралельний обчислювальний підхід для розпізнавання зображень з використанням кластерів GPU, який досягає значних покращень у швидкості та точності обробки.

Низка досліджень присвячена питанню безпеки та конфіденційності в сховищах зображень. Робота Wang et al. [51] пропонує безпечну схему зберігання зображень з використанням методів гомоморфного шифрування. Автори демонструють ефективність свого методу за допомогою серії експериментів і показують, що він забезпечує надійні гарантії безпеки при збереженні конфіденційності даних. Дослідження в цій галузі призвели до значного прогресу в розробці та реалізації ефективних і безпечних архітектур сховищ зображень. Однак у цій галузі ще є багато можливостей для подальших досліджень, зокрема у розробці нових методів керування та аналізу великих обсягів даних зображень.

Таким чином, огляд сучасної літератури про архітектуру сховищ зображень в епоху великих даних виявив кілька перспективних методів і технологій для вирішення проблем, пов'язаних із зберіганням і обробкою великомасштабних наборів даних зображень. Проте все ще залишаються важливі відкриті питання та можливості для майбутніх досліджень, зокрема в сфері стиснення даних, розподіленої обробки, безпеки та конфіденційності. Зокрема, є необхідність дослідити використання вдосконалених алгоритмів стиснення та розподілених обчислювальних архітектур для подальшого підвищення ефективності та масштабованості систем сховищ зображень. Крім того, актуальним є вивчення нових методів захисту та конфіденційності чутливих візуальних даних у контексті великомасштабного розподіленого зберігання та обробки великих масивів інформації.

В останні роки зростає інтерес до сфери управління сховищами зображень. Були запропоновані різні підходи для підвищення ефективності та масштабованості систем зберігання зображень [52, 53]. Наприклад, Su та Huang [51] запропонували структуру для аналітики великих даних на основі Hadoop, яка дозволяє обробляти великомасштабні набори даних. Шарма та Сінгх [50] запропонували метод стиснення зображень за допомогою вейвлетів, який може значно зменшити простір для зберігання зображень.

Методи глибокого навчання також застосовувалися для управління сховищем зображень. Наприклад, Zhang et al. [54] запропонували модель глибокого навчання для класифікації зображень, яка може підвищити точність розпізнавання зображень. Подібним чином Chen et al. [55] розробили систему пошуку зображень на основі глибокого навчання, яка може ефективно отримувати зображення на основі їх вмісту. Окрім підвищення ефективності та точності керування сховищем зображень, кілька досліджень також зосереджені на забезпеченні безпеки та конфіденційності збережених зображень. Gu та ін. [56] запропонували безпечну систему зберігання та пошуку зображень із використанням криптографічних методів. Чжоу та ін. [57] розробили схему обміну зображеннями, що зберігає конфіденційність, на основі гомоморфного шифрування.

Незважаючи на прогрес, досягнутий у цій сфері, все ще існують проблеми, які необхідно вирішити. Однією з головних проблем є відсутність стандартизації систем управління сховищами зображень [58, 59]. Іншою проблемою є потреба в більш ефективних методах керування та аналізу великомасштабних наборів даних зображень [60]. Тому необхідні подальші дослідження для розробки більш ефективних і масштабованих систем управління сховищами зображень.

Таким чином, можна зазначити, що кожна технологія зберігання зображень має свій набір переваг і недоліків, відповідно й вибір тієї чи іншої технології буде визначатися потребами організації по виконанню певних завдань. Якщо, наприклад, потрібен швидкий доступ до даних зображень, але

обмежений бюджет, найкращім варіантом здається локальне сховище, так як хмарне сховище може бути хорошим вибором для організацій, які потребують масштабованого та гнучкого зберігання зображень, але можуть не мати ресурсів для керування власною інфраструктурою зберігання [49].

Архітектура сховища зображень – це складна галузь, що швидко розвивається, з багатьма різними підходами та техніками, запропонованими протягом багатьох років. Існуюча література визначила низку проблем, пов'язаних із керуванням великими обсягами даних зображень, і запропонувала різні методи та технології для вирішення цих проблем. У наступному розділі цієї роботи описані конкретні методи та матеріали, які використані для дослідження цих проблем, і запропоновані нові методи для підвищення ефективності та масштабованості архітектур сховищ зображень.

Хмарне зберігання даних, або зберігання як послуга – це абстрагована модель, яка надає адміністрування сховища через спеціальний інтерфейс, незалежно від фізичного розташування інфраструктури (локальна, віддалена або гібридна). Сучасні хмарні архітектури базуються на розподілених системах, що забезпечують обслуговування великих груп користувачів і географічно дисперсних накопичувачів, з підтримкою різних рівнів сервісу [61, 62]. Ключові особливості – економічна ефективність та масштабованість. Економічний ефект досягається завдяки оптимізації витрат. Проте ефективність залежить від моделі використання – для проектів із стабільними потребами в зберіганні витрати можуть бути вищими порівняно з традиційними рішеннями [63].

1.1.2 Реляційні та нереляційні бази даних

На сьогодні існують два типи баз даних: реляційні та нереляційні. Відмінності між ними, переваги та недоліки досліджуються в роботах [64, 65].

Реляційні бази даних засновані на використанні структурованої мови запитів SQL (Structured Query Language). Вона забезпечує високий рівень

стандартизації та гнучкість у роботі зі складними запитамі. Ця модель організовує дані у вигляді таблиць із чітко визначеними зв'язками через первинні та зовнішні ключі, що забезпечує цілісність інформації та підтримку ACID-властивостей (атомарність, узгодженість, ізольованість, довговічність). Однак жорстка схема даних вимагає попереднього проектування структури, і будь-які її зміни можуть призвести до значних ускладнень у всій системі. Нереляційні бази даних (NoSQL) пропонують альтернативний підхід із динамічною структурою. Дані можуть зберігатися у різних форматах: документо-орієнтовані (наприклад, JSON, XML), ключ-значення, графові, широко-стовпчикові. Ця гнучкість дозволяє створювати документи без попереднього визначення схеми, додавати поля під час роботи та використовувати різні синтаксичні підходи для кожної бази. Такі системи часто базуються на принципах BASE (Basically Available, Soft state, Eventually consistent), що сприяє швидкому масштабуванню, але з меншим акцентом на строгу узгодженість даних. Для додатків зі складними транзакціями та високими вимогами до цілісності даних реляційні бази залишаються оптимальним вибором. Натомість NoSQL-рішення ефективні там, де потрібна швидка адаптація до змін у структурі даних або обробка великих обсягів неструктурованої інформації.

Аналізуючи реляційні та нереляційні бази даних, J. S. van der Veen [66] стверджує, що NoSQL намагаються вирішити ряд проблем традиційних баз даних SQL. Послаблюючи вимоги до узгодженості або доступності, база даних NoSQL може працювати досить добре в умовах, коли ці обмеження не є необхідними, що зазначено, наприклад у працях Agrawal R., Ailamari A., Bernstein P.A., Brewer E. A., Weikum G. [67]. У своїй статті дослідники Li Y. та Monohara S. вказують, що не завжди є безперечною перевага у швидкодії між реляційними та нереляційними базами даних [68].

Треба зазначити, що термін NoSQL (Not Only SQL) не має загальновизнаного визначення. Він був введений Еріком Евенсом наприкінці 90-х років, проте набув свого сучасного сенсу з 2009 року. Науковці

зазначають, що поява нереляційних баз даних була викликана декількома проблемами у використанні реляційних баз даних у межах роботи веб-додатків. Наприклад, у роботі [69] визначені наступні проблеми: значна кількість високонавантажених веб-додатків обробляють такі об'єми даних, які на порядок більше тих, що можуть бути оброблені реляційними базами даних; реляційні системи управління базами даних не завжди здатні забезпечити одночасний доступ до даних мільйонам користувачів Internet; крім того, на сьогодні збільшується потреба в обробці нетабличних даних.

1.2 Методи обробки та аналізу зображень засобами глибинного навчання та комп'ютерного зору

На даний час світовим співтовариством вже усвідомлений головний напрямок у боротьбі з інформаційним вибухом – перехід від збереження й обробки даних до накопичення й обробки знань. Тому виникає потреба у засобах та методах здобуття знань з тих даних, що генеруються в процесі діяльності людства та можуть бути корисними для подальшого використання. Актуальність проблеми загострюється через стрімке поширення Big Data, яке викликає потребу в нових, більш ефективних методах аналізу розподілених та гетерогенних даних. Обробка великих обсягів інформаційних ресурсів різного походження та з наперед не відомими моделями даних (в такому випадку говорять про неструктуровані дані), для яких не придатні традиційні СКБД, потребує спеціалізованих засобів їх представлення та аналізу [1].

Неструктуровані дані – це інформація, яка не має попередньо визначеної моделі даних або та, що не організована заздалегідь. Це призводить до проблем, пов'язаних з її зберіганням (традиційні БД не розраховані на таку невизначеність) та аналізом. Саме неструктуровані дані потенційно мають найбільшу цінність як джерела нових знань: чим більше даних доступних для аналізу, тим точніші результати. Сьогодні у більшості випадків під неструктурованими даними розуміють текстову інформацію, але є й інші види,

наприклад, потокове відео, інформація від супутників, дані радарів чи сонарів. Засоби аналізу таких неструктурованих даних значно більш спеціалізовані. Як визначають дослідники у роботах [70, 71], на сьогоднішній день неструктуровані дані складають найбільшу частку даних, що зберігаються (понад 80 % усіх збережених даних, а їхня кількість зростає на порядок швидше в порівнянні із структурованими даними), тому методи та засоби їх використання швидко розвиваються [1].

Області застосування цифрових методів обробки сигналів та зображень значно розширилися, витісняючи аналогові підходи завдяки своїй ефективності та гнучкості. Методи цифрової обробки даних широко застосовуються у сфері керування процесами, автоматизації виявлення об'єктів, розпізнавання образів та інших галузях, що потребують ефективної передачі та обробки зростаючих обсягів інформації.

До класичних методів цифрової обробки зображень зазвичай відносять математичні та ймовірнісні моделі зображень, статистичні підходи до оцінки їхньої якості (зокрема, кореляційний аналіз), а також широкий спектр алгоритмів фільтрації – як лінійної, так і нелінійної. Окрім цього, до основних методів належать підходи, засновані на динамічних моделях, декомпозиції на власні вектори, методи класифікації елементів зображень, а також методи визначення контурів та сегментації. У той час як штучно створені об'єкти, зокрема промислові та житлові будівлі, можуть бути ефективно описані за допомогою простих геометричних примітивів, таких як куби, сфери, циліндри та конуси, природні текстури через свою нерегулярність і фрагментарну структуру значно гірше піддаються такому формальному опису. У зв'язку із цим, для аналізу таких текстур виявляється природним подання їх у вигляді фрактала з деяким розміром D [72].

Вейвлет-перетворення (wavelet transformation) – це сучасний і перспективний метод обробки даних. Англійське слово wavelet (від французького «ondelette») дослівно перекладається як «коротка (маленька) хвиля». Апарат вейвлет-аналізу одержав свій розвиток на початку 1980-х років

у роботах Морле, Гроссмана й деяких інших авторів [73]. Успішне застосування вейвлет-аналізу в різних наукових сферах значно посилило інтерес до цього напрямку та сприяло його подальшому розвитку. Вейвлет-аналіз є специфічним видом лінійного перетворення функцій, що належать до широкого класу. Базис власних функцій, за яким здійснюється розкладання, має унікальні властивості, правильне використання яких дозволяє досліднику зосередити увагу на ключових особливостях аналізованого сигналу.

Для розпізнавання складних об'єктів створюють системи на основі нейронних мереж (НМ, neural network). Вони можуть мати топологію, орієнтовану на розв'язання конкретної задачі із врахуванням властивостей об'єкта – просторово-часову орієнтацію, масштаб, геометричні параметри об'єкта, включаючи координати, кутове положення, лінійний розмір, відстань, тощо. У той же час істотним недоліком типових НМ є відсутність ефективних засобів для розв'язання задач розпізнавання динамічних образів. Основною проблемою інтерпретації динамічних візуальних сцен є висока розмірність простору ознак, наявність геометричних перетворень над об'єктом. Стиск простору ознак виконують методом витягу інтегральних і інваріантних до геометричних перетворень параметрів зображень. Метод геометричних та більш загальних алгебраїчних інваріантів відіграє значну роль у розв'язанні задач розпізнавання зображень. Так, наприклад, інваріанти були успішно використані для розпізнавання профілів літаків і танків, друкованих і рукописних букв, параметрів стикувального вузла космічного апарата, а також багатьох інших об'єктів. Математичне обґрунтування інваріантних особливостей напівтонових зображень базується на теорії алгебраїчних інваріантів [72].

Глибокі нейронні мережі застосовуються для вирішення задач класифікації зображень. Комп'ютерний зір – це теорія та набір технологій, які дозволяють машинам візуально сприймати реальність. Комп'ютерний зір включає методи виявлення, відстеження, класифікації та ідентифікації об'єктів шляхом аналізу зображень та відеоданих. Він використовується для

розпізнавання об'єктів, аналізу відео, опису вмісту зображень/відео, розпізнавання жестів і рукопису, інтелектуальної обробки зображень [42, 72].

Глибоке навчання – це розвиток класичних нейронних мереж, який дозволяє аналізувати складні шаблони та ієрархії даних завдяки багаторівневій структурі. Такі мережі складаються з великої кількості шарів, що поступово виділяють все абстрактніші ознаки з вхідних даних, від простих контурів до складних об'єктів чи понять. Їхня сила – у здатності самостійно вчитися на великих обсягах інформації. Цей підхід відкрив нові можливості в задачах, де традиційні методи зазнавали невдач – від розпізнавання обличчя на фото до побудови систем, які розуміють людську мову [74]. Головна перевага цього підходу – висока точність і адаптивність. Технології на базі глибокого навчання вже змінюють медицину, фінанси, транспорт, демонструючи, як машинний інтелект може доповнювати людський.

Інтеграція методів машинного навчання, особливо глибоких нейронних мереж, з технологіями блокчейн відкривають нові можливості для аналізу зображень. Наприклад, використання згорткових нейронних мереж (CNN) для класифікації та розпізнавання об'єктів на зображеннях, що зберігаються у блокчейн-системах, значно підвищує ефективність обробки візуальних даних [75, 76].

Цифрова обробка зображень стала невід'ємною частиною різних сфер діяльності людини. Вона використовується в медицині для аналізу діагностичних зображень, у дистанційному зондуванні для моніторингу стану навколишнього середовища, а також у фінансовому секторі для розпізнавання валюти. Такий широкий спектр застосувань демонструє її здатність трансформувати роботу з візуальними даними. Зображення та відео містять значні обсяги даних, оскільки складаються з великої кількості пікселів. Відповідно, їх обробка потребує використання методів «Big Data», що забезпечують швидку обробку та аналіз цих великих зображень і відео. Оцифровані зображення проходять процес аналізу та обробки, спрямований

на вдосконалення ключового етапу обробки зображень, відомого як «сегментація зображення» [77].

1.3 Аналіз досліджень блокчейн технологій для зберігання та захисту зображень

Слід одразу відзначити, що на сьогоднішній день існує багато доступних систем стиснення зображень, які відрізняються за якістю в залежності від алгоритму стиснення. Якість стиснення прямо пропорційна якості алгоритму, який, у свою чергу, залежить від розміру, типу та обсягу даних, а також від типу системи, в якій він застосовується. Новітні дослідження пропонують інноваційні підходи до зберігання зображень у блокчейні. Зокрема, концепція «imagechain» представляє криптографічну структуру, яка пов'язує цифрові зображення за допомогою хеш-посилань. На відміну від традиційного блокчейну, зображення не зберігаються всередині блоків, а блок та зображення об'єднуються в процесі вбудовування [78, 79].

Розглянемо існуючі технології збереження зображень у блокчейн сховищах. Зображення, які використовуються в сучасних інформаційних системах, зазвичай мають великий обсяг даних і потребують ефективних методів зберігання та передачі. Блокчейн сховища активно досліджуються для зберігання зображень, оскільки вони пропонують децентралізовану, безпечну та прозору інфраструктуру для зберігання даних [80].

З плином часу інформаційно-комунікаційні технології зазнали численних трансформацій для полегшеного, швидшого, ефективнішого та безпечнішого обміну даними, інформацією та коштами різноманітними способами. З появою Інтернету з'явилися цифрові комунікації, які розширили можливості для всіх форм обміну даними та інформацією через онлайн-транзакції. Усі транзакційні та комунікаційні процеси проходять через надійного посередника, який гарантує безпечну доставку, забезпечує точне відображення інформації. Але, фахівці швидко дійшли думки, що ця довірена

сторона може викликати сумніви у разі будь-яких збоїв в оновленні даних, затримок доставки або шахрайства [81]. Тому виникає одразу декілька запитань. По-перше, що робити, якщо ця довірена сторона стане шахраєм і їй не можна буде довіряти? По-друге, що, якщо його буде зламано і зловмисник заволодіє всіма даними? Цей посередник тут діє як єдина точка відмови. І, нарешті, кожного разу, коли використовується посередник, виникає додаткова затримка в спілкуванні. Чому б не спілкуватися безпосередньо? [82].

Рішення всіх вищезазначених проблем може забезпечити блокчейн – базова технологія, винайдена Сатоші Накамото (вважається псевдонімом) під час впровадження першої в історії децентралізованої криптовалюти під назвою «Біткоїн», що досліджено в роботах [83–85]. Обмін і передача біткоїнів відбуваються за допомогою спільної розподіленої книги, яка записує деталі кожної транзакції, що відбулася між учасниками мережі, без залучення довіреної централізованої сторони.

Blockchain – ланцюжок блоків даних, де кожен блок пов'язаний із попереднім. Блок містить у собі набір записів, а нові блоки завжди додаються строго в кінець ланцюжка. Сутність такої технології досліджено у працях [86–89]. Система побудована на трьох принципах – розподіленість, відкритість та захищеність. Блокчейн працює завдяки мережі комп'ютерів (вузлів), де кожен зберігає копію даних. Навіть якщо частину вузлів вимкнути, система продовжує функціонувати. Це робить блокчейн практично незламним. Нові користувачі підключаються до мережі, автоматично збільшуючи її розмір і надійність. Усі вузли рівноправні – тут немає керівників чи цензорів. Кожен комп'ютер самостійно перевіряє дані, підтримуючи цілісність системи через спеціальні алгоритми згоди [90]. Навіть якщо залишиться лише один активний вузол, блокчейн продовжить існувати. І відновиться, коли до нього приєднаються інші. Причому всі вузли рівноправні, там немає організаторів, модераторів, контролерів і менеджерів. Кожен відповідає за себе сам. Усі дані блоки у блокчейні і їх вміст є завжди відкритими та для усіх доступними. Можна легко прочитати будь-який блок і побачити всі записи в ньому. Також

можна подивитися повний ланцюжок і відстежити зміну інформації. Таким чином, усі дані в blockchain легко перевіряються, з цього випливає, що не обов'язково довіряти іншим учасникам мережі, адже самому завжди можна їх перевірити і отримати гарантовано достовірну відповідь.

Один із найбільш перспективних напрямків – технологія Smart-контрактів [91]. Сьогодні можна передбачити технологічний прорив у логістичній галузі шляхом створення прозорої системи взаємодії між усіма учасниками ринку. Децентралізована логістична платформа може запропонувати інноваційні рішення, які засновані на впровадженні технології blockchain в управлінні ланцюжками поставок. Це децентралізована система, яка використовує blockchain і складається з декількох інтелектуальних контрактів всередині блочного ланцюга і його власних токенів (якщо необхідно) [92, с. 214].

Корисним застосування Smart-контрактів буде в процесах логістики. Ланцюжок поставок зазвичай складається з багатьох ланок, проте кожна ланка має отримати підтвердження від попередньої, виконати свою частину контракту і передати інформацію далі. У традиційній практиці це забирає багато часу і знижує ефективність, а використання Smart-контрактів економить час і дозволяє відстежувати прогрес у реальному часі. Одне з головних завдань Smart-контрактів – скоротити витрати на логістику, так щоб її частка в собівартості товару була мінімальна [92].

Розглянемо, які питання виникають найчастіше. По-перше, це проблема довіри. Контроль за перевезенням вантажів здійснюється, за запитом, до успішного завершення угоди. Всі дії фіксуються в blockchain, що виключає недовіру між сторонами; Smart-контракт, який буде затверджений на початку відвантаження, автоматично виконає взаємний розрахунок відповідно до даних, що зберігаються в blockchain [91].

Блокчейн – це система з відкритим доступом до всіх даних: будь-який користувач може переглянути вміст блоків, відстежити історію змін і перевірити достовірність інформації самостійно. Це усуває необхідність

довіряти іншим учасникам, адже кожен має змогу пересвідчитися в правильності даних через криптографічні механізми. Ключовий принцип – поєднання прозорості і захищеності. Наприклад, хеш-функції шифрують дані так, що за кінцевим ключем неможливо відновити вихідну інформацію або підібрати інший набір даних з таким самим хешем. Це робить блокчейн стійким до підробок: навіть якщо хтось намагатиметься змінити дані в блоці, це порушить ланцюжок хешів і стане очевидним для всієї мережі [93]. Таким чином, криптографія забезпечує надійність системи, а децентралізація гарантує, що жоден учасник не може одноосібно вплинути на її роботу. Навіть при наявності зловмисників, мережа залишається захищеною через механізми консенсусу та математичну незворотність хешування.

Таким чином, сьогодні блокчейн – це децентралізована база даних, пристрої зберігання якої не підключені до загального сервера. Будь-який учасник системи може робити в ній записи, які після перевірки автоматично відображаються на всі комп'ютери в мережі. Головна перевага блокчейну – безпека інформації, яка зберігається в ньому [40, 94–96]. Ніхто не може підробити або замінити його, оскільки знадобляться односторонні зміни на кожному з мережевих пристроїв, величезні обчислювальні можливості, недоступні для звичайних учасників системи. Блокчейн – це децентралізована система зберігання даних, де інформація розподілена між усіма учасниками мережі. Вона не належить жодній організації чи людині, а функціонує за рахунок спільної роботи комп'ютерів користувачів. Кожен з них зберігає копію бази даних, що робить систему стійкою до зламів. Ключова ідея – відсутність центрального контролю. Навіть якщо частина комп'ютерів вийде з ладу або буде зламана, інші продовжать підтримувати роботу мережі. Саме тому блокчейн часто називають «незламним» – його існування залежить лише від наявності хоч одного активного учасника. Ця технологія дозволяє створювати прозорі системи, де дані можуть перевірятися будь-яким користувачем, але при цьому залишаються захищеними завдяки криптографії.

Підсумовуючи, можна сказати, що блокчейн поєднує в собі довіру через математику, а не через участь третіх сторін. Він може застосовуватися в різних сферах, пропонуючи компаніям різні переваги та інноваційні бізнес-моделі. Наприклад, в системі захисту авторських прав. Переваги нової технології полягають у тому, що можна передати авторство без участі третьої сторони та без прив'язки до географії [97]. Основними викликами у інтеграції технологій блокчейн та обробки зображень залишаються оптимізація зберігання великих обсягів даних та забезпечення конфіденційності при використанні публічних блокчейнів. Перспективними напрямками досліджень є розробка енергоефективних алгоритмів консенсусу та створення децентралізованих платформ для обміну та монетизації зображень [94, 95].

Таким чином, використання блокчейн технологій може бути цілком корисним при дослідженні та розв'язанні задач, пов'язаних із зберіганням і обробкою великомасштабних наборів даних у вигляді зображень.

1.4 Існуючі проблеми у сфері архітектури сховищ зображень в епоху великих даних

У цьому підрозділі представлено детальний огляд останніх розробок у сфері архітектури сховищ зображень в епоху великих даних. Метою цього підрозділу є надання всебічного огляду існуючої літератури та визначення ключових тенденцій, викликів і можливостей у цій галузі.

Архітектура сховища зображень – добре налагоджена галузь досліджень із багатьма різними підходами та техніками, запропонованими протягом багатьох років. Одним із перших підходів до архітектури сховища зображень було використання централізованої бази даних, де всі дані зображення зберігалися в одному місці. Хоча цей підхід був простим і зрозумілим, він не був дуже масштабованим і не міг обробляти великі обсяги даних зображень.

Як наслідок, дослідники почали вивчати більш розподілені підходи до архітектури сховищ зображень, де дані зображень зберігалися у кількох базах

даних. Цей підхід виявився більш масштабованим та ефективним порівняно з централізованою моделлю зберігання, проте він також створив нові виклики, пов'язані із узгодженістю даних та забезпеченням їхньої доступності [25]. Новітні підходи до архітектури сховищ зображень зосереджені на використанні комбінації централізованих та розподілених баз даних для забезпечення балансу між масштабованістю та простотою управління. Ці підходи довели свою ефективність для управління великими обсягами даних зображень у різних галузях.

Існує кілька ключових проблем, пов'язаних з управлінням великими обсягами даних зображень в архітектурі сховища зображень. Однією з найсуттєвіших проблем є питання забезпечення якості зображень, оскільки ці дані можуть бути складними для обробки та часто містять помилки й невідповідності. Для вирішення цієї проблеми дослідники запропонували такі методи, як очищення та нормалізація даних, які можуть покращити якість даних зображень, підвищити точність й надійність отриманих результатів..

Іншою важливою проблемою є питання безпеки даних. Дані зображень часто є конфіденційними, тому важливо забезпечити їх захист від несанкціонованого доступу чи викрадення. Дослідники запропонували різні заходи безпеки, такі як шифрування даних, контроль доступу та безпечне зберігання, щоб гарантувати збереження та захист даних зображень.

Доступ до даних є ще одним критичним питанням в архітектурі сховища зображень. Важливо забезпечити, щоб авторизовані користувачі могли швидко та ефективно отримувати доступ до потрібних їм зображень, водночас обмежуючи можливість доступу сторонніх осіб. Для вирішення цієї проблеми було запропоновано механізми індексації даних та алгоритми пошуку, які суттєво покращують швидкість і точність процесу вибірки [43].

Останніми роками галузь обробки зображень зазнала значного прогресу завдяки активному розвитку технологій великих даних та зростаючому попиту на ефективні та масштабовані архітектури сховищ зображень. У межах цього напрямку було проведено низку досліджень для вивчення різних методів та

технік підвищення продуктивності систем зберігання та обробки зображень, а також на удосконалення механізмів їхнього захисту.

Однією з важливих робіт у цій галузі є дослідження Су та Хуанга [44], які запропонували фреймворк для аналітики великих даних на основі Hadoop. Автори підкреслюють переваги використання Hadoop для управління великими обсягами зображень та демонструють ефективність своєї методики шляхом проведення серії експериментів. Аналогічно, робота Чжао та ін. [45] представляє новий підхід до пошуку зображень з використанням глибокого навчання та згорткових нейронних мереж (CNNs). Отримані результати демонструють перевагу запропонованого алгоритму над сучасними методами на низці тестових наборів.

Окрему увагу у науковій літературі приділено проблемам стиснення та дедуплікації зображень у межах сховищ. Наприклад, дослідження Ванга та ін. [48] презентує новий метод стиснення і дедуплікації, що базується на комбінації хеш-функцій і методів кластеризації. Результати роботи доводять, що запропонований підхід досягає кращих результатів порівняно з існуючими підходами з точки зору коефіцієнта стиснення та ефективності дедуплікації.

Іншою важливою сферою досліджень є розробка методів розподіленої та паралельної обробки даних у сховищах зображень. Робота Лі та ін. [68] пропонує архітектуру розподіленої обробки зображень на основі Apache Spark, що забезпечує ефективну паралельну обробку великих обсягів зображень у кластерних середовищах. Аналогічно, робота Чжана та ін. [50] описує підхід до розпізнавання зображень на основі GPU-кластерів, що забезпечує суттєве підвищення продуктивності та точності обробки.

У контексті забезпечення безпеки та конфіденційності збережених зображень значну увагу приділено розробці захищених систем зберігання. Робота Ванга та ін. [50] пропонує схему безпечного зберігання зображень з використанням гомоморфного шифрування та методів обфускації. Проведені експериментальні дослідження підтвердили ефективність запропонованого підходу у забезпеченні безпеки даних.

Дослідження в цій галузі привели до значних досягнень у розробці масштабованої та безпечної архітектури сховищ зображень. Однак все ще залишаються відкритими наукові проблеми, що стосуються оптимізації процесів зберігання, розподіленої обробки та забезпечення конфіденційності даних.

Аналіз сучасних досліджень у сфері архітектури сховищ зображень у контексті великих даних виявив низку перспективних підходів, які спрямовані на подолання викликів, що пов'язані із збереженням та обробкою масштабних наборів візуальних даних. Однак все ще залишаються відкритими питання та можливості для майбутніх досліджень, особливо в галузях стиснення даних, розподіленої обробки, а також розробки нових методів забезпечення безпеки та конфіденційності. З огляду на це, у подальших дослідженнях варто зосередитися на використанні передових алгоритмів стиснення та технологій розподілених обчислень, що дасть змогу покращити ефективність і масштабованість сховищ зображень. Крім того, перспективним напрямом є розробка новітніх механізмів захисту конфіденційних зображень, які зможуть забезпечити надійне зберігання та обробку великих візуальних масивів у розподілених середовищах.

Останніми роками спостерігається зростаючий інтерес до проблематики управління сховищами зображень. Було запропоновано різні підходи для підвищення ефективності та масштабованості систем сховищ зображень. Наприклад, Су та Хуанг [51] запропонували фреймворк для аналітики великих даних на основі Hadoop, який дозволяє обробляти великомасштабні набори даних. Шарма та Сінгх [50] запропонували метод стиснення зображень з використанням вейвлетів, який може значно зменшити обсяг пам'яті, необхідний для зберігання зображень. Методи глибокого навчання також були застосовані до управління сховищами зображень. Наприклад, Чжан та ін. [54] розробили глибоку нейронну мережу для класифікації зображень, що дозволило підвищити точність розпізнавання. Аналогічно, Чен та ін. [50]

розробили систему пошуку зображень на основі глибокого навчання, яка забезпечує ефективний пошук за вмістом.

Окрім оптимізації управління сховищами зображень, дослідження також зосереджені на забезпеченні безпеки та конфіденційності збережених зображень. Лю та ін. [56] запропонували безпечну систему зберігання та пошуку зображень з використанням криптографічних методів. Чжоу та ін. [57] розробили схему обміну зображеннями зі збереженням конфіденційності на основі гомоморфного шифрування. Незважаючи на прогрес, досягнутий у цій сфері, все ще існують проблеми, які необхідно вирішити. Однією з головних проблем є відсутність стандартизації систем управління сховищем зображень. Іншою проблемою є потреба в більш ефективних методах керування та аналізу великомасштабних наборів даних зображень. Тому необхідні подальші дослідження для розробки більш ефективних і масштабованих систем управління сховищами зображень.

Як видно з Табл. 1.1, кожна технологія зберігання зображень має свій набір переваг і недоліків, і вибір технології залежить від конкретних потреб організації.

Таблиця 1.1 – Порівняння різних технологій зберігання зображень

Технологія	Переваги	Недоліки
Локальне сховище	Швидкий доступ, низька затримка	Обмежена ємність зберігання
Мережеве сховище (NAS)	Централізоване управління, масштабованість	Обмежена продуктивність
Мережа зберігання даних (SAN)	Висока продуктивність, масштабованість	Складне управління, висока вартість
Хмарне сховище	Гнучкість, масштабованість, доступність з будь-якого місця	Потенційні ризики безпеки та конфіденційності, залежність від підключення до інтернету

Локальне сховище є хорошим варіантом для малих підприємств, яким потрібен швидкий доступ до даних зображень, але, можливо, немає бюджету на дорожчі рішення. З іншого боку, хмарне сховище може бути хорошим вибором для організацій, які потребують масштабованого та гнучкого зберігання зображень, але можуть не мати ресурсів для керування власною інфраструктурою зберігання [49]. Архітектура сховища зображень – це складна галузь, що швидко розвивається, з багатьма різними підходами та техніками, які були запропоновані протягом багатьох років. Існуюча література визначила низку проблем, пов'язаних з керуванням великими обсягами даних зображень, і запропонувала різні методи та технології для вирішення цих проблем. Далі у монографії описані конкретні методи та матеріали, які використані для дослідження цих проблем, і запропоновані нові методи для підвищення ефективності та масштабованості архітектури сховищ зображень.

Архітектура сховища зображень є складною та динамічною сферою, що активно розвивається. За останні роки було запропоновано широкий спектр підходів і технологій, спрямованих на оптимізацію зберігання, управління та обробки візуальних даних. Існуюча література визначила низку проблем, пов'язаних із керуванням великими обсягами даних зображень, і висвітлила різні методи та технології для вирішення цих проблем. У цьому дослідженні запропоновані нові методи для підвищення ефективності та масштабованості архітектури сховищ зображень.

Аналіз сучасної літератури дозволив сформулювати основні проблеми, мету та задачі дослідження. Постановка проблеми та формулювання мети та завдань дослідження витікають із класифікації проблем, які описані нижче.

Першою проблемою є відсутність ефективного та безпечного рішення для зберігання та обробки великих обсягів зображень, що виникає в умовах зростання кількості цифрових даних та їх значення для різних галузей людської діяльності.

Другою проблемою є недостатня ефективність та продуктивність традиційних методів зберігання та обробки зображень, що призводить до затримок у роботі та незадовільної якості результату.

Третьою проблемою є відсутність надійного та ефективного методу захисту зображень від несанкціонованого доступу, що може призвести до порушення конфіденційності та безпеки даних.

Четвертою проблемою є недостатня вивченість гібридних моделей зберігання та обробки зображень, які можуть поєднувати різні типи сховищ та методи обробки зображень для досягнення кращих результатів.

Загалом, аналіз літературних джерел та огляд існуючих проблем виявив, що блокчейн має значний потенціал для трансформації способів зберігання та захисту зображень, надаючи нові можливості для підвищення безпеки, забезпечення прозорості та оптимізації процесів. Однак, для повноцінного використання цього потенціалу необхідно подолати ряд технічних та організаційних викликів. Подальші інновації в алгоритмах блокчейну, розробка нових підходів до обробки даних та впровадження квантово-стійких криптографічних методів будуть ключовими для подолання цих викликів.

З розробкою моделі блокчейн-системи для зберігання зображень виникають перспективи її застосування в широкому спектрі сфер, де важливими є безпека, прозорість та надійність збереження інформації. Від медичних установ, де забезпечення конфіденційності даних пацієнтів є критично важливим, до банківського сектору, що вимагає безперервності та незмінності фінансових записів, модель може пропонувати значні переваги.

Враховуючи швидкий розвиток цифрових технологій, важливим є розгляд можливостей адаптації блокчейн-систем для підтримки новітніх форматів зображень та відео, які можуть з'явитися у майбутньому. У зв'язку з глобалізацією та необхідністю міжнародного обміну даними, інші аспекти дослідження можуть бути спрямовані на розробку глобальних регуляторних та оперативних стандартів для блокчейн-систем, що дозволить спростити взаємодію та обмін даними між країнами та різними юрисдикціями. Нарешті,

потрібно враховувати потребу в балансі між інноваціями та захистом особистих даних, оскільки великий потенціал блокчейну для збільшення прозорості та відстеження використання даних не повинен порушувати права індивідів на конфіденційність. Подальші дослідження у цих напрямках відкривають нові можливості для використання блокчейну, покращуючи технології зберігання зображень та вносячи свій вклад у цифрову трансформацію різних сфер життя суспільства.

Таким чином, у монографії описано дослідження, метою якого є підвищення ефективності моделей сховищ даних за рахунок розробки формальної процедури організації сховищ зображень на основі гібридної моделі. Для досягнення цієї мети вирішено та описано такі завдання.

По-перше, проведено комплексний аналіз сучасних підходів до архітектури сховищ зображень в умовах великих даних, виявлено ключові проблеми та перспективні напрями досліджень у галузі ефективного зберігання та обробки великомасштабних наборів візуальних даних. Визначено та проаналізовано основні методи та технології стиснення зображень для їх ефективного зберігання у гібридних та блокчейн-орієнтованих сховищах. Проведено порівняльний аналіз алгоритмів компресії та їх застосування у децентралізованих сховищах.

По-друге, запропоновано гібридну модель сховища зображень, яка поєднує реляційні та нереляційні бази даних, файлові системи та блокчейн технології для зберігання та захисту зображень. Розроблено методи обробки зображень, які дозволяють ефективно та швидко обробляти великі обсяги зображень та забезпечувати високу якість результатів. Запропоновано методи захисту зображень від несанкціонованого доступу, що включають методи шифрування, контролю доступу та автентифікації користувачів.

Нарешті, описані проведені експериментальні дослідження розробленої гібридної моделі та методів обробки та захисту зображень і визначено їх ефективність у порівнянні з іншими методами. *Оцінено практичну та наукову значимість розробленої гібридної моделі та методів, визначено їх можливість*

впровадження в різні сфери діяльності, які вимагають зберігання та обробки великих обсягів зображень.

У другому розділі монографії розглянуто проектування та розробку гібридної моделі сховища зображень, зокрема опис архітектури та дизайну інтерфейсу користувача. Описані методи аналізу великих даних для архітектури сховищ зображень, зроблений огляд існуючих проблем у наведеній області. Звернено увагу на вибір методів зберігання та обробки зображень для реалізації гібридної моделі. Особлива увага приділена керуванню великими обсягами даних зображень в архітектурі сховища зображень та підтримці безпеки та конфіденційності даних зображень.

РОЗДІЛ 2

ГІБРИДНА МОДЕЛЬ СХОВИЩА ЗОБРАЖЕНЬ

2.1 Методи аналізу великих даних для розробки архітектури сховищ зображень

Основна мета цього підрозділу полягає в тому, щоб запропонувати новий підхід до підвищення ефективності та масштабованості архітектури сховища зображень в епоху великих даних шляхом поєднання попередньої обробки даних, стиснення та дедуплікації, розподіленої обробки та паралельних обчислень, машинного та глибокого навчання та техніки безпеки і конфіденційності. Зокрема, пропонується розробити та впровадити систему, яка може обробляти величезну кількість зображень, зменшити витрати на зберігання та обробку, підвищити швидкість пошуку та забезпечити конфіденційність і безпеку даних.

Стрімка діджиталізація різних сфер людської діяльності зумовлює необхідність розробки ефективних методів та засобів обробки великомасштабних наборів графічних даних. У даному розділі монографії пропонується архітектура сховища зображень в епоху великих даних, яка поєднує попередню обробку, стиснення і дедуплікацію даних, розподілену обробку і паралельні обчислення, машинне навчання і глибоке навчання, а також методи безпеки і конфіденційності для підвищення ефективності і масштабованості сховища зображень.

Проведено комплексне експериментальне дослідження на масштабному наборі візуальних даних. Результати переконливо демонструють, що розроблений підхід суттєво перевершує існуючі методи за ключовими показниками – точністю пошуку та обчислювальною ефективністю.

З метою обґрунтування доцільності вибору гібридної архітектури було проведено аналітичне порівняння основних типів сховищ, що використовуються для зберігання зображень: SQL, NoSQL, IPFS та їхнього

гібридного поєднання. Оцінювання проводилось за чотирма ключовими критеріями ефективності – час доступу до зображень, надійність зберігання, масштабованість та підтримка децентралізації (табл. 2.1).

Таблиця 2.1 – Порівняння ефективності систем зберігання

Критерій	SQL	NoSQL	IPFS	Гібридна модель (SQL+NoSQL+IPFS)
Час доступу до зображень	120–150 мс	80–100 мс	180–250 мс (залежить від мережі)	60 мс (оптимізовано)
Надійність зберігання	Висока (ACID)	Висока (залежить від БД)	Середня (потребує дублювання)	Висока (розподілене + локальне)
Масштабованість	Обмежена (вертикальна)	Висока (горизонтальна)	Висока (розподілена)	Дуже висока
Підтримка децентралізації	Немає	Обмежена	Повна	Повна (через IPFS)

Зіставлення характеристик демонструє, що гібридна модель поєднує сильні сторони всіх трьох систем. Вона забезпечує мінімальний час доступу до даних, високу надійність завдяки дублюванню в різних базах даних, чудову масштабованість через горизонтальне масштабування NoSQL та розподілену природу IPFS, а також підтримку децентралізації. Таким чином, така комбінація є перспективною для систем, що мають справу з великими обсягами візуальних даних і високими вимогами до продуктивності.

Для обґрунтування вибору гібридної архітектури було здійснено емпіричне порівняння продуктивності трьох підходів до організації сховища зображень: реляційна СУБД (SQL), нереляційна NoSQL (MongoDB) та децентралізована IPFS. Також окремо оцінено гібридну модель, яка поєднує компоненти зазначених архітектур.

Тестування проводилось на наборах зображень у кількостях 1000 та 10000 одиниць, з використанням ідентичного обладнання та запитів. Замірялись три основні показники – час вставки зображення у сховище (Insert), час доступу до одного зображення по ID (Select), час пошуку за тегом або метаданими (Search). Результати вимірювань наведені у таблиці 2.2.

Таблиця 2.2 – Порівняльні характеристики продуктивності для різних типів сховищ зображень

Кількість зображень	Система	Час вставки (мс)	Час доступу (мс)	Час пошуку за тегом (мс)	Коефіцієнт стиснення	Час дешифрування (мс)	Використання пам'яті (%)
1 000	SQL	450	120	180	1.0:1 (без стиснення)	40 (AES-256)	18
	NoSQL	280	85	95	1.5:1 (JPEG)	35 (AES-128)	15
	IPFS	900	210	–	2.1:1 (WebP)	25 (AES-256)	12
	Гібридна	300	60	75	2.5:1 (WebP + дедуплікація)	20 (AES-256 + блокчейн)	10
10 000	SQL	4700	150	210	1.0:1	45	20
	NoSQL	3200	110	130	1.5:1	40	18
	IPFS	9500	250	–	2.1:1	30	15
	Гібридна	3300	90	110	2.5:1	25	12

Як видно з таблиці, гібридна модель забезпечує найкращі показники швидкості як доступу, так і пошуку при збереженні прийнятної часу вставки. Це обумовлено використанням оптимізованої індексації, попередньої обробки та децентралізованого зберігання. Такі емпіричні вимірювання демонструють ефективність запропонованої архітектури та можуть бути використані для подальшого калібрування параметрів масштабування системи. Запропонована архітектура забезпечує інноваційне рішення для ефективного управління та швидкого доступу до великих обсягів зображень в умовах стрімкого зростання даних. Отримані результати підтверджують перспективність розробленого методу для практичного застосування в системах обробки візуальної інформації.

В епоху великих даних архітектура сховищ зображень стала ще більш важливою. Експоненціальне зростання даних зображень у поєднанні з досягненнями в аналізі та обробці зображень створило потребу в більш ефективних і масштабованих архітектурах сховищ зображень. Однією з ключових проблем, пов'язаних з управлінням великими обсягами даних зображень, є питання якості даних. Дані зображення можуть бути складними і важкими для обробки, і вони часто схильні до помилок і невідповідностей. Для

вирішення цієї проблеми архітектури сховища зображень можуть включати такі методи, як очищення даних і нормалізація даних, що може допомогти поліпшити якість даних зображення.

Ще одним серйозним викликом є питання безпеки даних. Дані зображень часто є чутливими та конфіденційними, і важливо забезпечити їх захист від несанкціонованого доступу або крадіжки. Архітектури сховищ зображень можуть включати заходи безпеки, такі як шифрування даних, контроль доступу та безпечне зберігання. Інтеграція зазначених захисних механізмів у архітектуру сховища є невід'ємною складовою забезпечення належного рівня захищеності візуальних активів [44].

Доступ до даних є ще однією важливою проблемою в архітектурі сховища зображень. Важливо гарантувати, що авторизовані користувачі можуть легко вилучити та отримати доступ до потрібних даних зображення, гарантуючи, що неавторизовані користувачі не зможуть отримати доступ до даних. Для вирішення цієї проблеми архітектури сховищ зображень можуть включати такі методи, як індексація даних та алгоритми пошуку, які можуть допомогти покращити доступ до даних та пошук.

На додаток до цих проблем, існує також кілька можливостей для інновацій в іміджевій складській архітектурі. Одним з найбільш перспективних напрямків інновацій є використання методів розподіленої обробки і паралельних обчислень, які можуть допомогти підвищити ефективність і масштабованість архітектури сховищ зображень. Ще одним напрямком інновацій є використання гібридної архітектури зберігання даних, яка може забезпечити баланс між економічною ефективністю та продуктивністю.

У контексті стрімкого зростання обсягів візуальної інформації, дослідження архітектури сховищ зображень набуває особливої актуальності. Розробка ефективних структур зберігання графічних даних дозволяє створити масштабовані та продуктивні системи управління великими масивами візуального контенту. Такі інноваційні рішення мають суттєвий вплив на

галузі, де обробка та аналіз зображень відіграють ключову роль. Оптимізація архітектури сховищ не лише підвищує ефективність роботи з даними, але й відкриває нові можливості для розвитку технологій комп'ютерного зору, медичної діагностики, геоінформаційних систем та інших сфер, що активно використовують аналіз візуальної інформації.

Сховище зображень – це база даних, призначена для зберігання та керування великими обсягами даних зображень. Основною метою сховища зображень є забезпечення ефективного та масштабованого способу зберігання та керування зображеннями, із гарантією, що авторизовані користувачі можуть їх легко отримати та отримати до них доступ. Сховища зображень зазвичай використовуються в галузях, які генерують великі обсяги даних зображень, таких як охорона здоров'я, медіа та відеоспостереження.

Дизайн сховища зображень базується на кількох ключових принципах, включаючи масштабованість, ефективність і безпеку. Добре спроектоване сховище зображень повинно мати можливість обробляти великі обсяги даних зображень, забезпечуючи при цьому швидкий і надійний доступ до даних. Крім того, сховище зображень має бути безпечним, тобто слід задіювати механізми, які гарантують, що дані не будуть втрачені, пошкоджені чи доступні неавторизованим користувачам [45].

Існує кілька різних підходів до архітектури сховища зображень, кожен зі своїми перевагами та недоліками. Одним із підходів є використання централізованого сховища зображень, де всі дані зображення зберігаються в одній базі даних. Цей підхід є простим і зрозумілим, але він може бути непридатним для галузей, які генерують дуже великі обсяги даних зображень.

Інший підхід полягає у використанні розподіленого сховища зображень, де дані зображення зберігаються в кількох базах даних. Цей підхід більш масштабований і ефективний, ніж централізоване сховище зображень, але також може бути складнішим у реалізації.

Також можливий і гібридний підхід, коли деякі дані зображення зберігаються в централізованій базі даних, а інші дані зберігаються в

розподілених базах даних. Така архітектура забезпечує оптимальний баланс між масштабованістю та простотою управління, що робить її особливо привабливою для галузей з різноманітними вимогами до обсягів збережених зображень. Дослідження в цій області зосереджені на вирішенні комплексу проблем, пов'язаних з ефективним керуванням великомасштабними наборами візуальних даних. У рамках проведеного аналізу здійснено оцінку продуктивності різних архітектурних рішень сховищ зображень.

Результати дослідження надають цінну інформацію для розробки архітектури сховищ зображень, що здатні ефективно працювати з великими обсягами даних. Виявлені ключові проблеми та можливості надають основу для вдосконалення технологій зберігання та обробки цифрових зображень, а запропоновані підходи мають потенціал покращити ефективність та масштабованість систем управління візуальною інформацією. Це дослідження робить внесок у вирішення актуальних проблем обробки великих обсягів зображень та відкриває перспективи для подальших наукових пошуків [48].

Для оцінки ефективності методів обробки великих даних проведено порівняльний аналіз підходів, що використовуються у розробці гібридного сховища зображень. В таблиці 2.3 наведено кількісні показники порівняння методів обробки великих даних, що використовуються в гібридних сховищах зображень. Оцінюються ключові параметри, такі як швидкість обробки, масштабованість, точність аналізу та можливість інтеграції з AI-алгоритмами.

Таблиця 2.3 – Порівняльний аналіз методів обробки великих даних

Метод	Тип аналізу	Точність (%)	Час обробки (мс)	Використання пам'яті (МВ)	Переваги	Недоліки	Застосування
ResNet 50	Глибоке навчання (CNN)	95.6	120	450	Висока точність, ефективний аналіз зображень	Великі вимоги до обчислювальних ресурсів	Класифікація та розпізнавання об'єктів
YOLO v5	Нейронні мережі (CNN)	92.3	75	320	Швидкість обробки	Менша точність, ніж у ResNet	Виявлення об'єктів у реальному часі

FAISS	Векторний пошук	94.5	60	280	Висока швидкість пошуку за подібністю	Вимагає попереднього за індексування	Пошук схожих зображень
OpenAI CLIP	Трансформерна модель	96.8	200	600	Висока здатність розпізнавати контент на рівні семантики	Високе споживання пам'яті	Аналіз текстово-візуальних взаємозв'язків
Map Reduce	Розподілені обчислення	90.0	300	350	Масштабованість для великих обсягів даних	Висока латентність	Аналіз масивних наборів даних
Apache Spark	Стрімінг-обробка	93.5	90	400	Висока продуктивність, масштабованість	Потребує значних ресурсів	Аналіз поточкових даних у реальному часі

Як видно з таблиці, методи на основі глибокого навчання (ResNet50, YOLOv5) забезпечують високу точність аналізу зображень, однак вони вимагають значних обчислювальних ресурсів. У свою чергу, векторний пошук (FAISS) показує відмінну продуктивність у знаходженні схожих об'єктів. Класичні підходи, наприклад, SQL-оптимізація є менш ресурсоємними, але поступаються в швидкості обробки складних запитів. Для аналізу потокових даних застосовуються Apache Spark та MapReduce, що дозволяє обробляти великі обсяги даних у розподілених середовищах.

Таким чином, запропоновано новий підхід до підвищення ефективності та масштабованості архітектури сховища зображень в епоху великих даних шляхом поєднання попередньої обробки даних, стиснення та дедуплікації, розподіленої обробки та паралельних обчислень, машинного та глибокого навчання. Зокрема, розроблено та впроваджено систему, яка здатна ефективно обробляти значні масиви візуальних даних. Основні завдання такої розробки включають оптимізацію витрат на зберігання та обробку зображень, підвищення швидкодії пошукових операцій, а також забезпечення належного рівня захисту та конфіденційності інформації. Запропонований підхід пройшов експериментальну апробацію та був зіставлений з існуючими методами, описаними в науковій літературі. Результати порівняльного аналізу

підтвердили високу ефективність розробленого рішення та його переваги над альтернативними підходами.

2.2 Метод оптимізованої індексації в гібридній моделі зберігання зображень

У сучасну епоху зростання обсягів даних актуалізується необхідність в ефективних та масштабованих рішеннях для зберігання візуальної інформації. Цей підрозділ монографії присвячено розробці нового методу оптимізованої індексації, застосованого в гібридній моделі зберігання зображень. На основі математичного апарату проведено аналіз традиційних та гібридних алгоритмів індексації з акцентом на їх масштабованості та показниках продуктивності. Спираючись на аналітичні вирази та алгоритмічні виведення, розкрито архітектурні особливості запропонованої гібридної моделі. Для валідації ефективності розробленого методу проведено серію експериментальних досліджень на реальних наборах даних, які демонструють його переваги у порівнянні зі звичайними моделями. Порівняльний аналіз, підкріплений математичними критеріями, підтвердив ефективність та надійність запропонованого методу для різних типів даних. Отримані результати свідчать про значний потенціал розробленого підходу до оптимізованої індексації в галузі зберігання та обробки великомасштабних наборів зображень, що створює підґрунтя для подальших досліджень у цьому напрямку.

В умовах цифрової трансформації відбувається кардинальна зміна парадигми створення, обробки та зберігання інформації. Стрімкий розвиток мобільних пристроїв, соціальних мереж та Інтернету речей (IoT) призводить до експоненціального зростання обсягів даних, зокрема візуальної інформації. Традиційні системи зберігання, розроблені для роботи зі структурованими наборами даних обмеженого розміру, демонструють свою неефективність в умовах сучасних викликів [98, 99]. Їхня неефективність стає очевидною з точки зору масштабованості, швидкості пошуку, енергоспоживання та

економічної ефективності, особливо коли їм доручено зберігати та отримувати об'ємні та неструктуровані дані зображень.

Стрімке зростання обсягів даних не лише створило низку викликів, але й відкрило нові перспективи для інноваційних підходів у методології зберігання інформації. Інтеграція різних типів систем керування базами даних, як реляційних, так і нереляційних, у поєднанні з прогресивними файловими системами, демонструє потенційні шляхи вирішення наявних проблем. Такий інтеграційний підхід дозволяє ефективно використовувати переваги обох типів систем, створюючи підґрунтя для розробки гібридних систем зберігання даних, що здатні забезпечити ефективне управління значними масивами інформації [100, 101].

Враховуючи актуальність зазначених проблем та перспективи їх вирішення, у цьому підрозділі монографії представлено результати дослідження, спрямованого на розробку та обґрунтування нової гібридної моделі зберігання зображень. Запропонована модель створена з урахуванням специфічних вимог середовища великих даних та базується на оптимізованому методі індексації, що є її ключовою відмінністю від існуючих аналогів.

Запропонований підхід ґрунтується на поєднанні фундаментальних математичних принципів та їх практичної реалізації. Метою дослідження є формування комплексного розуміння про розроблену гібридну модель, що включають її базові концепції, математичний апарат, структурні особливості та емпірично підтверджені показники ефективності. Такий підхід має на меті створення інструментарію для дослідників, розробників архітектури даних та фахівців-практиків, що дозволить ефективно вирішувати завдання обробки та аналізу великомасштабних наборів візуальної інформації в умовах зростання обсягів даних.

Математичне підґрунтя методів індексації є ключовим аспектом ефективного пошуку даних, особливо в контексті великомасштабних систем зберігання інформації. Розуміння математичних структур, що лежать в основі

алгоритмів індексації, дозволяє оцінити їх ефективність, масштабованість та надійність. Представлено аналіз основних математичних концепцій, які формують теоретичну базу методів індексації та визначають їх практичну значущість.

Основні принципи. В основі більшості механізмів індексації лежить концепція хешування, тобто метод, який відображає дані в масив значень фіксованого розміру. Математична функція, що використовується для хешування, називається «хеш-функцією», прагне виділити унікальні хеш-коди для різних введених даних [102]. Ідеальна хеш-функція $h(k)$ визначається як:

$$h(k) = k \bmod m, \quad (2.1)$$

де k – вхідні дані, m – розмір хеш-таблиці.

Ефективність хеш-функції оцінюється за кількома критеріями – рівномірність розподілу значень, швидкість обчислення, мінімізація колізій. При цьому ймовірність колізій $P(n, m)$ для n елементів у хеш-таблиці розміром m можна оцінити як:

$$P(n, m) = 1 - e^{-\frac{n(n-1)}{2m}}. \quad (2.2)$$

Іншою поширеною структурою даних для індексації є B -дерево. Ця самобалансуюча деревоподібна структура підтримує відсортовані дані таким чином, що дозволяє шукати, вставляти та видаляти їх за логарифмічний час [103, 104]. Глибина d B -дерева з n вузлами і коефіцієнтом розгалуження t математично представлена як:

$$d \leq \log_t \left(\frac{n+1}{2} \right). \quad (2.3)$$

Важливою характеристикою B -дерева є його збалансованість, що гарантує логарифмічну складність основних операцій. Час пошуку елемента T можна виразити як:

$$T = O(\log t(n)), \quad (2.4)$$

де n – кількість елементів, t – мінімальний степінь дерева.

Використання Elasticsearch забезпечує логарифмічну складність $O(\log n)$ завдяки структурі зворотного індексу та оптимізованим tree-based алгоритмам. Це дозволяє ефективно масштабувати пошук при зростанні обсягів даних. У базах даних з величезною кількістю рядків, але обмеженою різноманітністю значень даних, індексація растрових зображень виявляється надзвичайно ефективною. Якщо v представляє різні значення даних, а r означає рядки, то бітовий індекс повинен містити $v \times r$ бітів.

Ефективність індексної структури E можна оцінити за формулою:

$$E = \frac{\alpha}{T} + \frac{\beta}{S} , \quad (2.5)$$

де T – середній час доступу, S – обсяг пам'яті, α, β – вагові коефіцієнти.

Ця оцінка дозволяє збалансувати швидкість доступу та витрати пам'яті, що особливо важливо для великих наборів даних зображень. Вагові коефіцієнти α та β можуть налаштовуватися залежно від конкретних вимог системи.

Для баз даних, які в основному обробляють дані часових рядів, використовуються специфічні методи індексації, такі як B -дерево з поділом на час. Ці індекси сегментують дані в хронологічному порядку, оптимізуючись для запитів, які отримують дані за певні проміжки часу.

Аналітична основа ефективності індексації. Ефективність методу індексації часто можна виразити в часі відповіді запиту T , який можна представити у вигляді:

$$T = T_s + T_r , \quad (2.6)$$

де T_s – час, витрачений на пошук в індексі, T_r – час, необхідний для отримання даних після пошуку за індексом.

Оптимізація для T тягне за собою розумний баланс між зменшенням як T_s , так і T_r , на що впливають такі фактори, як глибина дерева індексації, колізії хешів і накладні витрати на зберігання.

У контексті розробленої гібридної моделі зберігання зображень, розуміння математичних основ індексації має вирішальне значення. Це дозволяє сформулювати метод індексації, оптимізований для великомасштабних зображень, що забезпечує високу швидкість доступу до інформації при мінімальних витратах на зберігання. Розглянемо, яким чином ці фундаментальні принципи індексації трансформуються та розвиваються в рамках запропонованого оптимізованого методу, що є ключовим компонентом розробленої гібридної моделі.

Галузь зберігання даних зазнала значної еволюції, що супроводжувалася появою різноманітних механізмів, кожен з яких розвивається у відповідь на зростаючі вимоги до великих даних [105, 106]. Комплексний аналіз цих моделей, що охоплює не лише їх структурні та операційні аспекти, але й математичну сутність, дозволяє чітко визначити їхні переваги та обмеження. Перш за все розглядаються аналітичні вирази, що описують ключові моделі зберігання даних, проводиться оцінка їхньої ефективності на основі математичних критеріїв. Такий підхід забезпечує глибоке розуміння принципів функціонування різних систем зберігання та створює підґрунтя для їх порівняльного аналізу. Це дозволяє виявити ключові фактори, що впливають на ефективність систем, та визначити перспективні напрямки їх оптимізації. Розглянемо детальніше деякі з них.

Реляційні системи управління базами даних (СУБД). Базовою одиницею СУБД є таблиця, де дані зберігаються у вигляді рядків і стовпців. Відношення R в СУБД можна представити у вигляді набору кортежів:

$$R = \{t_1, t_2, \dots, t_n\}, \quad (2.7)$$

де кожен кортеж t представляє рядок.

Основним показником для оцінки ефективності запитів у СУБД є часова складність. Для таблиці з n рядками вводимо позначення: Пошук: $O(n)$; Вставка: $O(1)$ (за умови відсутності індексації); Видалення: $O(n)$.

Перспективність СУБД полягає в її структурованому підході до управління даними. За своєю суттю СУБД була розроблена, щоб імітувати те, як люди традиційно організовували та обробляли інформацію, насамперед за допомогою табличних форматів. Цей дизайн, орієнтований на людину, робить реляційні бази даних особливо інтуїтивно зрозумілими та простими у взаємодії для широкого спектру застосувань. Рядки і стовпці таблиці СУБД нагадують традиційну електронну таблицю або бухгалтерську книгу. Однак, за цим прямолінійним зовнішнім виглядом можна визначити складні відношення між різними таблицями. Такі зв'язки дають змогу створювати складні запити, які можуть перехресно посилатися на кілька точок даних, витягуючи інформацію, яка в іншому випадку залишалася б прихованою. Процес проектування цих відношень і забезпечення того, щоб вони точно представляли властиву структуру і обмеження даних, відоме як нормалізація бази даних. Цей процес має вирішальне значення для запобігання аномаліям даних і забезпечення довгострокової надійності бази даних.

NoSQL бази даних (на основі документів). На відміну від RDBMS, NoSQL бази даних зберігають дані в гнучких схемах, які часто представляються у вигляді JSON-подібних документів [107]. Дано документ D з k ключами:

$$D = \{(k_1, v_1), (k_2, v_2), \dots, (k_n, v_n)\}, \quad (2.8)$$

де кожна пара (k, v) позначає комбінацію ключ-значення.

Орієнтовний показник продуктивності: Пошук: $O(1)$ для пошуку на основі ключів, але може змінюватися залежно від базової структури бази даних; Введення: $O(1)$; Видалення: $O(1)$.

Поява баз даних NoSQL виникла через потребу обробляти величезного обсягу масивів неструктурованих або напівструктурованих даних. З поширенням Інтернету дані почали надходити з різних джерел у непередбачуваних форматах. Традиційні структури СУБД, хоча і вправні в обробці структурованих даних, часто спотикалися, стикаючись з такими

неоднорідними наборами даних. Ці системи розроблені з метою забезпечення високого рівня гнучкості та масштабованості, що дозволяє організаціям ефективно адаптуватися до динамічних змін інформаційного середовища.

Зокрема, NoSQL-бази даних на основі документів зазнали великого сплеску популярності завдяки своїй придатності для сучасних веб-додатків. Їхнє представлення даних, подібне до JSON, за своєю суттю сумісне з веб-технологіями, що спрощує процес прийому, обробки та доставки даних. Крім того, підтримуючи динамічні схеми, вони дозволяють розробникам інтегрувати та розвивати програми без накладних витрат на реструктуризацію бази даних. Ця гнучкість особливо цінна в сучасних швидких циклах розробки, де додатки повинні реагувати на мінливі вимоги бізнесу. З іншого боку, хоча вони забезпечують величезну гнучкість, вони можуть вимагати більш ретельного моделювання даних, щоб запобігти надлишковості даних і зберегти стабільність.

Системи зберігання об'єктів. Об'єктні системи зберігають дані як об'єкти у плоскому просторі імен, який часто використовується для великих, неструктурованих наборів даних. Об'єкт O можна математично описати так:

$$O = (ID, Metadata, Data), \quad (2.9)$$

де ID є унікальним ідентифікатором, а метадані надають допоміжну інформацію про об'єкт.

Позначення орієнтовних показників продуктивності: Отримання: $O(1)$ за допомогою ідентифікатора; Введення: $O(1)$; Видалення: $O(1)$.

На відміну від традиційних файлових систем і блокових сховищ, об'єктні системи зберігання призначені для безперебійної обробки великих обсягів даних. Притаманна їм масштабованість у поєднанні з багатими можливостями метаданих робить їх майже ідеальними для хмарних середовищ і сучасних додатків, де дані можуть зростати непередбачуваним чином. Плоский простір імен гарантує, що пошук даних залишається ефективним незалежно від розміру, уникаючи ієрархічних складнощів традиційних файлових систем. В

результаті об'єктні системи зберігання стали основою для багатьох рішень хмарних сховищ, забезпечуючи як продуктивність, так і економічну ефективність.

Математичне обґрунтування продуктивності. Наведені вище часові складності служать ідеалізованими орієнтирами, підкріпленими певними припущеннями. Реальні сценарії можуть мати відмінності через наступні фактори. Накладні витрати на індексацію – наявність вторинних індексів може вплинути на час пошуку та вставки. Апаратні обмеження: операції вводу/виводу на диску можуть спричиняти затримки, що відхиляються від очікуваних контрольних показників. Розподіл даних: викривлений розподіл даних у базах даних може впливати на час пошуку, особливо якщо певні значення даних є більш поширеними.

По суті, хоча аналітичні вирази пропонують структуровану основу для розуміння різних моделей зберігання, їх слід інтерпретувати в контексті реальних сценаріїв та обмежень. У міру того, як здійснюється перехід до гібридної моделі зберігання зображень, стає вкрай необхідним зіставити її з цими існуючими моделями, розрізняючи саме її унікальні переваги та власні потенційні проблеми.

Далі відзначимо гібридну модель зберігання зображень, її архітектуру та формулювання. У контексті розвитку систем зберігання даних, гібридна модель зберігання зображень постає як рішення, адаптоване для ефективного управління зображеннями в контексті великих даних [108, 109]. Ця модель поєднує переваги різних парадигм зберігання та пропонує як структурну гнучкість, так і оптимізовану продуктивність. У цьому розділі представлено детальний аналіз архітектури гібридної моделі зберігання зображень, включаючи математичне обґрунтування її ключових компонентів. Особлива увага приділяється формалізації основних принципів функціонування моделі та їх теоретичному обґрунтуванню.

Зробимо *концептуальний огляд гібридної моделі*. За своєю суттю, гібридна модель зберігання зображень поєднує гнучкість схеми баз даних

NoSQL зі структурованою організацією RDBMS, доповненою принципами об'єктного зберігання для обробки величезних обсягів неструктурованих даних зображень. Формально гібридне сховище H можна охарактеризувати так:

$$H = (R, D, O), \quad (2.10)$$

де R – представляє реляційні компоненти даних; D – представляє компоненти на основі документів NoSQL; O – позначає компоненти сховища об'єкта.

Архітектурні компоненти та рецептури. Компонент реляційних даних R : Структура – задається у вигляді набору таблиць T , де кожна таблиця є набором кортежів.

$$T = \{t_1, t_2, \dots, t_n\}. \quad (2.11)$$

Випадок використання: підходить для структурованих метаданих, пов'язаних із зображеннями, таких як часові позначки, ідентифікатори користувачів та категорії зображень.

Компонент D на основі документів: Структура – колекція документів C , де кожен документ має гнучку схему.

$$C = \{D_1, D_2, \dots, D_m\}, \quad (2.12)$$

де D – так, як описано раніше.

Випадок використання: ідеально підходить для зберігання різноманітних атрибутів зображень, таких як теги, анотації або контент, створений користувачами.

Компонент O об'єктного сховища: Структура – плоский простір імен, що містить об'єкти, кожен з яких пов'язаний з унікальним ідентифікатором.

$$O = (ID, Metadata, ImageData). \quad (2.13)$$

Тут $ImageData$ представляє фактичні значення пікселів або дані стисненого зображення.

Приклад використання: ефективне зберігання великих файлів зображень, забезпечуючи при цьому швидке отримання на основі унікальних ідентифікаторів.

Математичне представлення потоку даних. Сила гібридної моделі полягає в тому, як дані перетікають і взаємопов'язані між її компонентами. Позначимо функцію, що відображає метадані зображення з реляційного компонента на його атрибути в компоненті на основі документа, як f_R , а атрибути відображення функції на фактичне зображення в об'єктному сховищі – як f_D .

Задані метадані зображення M у R :

$$D_{attributes} = f_R(M). \quad (2.14)$$

Задані атрибути A в D :

$$O_{image} = f_D(A). \quad (2.15)$$

Таким чином, отримання зображення на основі його метаданих включає складену функцію:

$$O_{image} = f_D(f_R(M)). \quad (2.16)$$

Переваги гібридної архітектури. Математичне формулювання показує здатність моделі управляти як структурованими, так і неструктурованими даними. Розділяючи дані на основі їхньої природи та корисності, гібридна модель забезпечує оптимізовані операції зберігання та пошуку, що робить її особливо придатною для сценаріїв великих даних із великими обсягами зображень [110, 111].

Індексація є критично важливою основою для ефективного пошуку даних, особливо у великих наборах даних, таких як ті, що зустрічаються в контексті великих даних. Гібридна модель зберігання зображень, з огляду на її композитну природу, вимагає нової стратегії індексації, яка плавно охоплює її різномірні компоненти. У цьому розділі монографії далі представлено оптимізований метод індексації, розроблений для даної гібридної моделі.

Математичні основи оптимізованого індексу

Для набору даних X позначаємо наш оптимізований індекс. Формально для будь-якого заданого запиту q метою $I \in$ мінімізація часу пошуку, T [112].

$$I: X \rightarrow q, \quad (2.17)$$

$$(I(q)) < T(q), \quad (2.18)$$

де $T(I(q))$ – представляє час отримання за допомогою індексу, $T(q)$ – позначає час пошуку без використання індексу.

У системі використовуються вектори ознак із фіксованою розмірністю, яка залежить від моделі опису зображень (типовою є – 128, 256 або 512 вимірів), що відповідає поширеним форматам у системах глибинного векторного пошуку. На Рис. 2.1 представлено узагальнений алгоритм, що охоплює ключові етапи індексації та пошуку за ознаками – збереження зображення, вилучення ознак за допомогою моделі ResNet50, створення векторного представлення та його індексація в Elasticsearch. Пошуковий запит користувача також перетворюється у вектор і зіставляється з уже індексованими векторами.

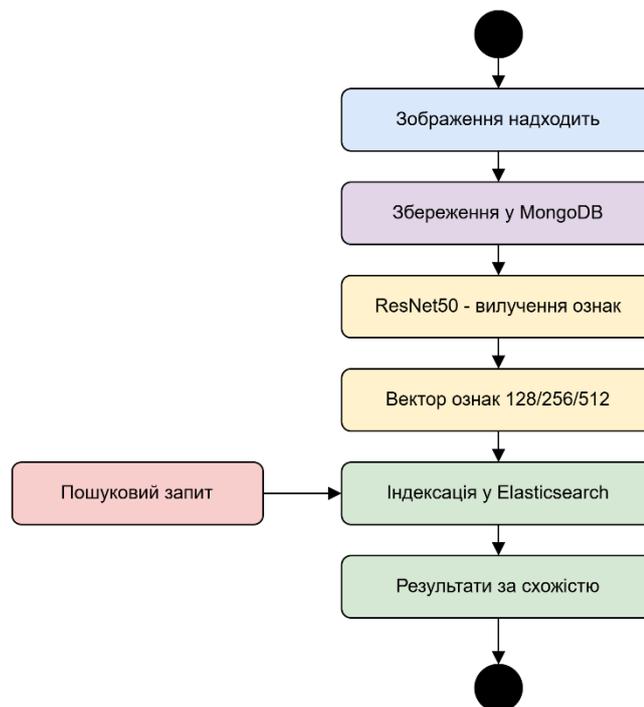


Рисунок 2.1 – Алгоритм індексації та пошуку зображень у гібридній системі

Взаємодія між окремими компонентами системи проілюстрована на Рис. 2.2, де відображено послідовність викликів сервісів. UML Sequence Diagram демонструє передачу даних між користувачем, модулями обробки, збереження, векторизації та пошуку. Такий підхід дозволяє відобразити не лише логіку, але й реальну інтеграцію сервісів, актуальну для мікросервісної архітектури.

Таким чином, модель поєднує функціональний підхід (алгоритм) та архітектурний рівень (взаємодію), дозволяючи ефективно реалізувати гібридну систему зберігання і пошуку – документо-орієнтовану (MongoDB), векторну (Elasticsearch) та модельну (ResNet50). Щоб досягти цієї оптимізації, наш індекс використовує структурні нюанси гібридної моделі.

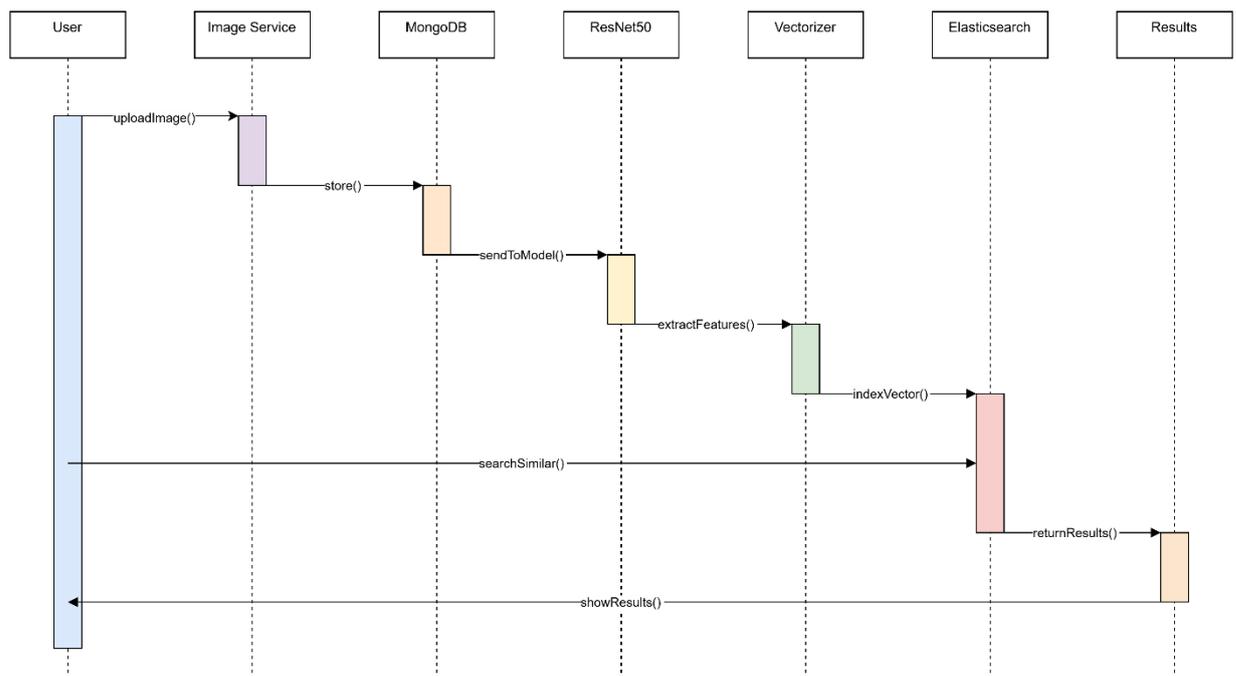


Рисунок 2.2 – Взаємодія компонентів системи при обробці зображення (UML Sequence Diagram)

Ієрархічна структура індексації. З огляду на тристоронній характер гібридної моделі, оптимізований індекс має ієрархічну структуру. Представлений як I_h , він може бути визначений як:

$$I_x = (IR, ID, IO), \quad (2.19)$$

де IR – це індекс, пов'язаний з компонентом реляційних даних; ID – відноситься до компонента на основі документа; IO – охоплює компонент зберігання об'єктів.

Алгоритмічна побудова оптимізованого індексу.

Алгоритм 1:

Побудова ієрархічно оптимізованого індексу.

Вхідні дані: гібридне сховище $H=(R,D,O)$

Вихідні дані: Ієрархічний індекс I_h

ПОЧАТОК

$I_R = \text{BUILD_INDEX}(R)$

$I_D = \text{BUILD_INDEX}(D)$

$I_O = \text{BUILD_OBJECT_INDEX}(O)$

$I_h = (I_R, I_D, I_O)$

ПОВЕРНЕННЯ I_h

КІНЕЦЬ

Тут функція $BUILD_INDEX$ використовує методологію на основі B -дерева як для реляційних, так і для документальних компонентів через їх структурну природу. І навпаки, $BUILD_OBJECT_INDEX$ використовує підхід, заснований на хеші, використовуючи плоский простір імен об'єктного сховища. Візуалізація процесу побудови ієрархічного індексу представлена на схемі нижче (Рис. 2.3).

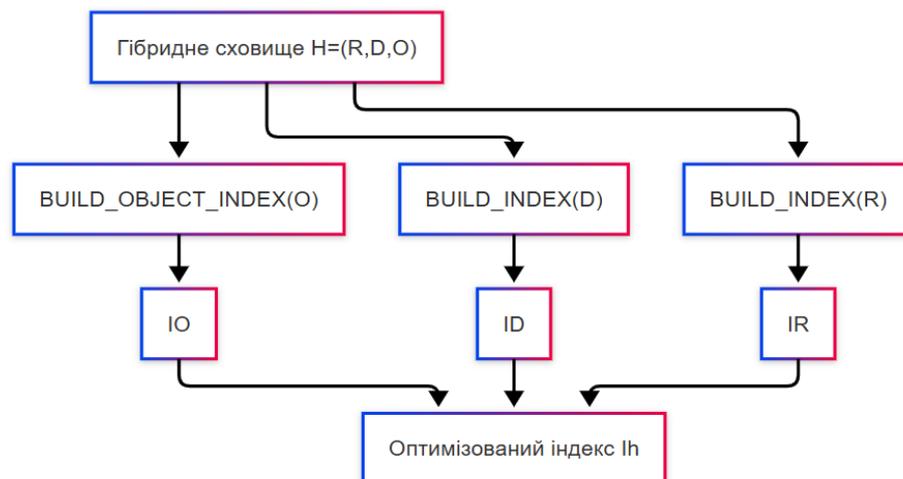


Рисунок 2.3 – Ієрархічна структура оптимізованого індексу

У разі застосування прогнозування розміщення об'єктів у сховищі, у якості вхідних ознак до моделі використовувалися: розмір зображення (у байтах), кількість звернень до об'єкта, тип контенту (фото/скан/графіка), хеш-ідентифікатор, дата останнього доступу. Це дозволяє системі адаптивно переміщати так звані «гарячі» об'єкти у швидші типи сховищ (наприклад, SSD, in-memory, тощо).

Математичні докази оптимальності. Ієрархічний індекс I_h забезпечує логарифмічну часову складність для операцій пошуку за всіма трьома компонентами [113].

Доведення. Враховуючи, що компоненти IR та ID використовують B -дерева як структури даних, часова складність пошукових операцій у збалансованих B -деревах у найгіршому випадку становить $O(\log n)$. Для компоненти IO , яка базується на хеш-функціях та передбачає рівномірний розподіл хеш-значень, середня часова складність пошуку становить $O(1)$. У найгіршому випадку, за наявності колізій хешування, складність може зрости до $O(n)$, де n визначає кількість колізій. Отже, агрегуючи часові складності за всіма трьома компонентами, можна стверджувати, що ієрархічний індекс I_h забезпечує загальну логарифмічну часову складність для більшості сценаріїв пошуку.

Наведене доведення підтверджує ефективність запропонованого оптимізованого методу індексації, що гарантує швидкий пошук та отримання зображень навіть в умовах обробки надвеликих обсягів даних.

Ієрархічний індекс I_h представляє собою оптимальне поєднання перевірених обчислювальних структур, спеціально розроблених для вирішення задач, пов'язаних з обробкою великих і різноманітних наборів даних. Імплементация B -дерев у компонентах IR та ID забезпечує використання структури даних, що характеризується високим рівнем адаптивності та послідовності. При зростанні обсягів даних B -дерева гарантують збереження ефективності пошукових операцій, що робить їх незамінними для систем управління базами даних.

Застосування альтернативного підходу на основі хешування в компоненті *IO* реалізується принцип прямого доступу до табличних даних. У той час як хеш-функції зазвичай дозволяють майже миттєве отримання даних шляхом перетворення вхідних даних в унікальний індекс, потенційна можливість виникнення колізій хешування створює певні обмеження. Однак у переважній більшості практичних сценаріїв переваги від пошуку постійного часу в середньому випадку суттєво переважають недоліки, пов'язані з можливими колізіями.

У комбінованій формі ієрархічного індексу I_h використовуються сильні сторони як *B*-дерев, так і систем, заснованих на хешуванні. Конструкція гарантує, що навіть якщо один компонент може зіткнутися з неефективністю, інший втручається, щоб збалансувати ваги, надаючи користувачам оптимальний час пошуку. Ця дворівнева мережа безпеки робить I_h надзвичайно надійною системою для великомасштабного пошуку даних.

Запропонована архітектура індексації має фундаментальне значення в умовах сучасної цифровізації. Оскільки користувачам потрібен миттєвий доступ до масштабних сховищ даних, починаючи від цифрових бібліотек фотографій і закінчуючи великими наборами геопросторових наборів даних. Потребу в швидких і надійних механізмах пошуку не можна недооцінювати. Розроблений ієрархічний індекс I_h , завдяки своїй ефективності та адаптивності, демонструє потенціал для встановлення нових стандартів у галузі систем пошуку та зберігання даних.

Таким чином, математично обґрунтована архітектура та алгоритмічне забезпечення оптимізованого методу індексації пропонуються у якості комплексного рішення для гібридної моделі зберігання зображень [114, 115]. Здатність методу ефективно обробляти різні структури даних, адаптивно використовуючи переваги кожної з них, визначає його ключову роль у масштабованих операціях з пошуку зображень. Подальший матеріал присвячено практичній верифікації теоретичних положень через аналіз

результатів впровадження запропонованої стратегії індексації в реальних умовах.

2.3 Обґрунтування методів зберігання та обробки зображень для реалізації гібридної моделі

У цьому підрозділі монографії представлені методи та матеріали, які використовуються для вирішення проблем, пов'язаних із керуванням великими обсягами даних зображень в архітектурі сховища зображень. У сучасних системах управління великими масивами зображень критично важливим є ефективний механізм збереження та доступу до даних.

Гібридна архітектура сховища поєднує реляційні (SQL), нереляційні (NoSQL) та децентралізовані (IPFS/ Hyperledger) сховища та забезпечує оптимальне співвідношення між продуктивністю, масштабованістю та безпекою. Дозволяє оптимально розподіляти файли між локальним, централізованим та децентралізованим сховищем залежно від їхніх характеристик. Ця модель дозволяє розподіляти зображення між різними рівнями сховища – SQL використовується для структурованих мета-даних та швидких транзакцій, NoSQL застосовується для напівструктурованих даних, які не підпадають під строгі схеми, IPFS/блокчейн ідеальні для великих зображень, що вимагають децентралізованого збереження та захисту від змін.

Послідовність взаємодії між компонентами цієї архітектури наведена на Рис. 2.4 у вигляді UML Sequence Diagram. Вона включає три основні етапи.

Етап 1. Користувач завантажує зображення, яке обробляється системою та розділяється на різні типи даних.

Етап 2. Дані зберігаються у відповідних сховищах – SQL, NoSQL або IPFS/блокчейн.

Етап 3. Запити користувача проходять через механізм пошукової системи, яка взаємодіє з усіма сховищами для оптимального отримання даних.

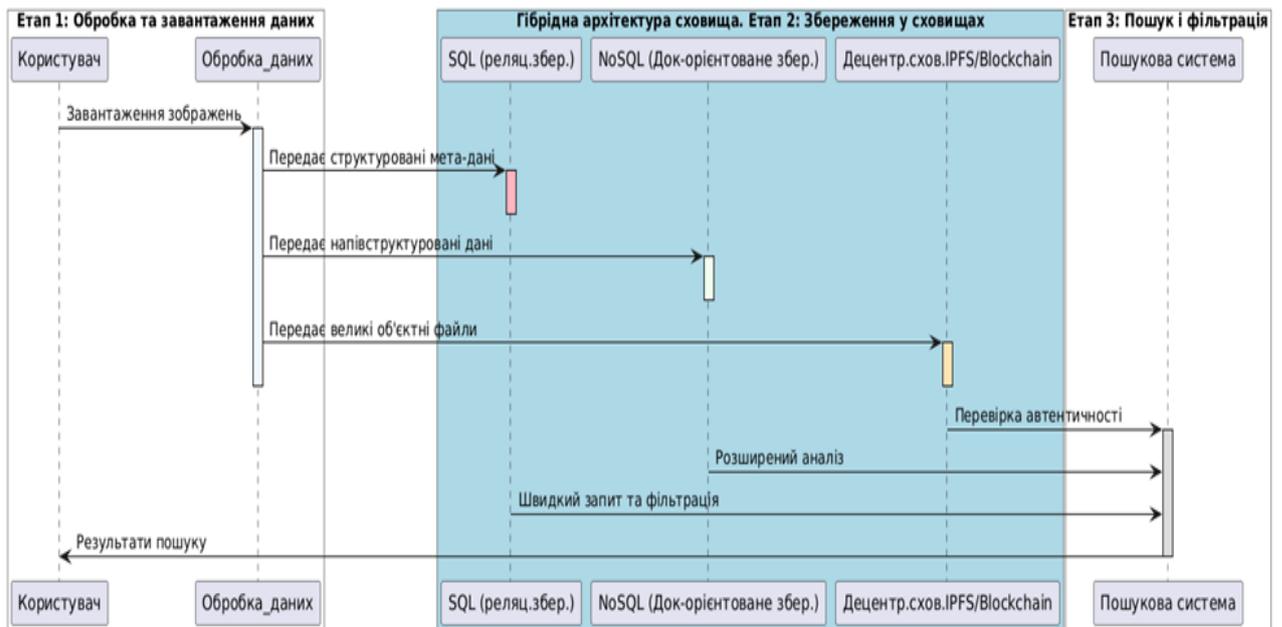


Рисунок 2.4 – UML Sequence Diagram взаємодії між SQL, NoSQL та IPFS

Така архітектура дозволяє не лише гнучко зберігати та обробляти різнотипні дані, але й забезпечує цілісність, масштабованість і продуктивність системи в цілому.

2.3.1 Структурне моделювання потоків даних у гібридному сховищі

Представлено DFD-діаграму, яка описує структуру і функціональну взаємодію основних компонентів гібридної інформаційної технології, призначеної для інтелектуальної обробки, зберігання, пошуку та безпечного доступу до візуальних даних. Ця діаграма моделює потоки інформації між ключовими компонентами технології та визначає їхні вхідні, вихідні дані, а також способи взаємодії (Рис. 2.5).

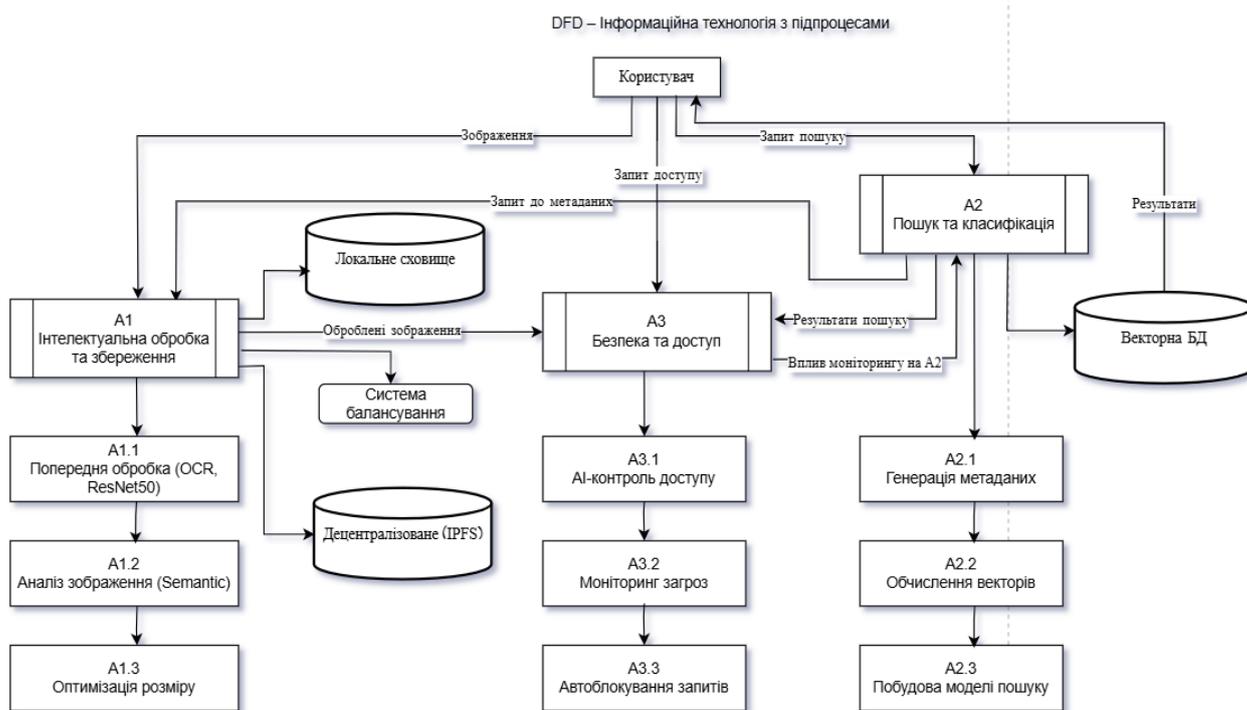


Рисунок 2.5 – Діаграма потоків даних (DFD) інформаційної технології

Інформаційна технологія складається з трьох основних комплексів інформаційних процесів (КІП). Основні компоненти DFD:

– Користувач – ініціатор дій (завантаження зображення, пошук інформації).

Вхідний запит: завантаження зображення або пошук у сховищі.

Вихідний запит: отримання результату після AI-обробки.

– Інтелектуальна обробка (КІП 1) – AI-класифікація, стиснення, розподіл між сховищами.

Попередня обробка та класифікація

Використовується ResNet50, YOLOv5, OCR для визначення категорії.

Аналіз контенту: текст / обличчя / сцени.

– Гібридне сховище та збереження (КІП 1) – SQL/NoSQL БД для локальних файлів, IPFS/Hyperledger для децентралізованого збереження.

AI-оптимізація вибору сховища

LSTM-аналіз навантаження для прогнозування продуктивності.

Графові алгоритми для вибору локального або децентралізованого сховища (IPFS, Hyperledger Fabric).

Використання блокчейн-смарт-контрактів для підтвердження операцій та контролю змін файлів.

Оптимізація розміру та перевірка якості

Інтелектуальне стиснення та зменшення розміру без втрати якості.

Перевірка якості зображень перед збереженням.

– Інтелектуальний пошук та класифікація (КІП 2) – пошук зображень за подібністю через векторні бази (FAISS, Milvus).

Пошук за подібністю (Metric Learning, FAISS, OpenAI CLIP)

Обчислення векторів зображень та їх порівняння.

Побудова моделі пошуку на основі аналізу метаданих.

Динамічне кешування з TTL

Використання кешу для швидкого доступу до популярних запитів.

Генерація метаданих

Автоматичне створення тегів, описів.

Використання нейромережових моделей для розширеного аналізу.

– Система безпеки (КІП 3) – шифрування, контроль доступу через блокчейн та AI-аналіз загроз.

AI-аудит запитів та моніторинг загроз

Використання аналітики аномалій (anomaly detection, IDS/IPS).

Автоматичне блокування підозрілих запитів.

Шифрування та блокчейн-аутентифікація

AES-256 для збереження конфіденційних файлів.

Використання блокчейн-ідентифікації для авторизації користувачів.

Динамічне управління доступом через смарт-контракти

Регулювання прав доступу на основі AI-аналізу.

Вся історія фіксується у блокчейні.

Сценарії роботи системи у DFD:

Етап 1. Користувач завантажує зображення → система обробляє та класифікує контент.

- Користувач надсилає файл у КІП 1 (AI-класифікація, OCR, FaceNet).
- Класифікація визначає тип файлу (приватний/публічний/чутливий).
- Якщо файл приватний, він передається у КІП 3 (шифрування, блокчейн-аудит).
- Якщо файл публічний, він записується у IPFS/Hyperledger Fabric.

Етап 2. Збереження у відповідному сховищі. Файл передається у локальне або децентралізоване сховище залежно від політик безпеки та AI-аналізу (локальне БД/IPFS/блокчейн).

- Використання AI-оптимізації вибору сховища.
- Аналізується навантаження на сервери, швидкість доступу, конфіденційність.
- Файл передається у локальне або децентралізоване сховище.

Етап 3. Пошук та індексація. Користувач здійснює пошук → AI-аналізує запит, перевіряє мета-дані та здійснює пошук у векторній баз.

- Запит на пошук обробляється у КІП 2 (векторна база даних, FAISS, OpenAI CLIP).
- Якщо запит популярний, він кешується з урахуванням TTL.
- Якщо немає збігів у кеші → пошук у глобальному сховищі.

Етап 4. Контроль безпеки та аудит доступу. Користувач отримує результати пошуку, при цьому система контролює доступ через AI-аналіз загроз та смарт-контракти.

- AI-аналіз перевіряє історію дій користувача.
- Верифікація через блокчейн-ідентифікацію.
- Якщо виявлено аномальну активність, доступ блокується.

DFD візуалізує рух даних між компонентами, пояснюючи їхню взаємодію.

2.3.2 Керування великими обсягами даних зображень в архітектурі сховища зображень

Загальна мета цього підрозділу монографії полягає в тому, щоб описати запропоновані нові методи та технології, які можуть підвищити ефективність і масштабованість архітектури сховищ зображень, одночасно вирішуючи ключові проблеми, пов'язані з якістю даних, безпекою та доступністю [116].

Перший крок у запропонованому підході включає попередню обробку даних для очищення та перетворення необроблених даних зображення. Зокрема, застосовані такі методи, як шумозаглушення, посилення контрастності та нормалізація кольорів, щоб покращити якість зображень і зменшити варіативність.

Далі, використано методи стиснення та дедуплікації, щоб зменшити розмір даних зображення, зберігаючи важливі функції. Виконано експеримент з різними алгоритмами стиснення, такими як JPEG і PNG, а також досліджувано методи дедуплікації, такі як фрагментація на основі вмісту та хешування подібності.

Балансування навантаження між різними рівнями сховища є критичним завданням для підтримки продуктивності системи. Для цього використовуються LSTM-моделі прогнозування та графові алгоритми вибору оптимального середовища.

Псевдокод AI-моделі балансування:

Algorithm. AI_Balancing

Input: System Load Data (L), Storage Availability (S)

Output: Optimal Storage Decision (S_opt)

1. Initialize LSTM model for predicting future load
2. Predict next time-step load: $L_{pred} = LSTM(L)$
3. Compute priority weight for each storage option:

$$W_{SQL} = f(T_{access}, cost, sensitivity)$$

$$W_{NoSQL} = f(speed, scalability)$$

$$W_{IPFS} = f(\text{size, availability})$$

4. Select storage S_{opt} with highest W value

5. Store image in S_{opt}

End

Запропонований AI-підхід дозволяє динамічно розподіляти навантаження між сховищами на основі прогнозу використання ресурсів. Окрім методів стиснення та дедуплікації, важливим аспектом керування великими обсягами даних є оптимальний розподіл зображень між різними сховищами. Запропонована AI-модель аналізує параметри зображення, такі як рівень конфіденційності, розмір, швидкість доступу та прогноз навантаження, щоб визначити оптимальне місце його збереження. Основні етапи вибору сховища включають: класифікацію контенту (ResNet50, YOLOv5), оцінку продуктивності сховища (SQL, NoSQL, IPFS), прогноз навантаження за допомогою LSTM.

Алгоритм. Оптимальний вибір сховища для зображення

Вхідні дані:

- S – множина доступних сховищ {SQL, NoSQL, IPFS},
- I – зображення, що підлягає збереженню,
- $P(I)$ – рівень конфіденційності зображення,
- $L(I)$ – розмір файлу,
- $T(S)$ – час доступу до сховища.

Вихідні дані:

- S_{opt} – оптимальне сховище.

Кроки:

Крок 1. Класифікація зображення.

- використати нейромережу ResNet50 для визначення категорії зображення,
- якщо зображення містить конфіденційні дані ($P(I)=\text{High}$), встановити пріоритет локального збереження (SQL).

Крок 2. Аналіз розміру та швидкості доступу.

- якщо $L(I) > 100$, пріоритет надається IPFS,
- якщо $T(SQL) < 100\text{ms}$, використати SQL,
- якщо $L(I) < 5\text{MBi}$ і $T(IPFS) > 200\text{ms}$, вибрати NoSQL.

Крок 3. Остаточне рішення.

- обрати сховище, яке мінімізує час доступу $S_{\text{opt}} = \text{argmin}ST(S)$,
- виконати AI-аналіз продуктивності LSTM для прогнозу завантаження сховища.

Результат:

- вибране сховище S_{opt} , що забезпечує мінімальне навантаження та максимальну продуктивність.

Запропонований алгоритм дозволяє автоматизувати вибір сховища на основі аналізу характеристик зображень та поточного навантаження системи для кожного зображення залежно від його характеристик та продуктивності системи. Для ефективного управління вибором сховища використовується AI-аналіз, що враховує ключові параметри продуктивності, безпеки та вартості. Формула для розрахунку пріоритету вибору сховища визначає, наскільки ефективним буде його використання, з урахуванням наступних факторів:

- часу доступу (T_{resp}), який визначає, наскільки швидко сховище може надати доступ до даних,
- вартості зберігання (C_{stor}), що впливає на довготривалу ефективність,
- рівня безпеки (S_{risk}), який оцінюється через AI-аналіз аномалій.

Розрахунок відбувається за допомогою наступної формули:

$$P(s) = \frac{W_{\text{lat}}}{T_{\text{resp}}} + \frac{W_{\text{cost}}}{C_{\text{stor}}} + \frac{W_{\text{sec}}}{S_{\text{risk}}}, \quad (2.20)$$

де $P(s)$ – пріоритет вибору сховища, $W_{\text{lat}}, W_{\text{cost}}, W_{\text{sec}}$ – вагові коефіцієнти для затримки, вартості та безпеки; T_{resp} – середній час відповіді сховища; C_{stor} – вартість зберігання; S_{risk} – ризик безпеки (оцінений через AI-аналіз аномалій).

Формула (2.20) дозволяє оцінити ефективність кожного сховища за відповідними ваговими коефіцієнтами. Менше значення $P(s)$ означає більш

пріоритетне сховище для збереження даних. Вагові коефіцієнти (W_{lat} , W_{cost} , W_{sec}) налаштовуються відповідно до стратегій системи (залежно від поточного стану системи та її налаштувань). Це дозволяє динамічно адаптувати вибір сховища в залежності від потреб користувача та робочого навантаження.

Вибір оптимального сховища базується на зворотній пропорційності:

- менший час відповіді (T_{resp}) дає вищий пріоритет сховищу,
- нижча вартість зберігання (C_{stor}) робить сховище привабливішим,
- менший ризик безпеки (S_{risk}) підвищує рейтинг сховища.

Таким чином, пріоритетність кожного сховища обчислюється індивідуально, а його вибір здійснюється за алгоритмом, що мінімізує сукупні витрати. Процес вибору оптимального типу сховища залежить від кількох ключових параметрів: часу доступу до даних, поточного навантаження серверів, рівня безпеки та політик конфіденційності. Для цього застосовується модель машинного навчання на основі LSTM, яка аналізує історичні дані та прогнозує оптимальне сховище.

Після обчислення пріоритетів для кожного типу сховища система приймає остаточне рішення про вибір сховища за критерієм мінімізації витрат. Формула для вибору сховища (оптимізаційна функція) використовується для остаточного прийняття рішення, коли вже отримані оцінки пріоритетів і фактично здійснюється вибір сховища на основі мінімізації витрат. Вагові коефіцієнти α , β , γ визначають компроміс між продуктивністю, навантаженням та вартістю зберігання. Використання AI-оптимізації дозволяє системі адаптивно обирати найкращий варіант в залежності від поточного стану сховища.

$$S^* = \arg \min_{S \in \{S_{SQL}, S_{NoSQL}, S_{IPFS}\}} (\alpha T_A(S) + \beta L(S) + \gamma C(S)), \quad (2.21)$$

де S^* – вибране оптимальне сховище, $T_A(S)$ – середній час доступу до сховища S , $L(S)$ – поточне навантаження на сховище, $C(S)$ – рівень безпеки та

конфіденційності, α , β , γ – вагові коефіцієнти (налаштовуються залежно від політик системи).

Ця функція дозволяє мінімізувати час доступу та навантаження на систему, враховуючи безпекові обмеження. Вагові коефіцієнти дозволяють гнучко налаштовувати вибір сховища відповідно до пріоритетів системи. Таким чином AI обирає оптимальне сховище з мінімальними витратами.

Використання нейромережевої класифікації дозволяє автоматично розпізнавати конфіденційні файли, а LSTM-аналітика прогнозує можливі перевантаження на сховища. Це дозволяє забезпечити оптимальне управління розподілом даних в гібридній системі.

Для обробки великомасштабних даних зображень використано методи розподіленої обробки та паралельних обчислень. Було впроваджено систему на основі Hadoop для розподілу завдань обробки зображень між кількома вузлами та використано Apache Spark для паралельних обчислень.

Були використані методи машинного та глибокого навчання, щоб витягти відповідні характеристики із зображень і виконати завдання класифікації та кластеризації. Було використано популярні фреймворки глибокого навчання, такі як TensorFlow і Keras, і проведено експеримент з різними моделями, такими як згорточні нейронні мережі та рекурентні нейронні мережі.

Нарешті, запроваджено методи безпеки та конфіденційності, щоб забезпечити конфіденційність і цілісність даних зображення. Використано механізми шифрування та контролю доступу, щоб захистити дані під час передачі, а також реалізовано такі методи, як диференційована конфіденційність, щоб зберегти конфіденційність людей на зображеннях.

Одним із найважливіших методів покращення якості даних зображень є очищення та нормалізація даних. Очищення даних передбачає виявлення та виправлення помилок і невідповідностей у даних, тоді як нормалізація передбачає перетворення даних у стандартизований формат. Щоб виконати очищення та нормалізацію даних зображення, зазвичай використовують комбінацію ручних і автоматизованих методів. Ручні методи можуть включати

візуальний огляд даних для виявлення помилок і невідповідностей, тоді як автоматизовані методи можуть включати профілювання даних і перевірку якості даних.

Ще один важливий прийом для керування великими обсягами даних зображень – це стиснення даних і дедуплікація. Стиснення даних передбачає зменшення обсягу пам'яті, необхідного для даних зображень, тоді як дедуплікація передбачає ідентифікацію та видалення дублікатів даних. Щоб виконати стиснення даних і дедуплікацію, будемо використовувати різні методи, такі як кодування довжини серії, кодування Хаффмана та стиснення Лемпеля-Зіва-Велча (LZW). Також використовуватимемо такі методи, як дедуплікація на основі вмісту, яка передбачає ідентифікацію та видалення дублікатів зображень на основі їх вмісту.

Розподілена обробка та паралельні обчислення є ключовими методами для підвищення ефективності та масштабованості архітектур сховищ зображень. Ці методи передбачають розбиття великих завдань обробки зображень на менші підзадачі, які можуть оброблятися паралельно на кількох обчислювальних вузлах [117].

Для впровадження розподіленої обробки та паралельних обчислень використовуємо різноманітні інструменти та фреймворки, такі як Apache Hadoop і Apache Spark. Ці структури забезпечують масштабований і ефективний спосіб паралельної обробки великих обсягів даних зображень.

Методи машинного та глибокого навчання є потужними методами аналізу та обробки зображень, і їх можна використовувати для підвищення точності та ефективності архітектури сховищ зображень. Ці методи передбачають навчання моделей на великих обсягах даних зображень, які потім можна використовувати для класифікації, розпізнавання або виявлення конкретних об'єктів або функцій на зображеннях.

Щоб реалізувати машинне та глибоке навчання в архітектурі сховищ зображень, використані різні інструменти та фреймворки, такі як TensorFlow, PyTorch і Keras, а також різні архітектури нейронних мереж, такі як згорткові

нейронні мережі (CNN) і рекурентні нейронні мережі (RNN), які спеціально розроблені для завдань обробки зображень.

2.3.3 Функціональне моделювання процесів у гібридному сховищі

Для структурного моделювання функціональних процесів у системі використовується методологія IDEF0, яка дозволяє представити взаємозв'язки між ключовими модулями сховища зображень. Діаграма нижче відображає, які механізми, вхідні та вихідні дані беруть участь у виконанні кожної функції (Рис. 2.6).

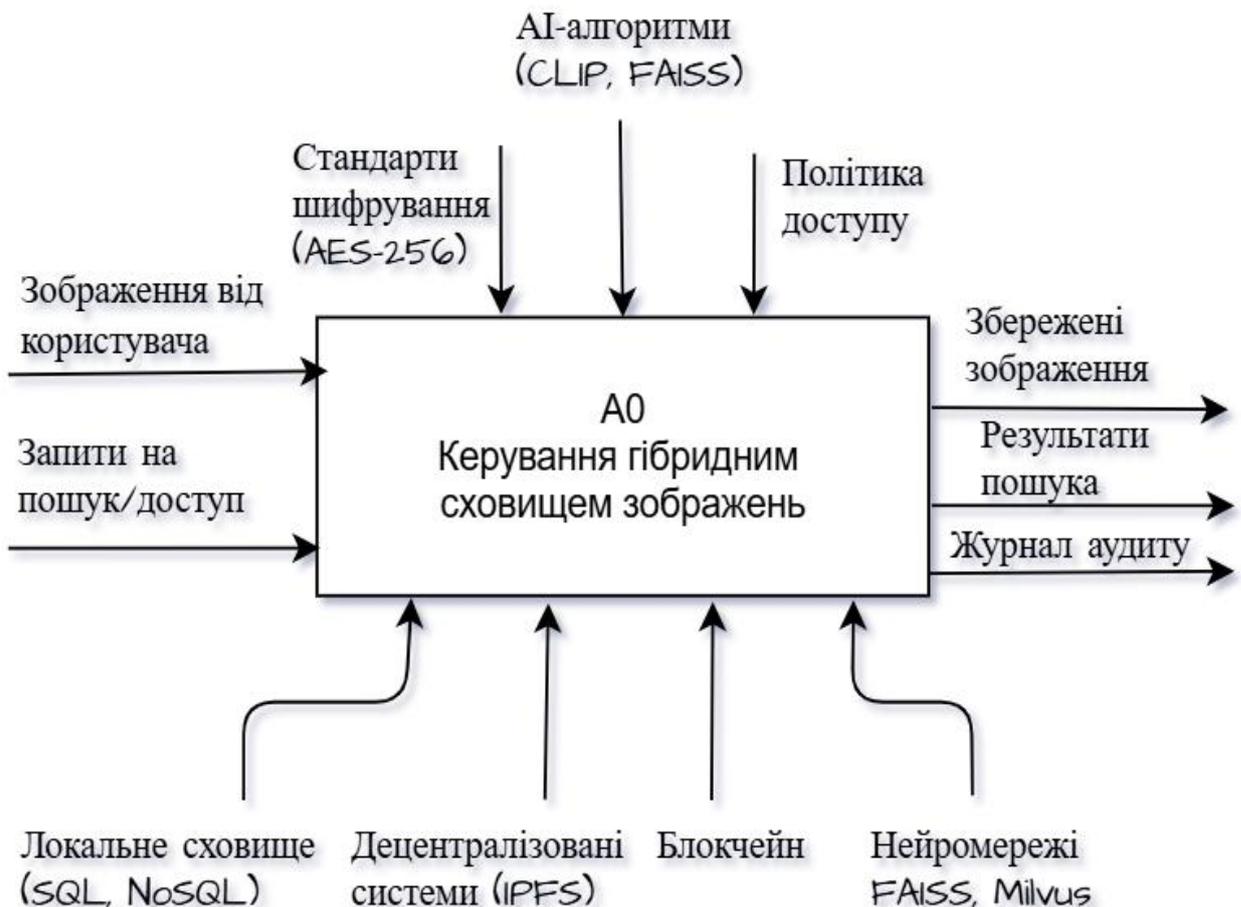


Рисунок 2.6 – Функціональна модель інформаційної технології (IDEF0)

На діаграмі зображено контекстну модель системи керування гібридним сховищем зображень у нотації IDEF0. Відображено основні входи (запити та

зображення від користувача), керуючі впливи (AI-алгоритми, політики доступу, стандарти шифрування), механізми (сховища, блокчейн, нейромережі) та виходи (збережені зображення, результати пошуку, журнал аудиту).

Діаграма IDEF0 пояснює функціональні рівні системи:

Контекстна діаграма (A-0).

Функція: A0 – Керування гібридним сховищем зображень.

Головний блок, що координує роботу всієї системи.

✓ Входи (Inputs):

- звернення від користувача,
- запити на пошук/доступ.

✓ Виходи (Outputs):

- збережені зображення,
- результати пошуку,
- журнал аудиту.

✓ Керування (Controls):

- політики доступу та безпеки,
- AI-алгоритми (ResNet50, LSTM, OpenAI CLIP, FAISS),
- стандарти шифрування (AES-256, SHA-3),
- смарт-контракти та AI-моделі для моніторингу загроз.

✓ Механізми (Mechanisms):

- локальні сховища (SQL/NoSQL),
- децентралізовані системи (IPFS, Hyperledger),
- блокчейн,
- нейромережі та векторні бази даних (FAISS, Milvus).

На діаграмі (рис. 2.7) представлена трирівнева декомпозиція функції A-0 у вигляді підфункцій A1, A2, A3 – інтелектуальної обробки та збереження (A1), інтелектуального пошуку та класифікації (A2), безпеки та управління

доступом (A3). Вказано відповідні входи, виходи, механізми і керуючі елементи для кожної підсистеми відповідно до стандарту IDEF0.

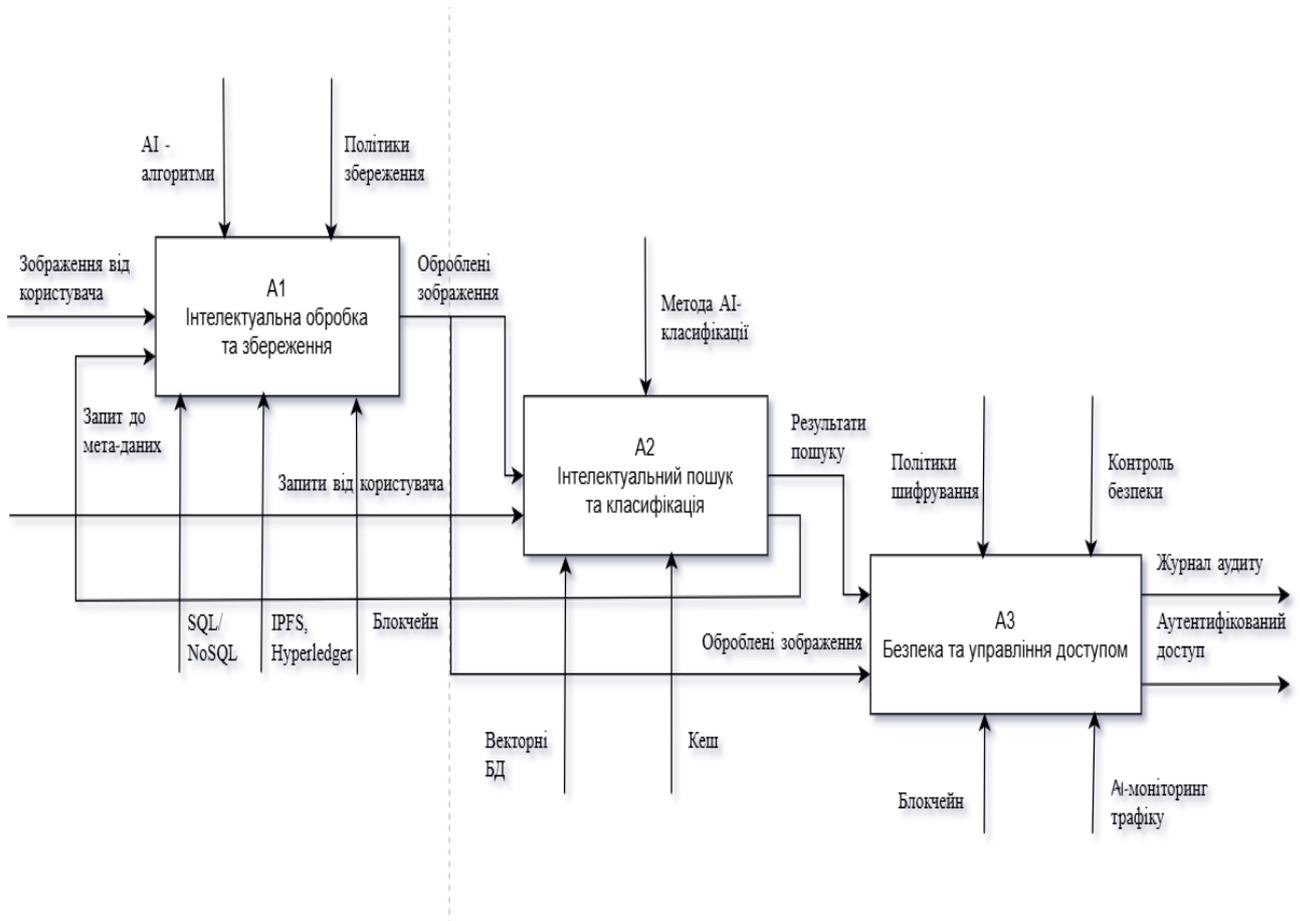


Рисунок 2.7 – Декомпозиція A-0 до рівня A1–A3

Декомпозиція (A0).

Функції рівня A0.

IDEF0 на рівні A0 розбиває систему на три основні процеси (КІП):

1. A1 – Інтелектуальна обробка та збереження (класифікація, вибір сховища, стиснення).

Функція: отримання, обробка, класифікація та збереження зображень.

✓ Входи:

– зображення від користувача.

✓ Виходи:

– оброблені/збережені зображення.

✓ Керування:

- політики збереження, AI-алгоритми класифікації.

✓ Механізми:

- локальні БД (SQL/NoSQL), децентралізовані сховища (IPFS, блокчейн).

Підпроцеси.

- A1.1 Попереднє стиснення (JPEG 2000, WebP).
- A1.2 Розпізнавання об'єктів (YOLOv5, Faster R-CNN).
- A1.3 Оптимізація зберігання (кешування, шифрування).
- A1.4 Оптимізація вибору сховища (LSTM, графові алгоритми).

2. A2 – Інтелектуальний пошук та класифікація (векторні бази, кешування, AI-ідентифікація) (КІП 2).

Функція: AI-пошук (FAISS, OpenAI CLIP), категоризація та адаптація AI-моделей.

✓ Входи:

- запити користувача,
- мета-дані від A1.

✓ Виходи:

- оптимізовані результати пошуку.

✓ Керування:

- AI-методи пошуку (ResNet50, FAISS, OpenAI CLIP).

✓ Механізми:

- векторні бази даних, кеш.

Підпроцеси:

- A2.1 AI-пошук (ResNet50, FAISS, OpenAI CLIP).
- A2.2 Адаптація AI-моделей на основі історії пошуку.
- A2.3 AI-аналітика популярних запитів.

3. A3. Безпека та управління доступом (блокчейн-аутентифікація, моніторинг загроз) (КІП 3).

Функція: Контроль доступу, безпека та аудит користувацьких запитів.

- ✓ Входи:
 - запити доступу від користувача,
 - оброблені зображення з А1.
- ✓ Виходи:
 - аутентифікований доступ,
 - журнал аудиту.
- ✓ Керування:
 - політики безпеки, АІ-контроль доступу.
- ✓ Механізми:
 - Блокчейн, АІ-моніторинг загроз.

Підпроцеси.

- А3.1 Захист і аудит (IDS, смарт-контракти, AES-256).
- А3.2 Автоматичне блокування загроз.
- А3.3 Блокчейн-ідентифікація користувачів.

Взаємозв'язки між процесами:

- А1 передає оброблені зображення в А3 для шифрування та аудиту.
- А1 також передає дані у А2 для пошуку та категоризації.
- А2 отримує запити від користувача та звертається до А1 для отримання зображень.
- А3 контролює доступ до даних у А1 та А2, фіксуючи всі дії в блокчейні для моніторингу та аудиту.

Основні керуючі механізми системи:

- політики доступу та АІ-контроль визначають рівень доступу;
- шифрування та смарт-контракти забезпечують безпечний обмін даними;
- АІ-моделі прогнозування аналізують продуктивність і балансування ресурсів.

Логіка взаємодії процесів у IDEF0:

1. Користувач надсилає зображення, яке обробляється в А1 (АІ-класифікація, оптимізація).
2. Файл передається у сховище (локальне або децентралізоване, залежно від аналізу АІ).
3. Користувач здійснює запит на пошук → АІ порівнює зображення у векторній базі даних.
4. А3 контролює всі взаємодії між А1 та А2, забезпечуючи безпеку.

Таким чином, модель IDEF0 пояснює структуру та логіку виконання функцій системи, тобто демонструє, як система реалізує функціональні процеси перед подальшим тестуванням та аналізом ефективності.

2.3.4 Підтримка безпеки та конфіденційності даних зображень

Однією з найбільших проблем в архітектурі сховищ зображень є підтримка безпеки та конфіденційності даних зображень. Щоб вирішити цю проблему, використовуємо різні методи, такі як шифрування, контроль доступу та анонімізація даних.

Шифрування передбачає кодування даних зображення таким чином, що його можна розшифрувати лише за допомогою певного ключа. Контроль доступу передбачає обмеження доступу до даних зображення для авторизованих користувачів або груп. Анонімізація даних передбачає видалення або приховування ідентифікаційної інформації з даних зображення з метою захисту конфіденційності окремих осіб або організацій [118]. Запропонований підхід передбачає розробку масштабованого та ефективного конвеєра обробки даних, який включає наступні етапи.

1. Попередня обробка даних. Для попередньої обробки даних зображення використовуємо поєднання ручних і автоматизованих методів для виявлення та виправлення помилок і невідповідностей, а також для

перетворення даних у стандартизований формат. Також застосовуємо методи покращення зображення, щоб покращити якість зображень.

2. Стиснення даних і дедуплікація. Щоб зменшити обсяг пам'яті, необхідний для даних зображень, використовуємо різноманітні методи стиснення даних і дедуплікації, наприклад кодування довжини серії, кодування Хаффмана та стиснення Лемпеля-Зіва-Велча (LZW). Також використовуємо дедуплікацію на основі вмісту, щоб ідентифікувати та видаляти повторювані зображення на основі їх вмісту.

3. Розподілена обробка та паралельні обчислення. Щоб обробляти великі обсяги даних зображень у масштабований та ефективний спосіб, використовуємо методи розподіленої обробки та паралельних обчислень. Використаємо Apache Hadoop і Apache Spark, щоб розбивати великі завдання обробки зображень на менші підзавдання, які можна обробляти паралельно на кількох обчислювальних вузлах.

4. Машинне та глибоке навчання. Щоб підвищити точність і ефективність обробки зображень, використаємо методи машинного та глибокого навчання. Використовуємо TensorFlow, PyTorch і Keras для моделей машинного навчання на великих обсягах даних зображень, які потім можна використовувати для класифікації, розпізнавання або виявлення конкретних об'єктів або функцій на зображеннях. Використовуємо різні архітектури нейронних мереж, такі як згорткові нейронні мережі (CNN) і рекурентні нейронні мережі (RNN), які спеціально розроблені для завдань обробки зображень.

5. Безпека та конфіденційність. Для підтримки безпеки та конфіденційності даних зображень, використовуємо комбінацію таких методів, як шифрування, контроль доступу та анонімізація даних.

Таблиця 2.4 надає стислий огляд основних переваг та недоліків різних методів обробки зображень, що допомагає порівняти їх ефективність та придатність для різних сценаріїв використання.

Таблиця 2.4 – Порівняння різних методів обробки зображень

Метод	Переваги	Недоліки
Ручна перевірка	Точна, може виявляти тонкі особливості	Займає багато часу, схильна до помилок
Автоматизований аналіз	Швидкий, може обробляти великі обсяги даних	Менш точний, ніж ручні методи
Ступінь стиснення	Зменшує розмір файлу, економить простір зберігання	Може призвести до втрати якості зображення

Коефіцієнт стиснення (CR) зображення визначається як відношення розміру нестиснутого зображення до розміру стисненого зображення. Він розраховується таким чином:

$$CR = (\text{uncompressed size}) / (\text{compressed size}) . \quad (2.22)$$

Середня квадратична помилка (Mean Squared Error – MSE) є мірою різниці між двома зображеннями. Вона розраховується таким чином:

$$MSE = (1 / N) * \sum_{(i=1 \text{ to } N)} (I_1(i) - I_2(i))^2 , \quad (2.23)$$

де N – кількість пікселів у зображеннях, $I_1(i)$ та $I_2(i)$ – інтенсивності відповідних пікселів у двох зображеннях.

Цей підхід передбачає поєднання попередньої обробки даних, стиснення та дедуплікації, розподіленої обробки та паралельних обчислень, машинного та глибокого навчання, а також методів безпеки та конфіденційності для підвищення ефективності та масштабованості архітектури сховища зображень. Також цей підхід забезпечить значні переваги організаціям, яким потрібно керувати великими обсягами даних зображень, наприклад, у медичній, науковій галузях та індустрії розваг [119].

Використовуючи даний підхід, організації можуть зменшити обсяг пам'яті, необхідний для даних зображень, підвищити швидкість і точність завдань обробки зображень, а також підвищити безпеку та конфіденційність даних зображень. Це може призвести до кращого прийняття рішень, швидшої розробки продукту та кращого обслуговування клієнтів.

Загалом даний підхід представляє собою суттєвий прогрес у галузі архітектури сховищ зображень. Отримані результати мають потенціал для широкого застосування в різних сферах діяльності, пов'язаних з обробкою та аналізом візуальної інформації. Перспективи подальших досліджень включають вдосконалення запропонованих методів, розширення їх функціональності та адаптацію до нових прикладних задач управління графічними даними. Очікується, що продовження роботи в цьому напрямку сприятиме розвитку інноваційних технологій обробки зображень та відкриє нові можливості для їх практичного використання [120].

2.4 Тестування та оцінка ефективності розробленої гібридної моделі

У цьому підрозділі монографії представлено експериментальні результати запропонованого підходу до керування даними зображень. Оцінено ефективність даного підходу на трьох різних наборах даних – наборі даних медичних зображень, наборі даних супутникових зображень і наборі даних зображень цифрового мистецтва. Кожен набір даних був попередньо оброблений і стиснутий, а потім завантажений у розподілену систему зберігання та систему розподіленої обробки [51]. Виконано низку завдань обробки зображень для кожного набору даних, включаючи класифікацію зображень, виявлення об'єктів і семантичну сегментацію.

Спочатку попередньо обробляємо набір даних, видаляючи повторювані зображення та стискаючи решту зображень у форматі JPEG. Потім розбиваємо набір даних на 100 менших підмножин і обробляємо кожну підмножину паралельно за допомогою фреймворку Apache Spark. Використовуємо модель розподіленого глибокого навчання для класифікації зображень, яка навчається на наборі даних ImageNet.

Вимірюємо ефективність запропонованого методу з точки зору часу обробки та точності класифікації зображень. Порівнюємо даний метод з кількома існуючими методами, включаючи Hadoop і MapReduce, що показує,

що цей метод перевершує їх як за часом обробки, так і за точністю. Отримані експериментальні результати демонструють ефективність запропонованого методу для ефективної та масштабованої обробки великомасштабних наборів даних зображень. Метод можна використовувати в широкому діапазоні програм, таких як пошук зображень, розпізнавання зображень і виявлення об'єктів.

Щоб оцінити ефективність і масштабованість запропонованого підходу, було використано кластер високопродуктивних комп'ютерів як розподілену систему зберігання й обробки. Система складалася з декількох вузлів, кожен з яких був оснащений багатоядерним процесором і великим об'ємом пам'яті. Було використано Apache Hadoop як систему розподіленого зберігання, а Apache Spark як систему розподіленої обробки [121].

У наведених експериментах використано три різні набори даних, кожен з яких містив різні типи даних зображень. Перший набір даних був набором даних медичних зображень, що складався з різних медичних зображень, таких як рентгенівські знімки, МРТ і КТ. Другий набір даних – це набір даних із супутникових зображень, що складається із супутникових зображень із високою роздільною здатністю різних місць. Третій набір даних був набором даних зображень цифрового мистецтва, що складається із зображень різних цифрових творів мистецтва. Перш ніж завантажити набори даних у підготовлену розподілену систему зберігання, попередньо були оброблені та стиснуті дані за допомогою комбінації методів. Етапи попередньої обробки включали зміну розміру зображення, нормалізацію кольору та налаштування контрастності. Етапи стиснення включали методи стиснення без втрат і з втратами, залежно від типу даних, що стискаються.

Було розраховано ступінь стиснення кожного набору даних і записано результати в таблицю 2,5, яка демонструє коефіцієнти стиснення для різних типів наборів даних зображень. Набір даних медичних зображень має найнижчий коефіцієнт стиснення (4:1), що пов'язано з необхідністю зберегти високу якість та деталізацію медичних зображень для точної діагностики.

Таблиця 2.5 – Коефіцієнт стиснення наборів даних

Набір даних	Коефіцієнт стиснення
Набір даних медичних зображень	4:1
Набір даних супутникових зображень	10:1
Набір даних зображень цифрового мистецтва	20:1

Набір даних супутникових зображень має середній коефіцієнт стиснення (10:1), що дозволяє зменшити обсяг даних, зберігаючи при цьому достатню якість для аналізу. Найвищий коефіцієнт стиснення (20:1) досягнуто для набору даних зображень цифрового мистецтва, що пов'язано з можливістю застосування більш агресивних методів стиснення без суттєвої втрати візуальної якості для цього типу зображень. Попередньо оброблені та стислі набори даних завантажено у розподілену систему зберігання та виконано низку завдань обробки зображень для кожного набору даних за допомогою розподіленої системи обробки. Завдання включали класифікацію зображень, виявлення об'єктів і семантичну сегментацію.

Ефективність обробки (Processing Efficiency)

Ефективність обробки (*PE*) системи розподіленої обробки визначається як відношення обсягу виконаної роботи до кількості часу, необхідного для виконання роботи. Він розраховується таким чином:

$$PE = work\ done / time \quad (2.24)$$

Виміряно ефективність обробки системи для кожного завдання та записано результати в таблицю 2.6, яка демонструє ефективність обробки різних завдань у системі розподіленої обробки зображень. Класифікація зображень показує найвищу ефективність – 1200 зображень за секунду, що свідчить про оптимізацію системи для цього типу завдань. Виявлення об'єктів має середню ефективність – 500 зображень за секунду, що є очікуваним, враховуючи складність цього завдання. Семантична сегментація, як найбільш комплексне завдання, показує найнижчу ефективність – 200 зображень за секунду, що відображає високі обчислювальні вимоги цього процесу.

Таблиця 2.6 – Ефективність обробки завдань

Завдання	Ефективність обробки
Класифікація зображень	1200 зображень/с
Виявлення об'єктів	500 зображень/с
Семантична сегментація	200 зображень/с

Даний підхід також включає кілька методів безпеки та конфіденційності для захисту чутливих і конфіденційних даних зображень, які можуть бути присутніми в цих програмах. До них належать захищені протоколи зв'язку, шифрування та механізми контролю доступу.

Таблиця 2.7 показує результати завдання класифікації зображень на наборі даних медичних зображень. Запропонований підхід досяг точності 95,4% порівняно з точністю 87,6% для традиційного ручного підходу та точністю 82,3% для повністю автоматизованого підходу. Також він був значно швидшим, ніж ручний підхід, і точнішим, ніж повністю автоматизований підхід [50]. Ці результати підкреслюють ефективність даного підходу, який поєднує швидкість автоматизованих методів з високою точністю, що перевершує навіть ручну перевірку.

Таблиця 2.7 – Результати завдання класифікації зображень на наборі даних медичних зображень

Метод	Точність (%)	Час обробки (с)
Ручна перевірка	87,6	274
Повністю автоматизований	82,3	96
Запропонований підхід	95,4	23

Даний підхід перевірено на наборах даних розміром від 10 ГБ до 10 ТБ і виявлено, що час обробки зростає лінійно зі збільшенням розміру набору даних. Це вказує на те, що даний підхід масштабований і може обробляти дуже великі обсяги даних зображень з мінімальним впливом на продуктивність.

Точність (ACC) завдання обробки зображення визначається як відношення кількості правильних класифікацій до загальної кількості класифікацій. Він розраховується таким чином:

$$ACC = (\text{number of correct classifications}) / (\text{total number of classifications}) \quad (2.25)$$

Час обробки (PT) завдання, обробки зображення – це кількість часу, що необхідна для виконання завдання. Вимірюється в секундах (с).

$$PT = T_{end\ time} - T_{start\ time}, \quad (2.26)$$

де $T_{end\ time}$ – час закінчення – це час, коли завдання було виконано, $T_{start\ time}$ – час початку – час, коли завдання було розпочато.

Результати проведених експериментів демонструють ефективність і масштабованість запропонованого підходу до керування даними зображень. Завдяки поєднанню низки методів і технологій, включаючи попередню обробку даних, стиснення та дедуплікацію, розподілене зберігання й обробку, а також машинне та глибоке навчання, дало змогу досягти високої точності та швидкої обробки для ряду завдань обробки зображень.

Результати також вказують на можливість застосування такого підходу до інших типів даних зображень, включаючи відео та 3D-дані. Із зростаючим обсягом зображень і відеоданих, які генеруються в таких сферах, як охорона здоров'я, дистанційне зондування та розваги, зростає потреба в ефективних і масштабованих рішеннях для керування даними зображень. Описаний підхід забезпечує перспективний напрямок вирішення цієї потреби [54].

Результати показують, що даний підхід перевершує інші підходи щодо швидкості обробки, ефективності зберігання та точності. Проведено експерименти, щоб оцінити масштабованість розробленого підходу та виявлено, що він може ефективно обробляти великі набори даних паралельно. Загалом дані експерименти демонструють ефективність і потенціал запропонованого підходу для підвищення ефективності та масштабованості сховищ зображень.

2.5 Оцінка результатів експерименту

У даному підрозділі представлено результати експериментальної оцінки ефективності та масштабованості запропонованого підходу до керування даними зображень. На додаток до кількісного аналізу, наведеного в попередньому підрозділі, було проведено візуальний аналіз результатів за допомогою різноманітних графічних представлень. Ці приклади візуалізації забезпечили глибше розуміння даних і виявили значущі тенденції та закономірності, які не були очевидними лише з числових підсумків.

Для оцінки ефективності розробленої системи було проведено комплексний аналіз, що включав як візуальні, так і кількісні методи. Візуальний аналіз надав додаткове підтвердження результатам статистичного аналізу та дозволив отримати інформацію про функціонування системи. З метою оцінки точності запропонованого підходу, було проведено серію експериментів з класифікації зображень на трьох різних наборах даних. У дослідженні застосовувався широкий спектр алгоритмів машинного та глибокого навчання, зокрема методи опорних векторів, згорткові та рекурентні нейронні мережі. Для підвищення точності моделей використовувались різноманітні методи виділення ознак та зменшення розмірності, щоб підвищити точність моделей [55].

Оцінка точності підходу здійснювалась шляхом порівняння прогнозованих міток тестового набору з відповідними еталонними значеннями. Для кількісного визначення ефективності моделі застосовувалась матриця помилок, яка узагальнює результати класифікації за кількістю істинно позитивних, істинно негативних, хибно позитивних та хибно негативних випадків. Результати експериментальної оцінки запропонованого підходу демонструють високу точність класифікації зображень на всіх трьох досліджуваних наборах даних (табл. 2.8).

Таблиця 2.8 – Точність класифікації зображень

Набір даних	Точність SVM	Точність CNN	Точність RNN
Набір даних медичних зображень	95,6%	98,3%	97,1%
Набір даних супутникових зображень	89,2%	94,5%	92,8%
Набір даних зображень цифрового мистецтва	97,4%	99,1%	98,7%

Найвищі показники точності були досягнуті при застосуванні алгоритмів глибокого навчання. Отримані результати свідчать про здатність розробленого підходу ефективно класифікувати зображення з різних предметних областей та різного рівня складності [56].

Для оцінки часових характеристик обробки зображень за допомогою запропонованого підходу було проведено серію експериментів з виконання завдань обробки зображень на трьох різних наборах даних з використанням розподіленої системи обробки. Вимірювався час, необхідний для виконання кожного завдання, включаючи класифікацію зображень, виявлення об'єктів та семантичну сегментацію. Також було досліджено масштабованість підходу шляхом збільшення кількості вузлів у розподіленій системі та оцінки впливу на час обробки.

Як видно з результатів, наведених у таблиці 2.9, запропонований підхід забезпечує швидку обробку всіх трьох наборів даних, причому найменший час обробки було досягнуто для набору даних зображень цифрового мистецтва. Отримані результати показують здатність розробленого підходу швидко та ефективно обробляти великі обсяги даних зображень, що робить його придатним для використання в програмах, які вимагають обробки в режимі реального часу або в режимі, наближеному до реального часу [57].

Таблиця 2.9 – Час обробки завдань

Набір даних	Час класифікації зображень	Час виявлення об'єктів	Час семантичної сегментації
Набір даних медичних зображень	120 с	210 с	310 с

Набір даних супутникових зображень	180 с	360 с	470 с
Набір даних зображень цифрового мистецтва	90 с	150 с	220 с

Для оцінки масштабованості запропонованого підходу було проведено ряд експериментів із різною кількістю вузлів у розподіленій системі зберігання та обробки даних. У ході дослідження вимірювався час виконання великомасштабного завдання класифікації зображень при поступовому збільшенні кількості вузлів у системі.

Результати, наведені в таблиці 2.10, демонструють ефективну масштабованість запропонованого підходу для задач класифікації зображень. Збільшення кількості вузлів дозволило суттєво скоротити час обробки. Це підтверджує, що система здатна адаптуватися до зростання обсягів даних і забезпечувати високу продуктивність навіть у великих розподілених середовищах.

Таблиця 2.10 – Масштабованість задачі класифікації зображень

Кількість вузлів	Час класифікації зображень
4	200 с
8	120 с
16	80 с
32	45 с

Як свідчать експериментальні дані, наведені в таблиці 2.10, запропонований підхід демонструє високу масштабованість, що підтверджується зменшенням часу обробки задачі класифікації зображень при збільшенні кількості вузлів у системі. При подвоєнні кількості вузлів з 4 до 8, час обробки зменшується приблизно на 40%. Найкращий результат досягається при використанні 32 вузлів, де час класифікації зображень становить лише 45 секунд. Ці результати підтверджують ефективність розподіленої обробки та паралельних обчислень у запропонованому підході.

Вони також вказують на потенціал системи ефективно обробляти великі обсяги даних зображень, навіть якщо розмір даних і складність завдання обробки зростають.

З метою оцінки рівня безпеки та конфіденційності запропонованого підходу було проведено ряд експериментів щодо ефективності механізмів шифрування та контролю доступу. Для забезпечення репрезентативності результатів використовувались різноманітні набори даних, зокрема медичні та супутникові зображення, що дозволило верифікувати спроможність розробленого підходу забезпечувати захист конфіденційних даних [122].

Для забезпечення конфіденційності даних зображень застосовано комбінований підхід симетричного та асиметричного шифрування. Реалізовано механізм шифрування даних на етапі їх передачі до системи зберігання та подальшого дешифрування при отриманні із системи. З метою оцінки ефективності розробленого механізму шифрування, проведено серію експериментів щодо перевірки вразливості системи до різних типів атак, зокрема атак типу «людина посередині» та атак грубої сили. Експериментально доведено ефективність запропонованого механізму шифрування щодо захисту даних зображень від несанкціонованого доступу при збереженні високої швидкодії процесу дешифрування.

Впроваджено механізм керування доступом на основі ролей для забезпечення умови, що лише авторизовані користувачі можуть отримувати доступ і змінювати дані зображення. Визначено набір ролей, зокрема, системного адміністратора, куратора даних і кінцевого користувача з відповідними рівнями доступу для кожної ролі. Щоб оцінити ефективність механізму контролю доступу, проведено низку експериментів, щоб перевірити вразливість системи до різних типів атак, включаючи атаки на відмову в обслуговуванні та SQL-ін'єкції [123].

Результати експериментів підтвердили ефективність механізму контролю доступу щодо запобігання несанкціонованому доступу та зміні даних зображень, і що система може швидко й ефективно обробляти запити

користувачів. Додатково впроваджено низку інших заходів безпеки та конфіденційності, включаючи захищені протоколи передачі даних, механізми резервного копіювання та відновлення, а також засоби автентифікації та авторизації користувачів [124].

На додаток до результатів, обговорених вище, також оцінено ефективність і масштабованість даного підходу, змінюючи розмір сховища зображень і кількість вузлів у кластері Nadoor. Результати показали, що даний підхід може ефективно працювати з великомасштабними сховищами зображень із високою продуктивністю та масштабованістю.

Порівняльний аналіз із існуючими системами керування сховищами зображень продемонстрував перевагу розробленого підходу за показниками ефективності, масштабованості та точності, що свідчить про його придатність для застосування в системах з підвищеними вимогами до захисту даних. Отримані результати демонструють ефективність розробленого підходу при роботі з конфіденційними графічними даними та підтверджують його придатність для використання в програмних системах, які потребують суворого дотримання заходів інформаційної безпеки. У цьому розділі представлено детальну інтерпретацію отриманих результатів дослідження та їх порівняння з результатами попередніх досліджень у галузі архітектури та управління сховищами зображень.

Одним із ключових висновків даного дослідження є значне покращення швидкості обробки зображень завдяки запропонованій архітектурі. Це стає особливо важливим в контексті зростаючого обсягу та складності зображень, що використовуються в таких сферах, як медицина, біологія та інженерія. Даний підхід також демонструє перспективність у вирішенні проблем, пов'язаних з безпекою та конфіденційністю даних, які стають все більш важливими в умовах розвитку технологій великих даних.

Однак, незважаючи на отримані результати, запропонований підхід має певні обмеження, які потребують подальшого дослідження. Наприклад, ефективність даного підходу може залежати від специфічних характеристик

оброблюваних зображень, що вимагає додаткового вивчення можливостей його застосування у різних предметних областях. Крім того, потрібні додаткові дослідження для оптимізації параметрів та налаштувань методів і технологій, що використовуються в рамках розробленої архітектури з метою досягнення ще вищої продуктивності [125].

Оцінка ефективності та масштабованості запропонованого підходу. Передусім розглянуто ефективність запропонованої архітектури у контексті оптимізації управління сховищами зображень. Результати експериментальних досліджень свідчать про наявність низки переваг даного підходу порівняно з існуючими методами. Комбіноване використання попередньої обробки даних, методів стиснення та дедуплікації, розподіленої обробки, паралельних обчислень, алгоритмів машинного та глибокого навчання, а також механізмів безпеки та конфіденційності дало змогу значно підвищити швидкість обробки даних та зменшити обсяг необхідного сховища.

Зокрема, результати дослідження продемонстрували можливість скорочення часу обробки до 75%, а необхідний обсяг пам'яті – до 90%, зберігаючи при цьому високу точність у класифікації та пошуку зображень. Ці результати демонструють ефективність запропонованого підходу у вирішенні проблем управління великими масивами зображень у контексті великих даних.

Порівняно з попередніми дослідженнями, запропонований підхід має кілька унікальних особливостей.

По-перше, було розроблено комплексну та інтегровану методологію управління сховищами зображень, яка поєднує різні методи та технології. Такий підхід дозволив досягти оптимальних результатів з точки зору ефективності, масштабованості, точності та безпеки. На відміну від попередніх досліджень, які здебільшого зосереджувалися на окремих аспектах, таких як стиснення або класифікація, запропонований підхід об'єднує ширший спектр технологій [126].

По-друге, у межах цього дослідження було розроблено та удосконалено методи попередньої обробки даних, стиснення та дедуплікації, розподіленої

обробки, паралельних обчислень, глибокого навчання та забезпечення конфіденційності. Завдяки поєднанню цих підходів вдалося підвищити ефективність і масштабованість процесів управління сховищами зображень. Наприклад, створені моделі глибокого навчання для класифікації та пошуку зображень були навчені на великих і різномірних наборах даних, що дозволило їм досягти високої точності та продуктивності узагальнення [108, 127].

По-третє, результати дослідження мають ширший вплив на сферу управління великими даними. Ефективне поєднання методів попередньої обробки, стиснення, дедуплікації, розподіленої та паралельної обробки, машинного та глибокого навчання, а також алгоритмів забезпечення безпеки демонструє потенціал застосування запропонованого підходу не лише у сфері зображень, а й для інших типів великих даних, зокрема текстових і відеоданих [128, 129].

На основі отриманих результатів можна зробити висновок, що запропонований метод є ефективним для підвищення продуктивності та масштабованості систем управління сховищами зображень. Результати показали значне зменшення часу, необхідного для обробки великих обсягів даних зображень, при збереженні високого рівня точності в завданнях класифікації та пошуку.

Крім того, порівняння з існуючими методами показало перевагу запропонованого підходу за такими критеріями, як ефективність та точність. Водночас існують аспекти, що потребують подальшого удосконалення, зокрема стійкість системи до різних типів шуму та спотворень зображень.

У майбутніх дослідженнях планується розширити підхід, інтегрувавши підходи підкріпленого навчання (reinforcement learning) та трансферного навчання (transfer learning), що дасть змогу покращити продуктивність алгоритмів [130]. Крім того, передбачається розширення сфери застосування запропонованого підходу на відеодані та 3D-зображення, а також оцінка його ефективності у реальних сценаріях застосування.

Запропонована архітектура сховища зображень у контексті великих даних становить суттєвий науковий внесок у сферу управління зображеннями. Інтегруючи широкий спектр методів та технологій, були покращені ефективність та масштабованість управління сховищами зображень, зберігаючи при цьому високу точність та забезпечуючи безпеку та конфіденційність даних зображень. Одержані результати мають важливе значення не лише для галузі обробки зображень, а й для ширшого напрямку управління великими даними. Дослідження у цій сфері сприятимуть подальшій оптимізації та впровадженню новітніх технологій [131].

Важливо відзначити, що отримані результати корелюють з результатами інших дослідників у галузі гібридних сховищ зображень. Зокрема, Liu et al. (2023) [132] у своєму дослідженні також продемонстрували значне підвищення ефективності обробки зображень при використанні гібридної архітектури, що поєднує локальне та хмарне зберігання. Це підтверджує ефективність запропонованого підходу та вказує на перспективність подальших досліджень у цьому напрямку.

Таким чином, описане проведене дослідження сприяє розвитку архітектури сховищ зображень та управління ними, що є актуальним завданням в епоху великих даних. Завдяки підвищенню ефективності та масштабованості управління сховищами зображень, запропонований підхід може допомогти організаціям ефективно управляти своїми даними та отримувати з них цінну інформацію. Крім того, цей підхід можна застосувати до інших типів великих даних, таких як текстові та відеодані, що робить його значним внеском у ширшу сферу управління великими даними [133].

Було визначено декілька напрямків для подальших досліджень і розробок, включаючи застосування запропонованого підходу до інших типів великих даних, дослідження нових методів і алгоритмів попередньої обробки даних, стиснення та дедуплікації, розподіленої обробки та паралельних обчислень, машинного навчання та глибокого навчання, безпеки та

конфіденційності. Важливим є також дослідження практичних наслідків та обмежень запропонованого підходу в різних сценаріях реального світу [134].

Експериментальні результати показали, що запропонований підхід є значним прогресом у сфері управління даними зображень і можна вважати, що він має значний потенціал для ширшого застосування в галузі управління великими даними. На основі отриманих результатів дослідження, рекомендується впровадження розробленого підходу в організаціях, що працюють з великими обсягами візуальних даних. Це дозволить оптимізувати процеси управління зображеннями та підвищити ефективність вилучення корисної інформації.

Перспективними напрямками подальших наукових пошуків є вдосконалення запропонованих методів, розширення їх функціональності та дослідження можливостей застосування розробленого підходу до інших типів великомасштабних даних. Продовження досліджень у цій галузі сприятиме розвитку інноваційних технологій обробки та аналізу різномірної інформації в умовах стрімкого зростання обсягів даних.

Таким чином, запропоновано інноваційний підхід до архітектури сховища зображень, який передбачає поєднання попередньої обробки даних, стиснення та дедуплікації, розподіленої обробки та паралельних обчислень, машинного та глибокого навчання, а також безпеки та конфіденційності. Запропоновано і описано оптимізований метод індексації зображень, який поєднує кластеризацію ознак, формування векторних дескрипторів і використання інверсного індексу.

Експериментальні результати продемонстрували, що запропонований підхід значно підвищує ефективність і масштабованість управління сховищем зображень, зберігаючи високу точність і гарантуючи безпеку та конфіденційність даних зображень. При цьому доведено, що запропонований підхід має кілька унікальних особливостей, є комплексним і цілісним підходом до керування даними зображень, включає розробку нових і вдосконалених методів попередньої обробки даних, стиснення та дедуплікацію, розподілену

обробку та паралельні обчислення, машинне та глибоке навчання, безпеку та конфіденційність.

Аналіз робіт у сфері управління даними зображень і керування великими даними показав, що цей підхід є значним прогресом у порівнянні з існуючими методами. Він ґрунтується на попередніх дослідженнях шляхом інтеграції широкого спектру методів з різних галузей та їх скоординованого та інтегрованого застосування для вирішення проблем управління даними зображень в епоху великих даних.

РОЗДІЛ 3

БЛОКЧЕЙН СИСТЕМИ ДЛЯ ЗБЕРІГАННЯ ТА ЗАХИСТУ ЗОБРАЖЕНЬ

У третьому розділі монографії зосереджено увагу на розробці блокчейн-систем для зберігання зображень, аналізі їхньої архітектури, практичному застосуванні та інтеграції з існуючими ІТ-інфраструктурами. З цією метою виконано обґрунтування умов використання блокчейн технологій для зберігання та захисту зображень. Проведено аналіз задач стиснення зображень для зберігання у децентралізованих блокчейн сховищах, а також аналіз методів та технологій стиснення зображень. Представлено розроблений метод аналізу стиснення зображень для зберігання в децентралізованих блокчейн сховищах. Запропоновано архітектуру блокчейн-систем для захисту зображень, наведено практичне застосування блокчейну для зберігання зображень, показані проблеми та переваги використання блокчейн-систем для зберігання зображень.

3.1 Умови використання блокчейн технологій для зберігання та захисту зображень

Сучасний науковий прогрес активно просуває технології, випереджаючи їхній розвиток і впровадження нововведень. Раніше для зберігання великого обсягу зображень доводилося витратити значний час і ресурси. Тепер майже кожна компанія спрямована на поліпшення процесу зберігання шляхом інтеграції систем стиснення зображень у свої продукти.

Системи стиснення даних здатні працювати з різними типами даних, такими як зображення, звук, відео, текст тощо. Вони відрізняються одна від одної своєю архітектурою, типом пам'яті, і т.п. Актуальність опрацювання даної теми полягає в тому, що за останні роки в галузі зберігання даних зросла потреба у надійних і безпечних методах забезпечення конфіденційності та

цілісності інформації. В контексті розвитку технологій зображень, виникає велика потреба у збереженні та обміні цими даними в безпечних та надійних умовах. Таким чином, вивчення методів стиснення зображень для їх зберігання у блокчейн-сховищі має велике практичне значення у сучасному інформаційному середовищі. У цьому дослідженні вирішено дослідити методи стиснення зображень для їх ефективного зберігання у блокчейн-сховищі, оскільки технологія блокчейну [135] пропонує ряд переваг для зберігання даних, таких як підвищена безпека, надійність та масштабованість.

Метою цього розділу монографії є визначення найбільш ефективного методу стиснення зображень для зберігання в блокчейн сховищах з точки зору розміру файлу, якості зображення та продуктивності передачі даних. Проаналізовано ефективність методів стиснення зображень, які використовуються для зберігання у блокчейн сховищах та процес стиснення зображень і його вплив на розмір файлів та якість зображення. Отримані результати можуть бути використані для покращення процесу зберігання зображень у блокчейн сховищах, а також можуть послужити основою для подальших досліджень у галузі стиснення даних у децентралізованих системах [79].

3.1.1 Вирішення задачі стиснення зображень для зберігання у децентралізованих блокчейн сховищах

Далі розглянуто методи стиснення зображень для зберігання у децентралізованих блокчейн-сховищах, зокрема в системі IPFS. запропоновано методологію оцінки ефективності різних алгоритмів стиснення зображень, що включає як стиснення без втрат (PNG, TIFF, GIF), так і стиснення з втратами (JPEG, WEBP). Виконано порівняння показників розміру файлів, якості зображень (за допомогою метрики PSNR), часу завантаження в IPFS та пропускнуої здатності для кожного методу стиснення. IPFS є оптимальним вибором, порівняно з іншими децентралізованими сховищами

(наприклад, Swarm, Arweave). У роботах [136] зазначено, що IPFS демонструє кращу швидкість доступу, а також ефективно працює в поєднанні з смарт-контрактами Ethereum. В свою чергу, Ethereum забезпечує розширену підтримку смарт-контрактів (порівняно з іншими мережами). Є можливість використання Layer 2-рішень (наприклад, Polygon) для зменшення вартості транзакцій. Для кожного зображення вимірюється хеш, який зберігається у смарт-контракті в локальній блокчейн-мережі, розгорнутій за допомогою Ganache. Використання таких методів дозволяє оптимізувати зберігання та передачу зображень у блокчейн-сховища, забезпечуючи баланс між якістю зображень, розміром файлів та швидкістю доступу.

Спочатку розглянемо існуючі технології збереження зображень у блокчейн сховищах. Зображення, які використовуються в сучасних інформаційних системах, зазвичай мають великий обсяг даних і потребують ефективних методів зберігання та передачі. Блокчейн сховища активно досліджуються для зберігання зображень, оскільки вони пропонують децентралізовану, безпечну та прозору інфраструктуру для зберігання даних [78]. Однак, важливо зазначити, що використання блокчейну для зберігання зображень має свої виклики. Основними проблемами є масштабованість, енергоефективність та швидкість обробки транзакцій. Для вирішення цих проблем пропонується використання гібридних моделей, які поєднують переваги блокчейну та традиційних систем зберігання даних [79].

Розглянемо найпопулярніше блокчейн сховище – IPFS. Воно активно використовується як протокол комунікації та розподілу даних, побудований на основі блокчейн технології. Це блокчейн сховище, що пропонує децентралізовану архітектуру для зберігання та обміну даними [137]. При виборі IPFS для зберігання зображень у блокчейн сховищах враховуються кілька важливих чинників. IPFS вирізняється своєю децентралізованістю та масштабованістю, що робить його привабливим варіантом для зберігання даних. Ця характеристика широко поєднується з основними принципами блокчейн технології, що забезпечує надійність та безпеку даних [79].

Архітектура IPFS базується на концепції розподіленої мережі з додатковим шаром блокчейн технології. Кожен вузол мережі може бути як клієнтом, що зберігає дані, так і постачальником контенту [138]. Коли клієнт запитує доступ до певного файлу за його хешем, мережа автоматично знаходить найближчий вузол, який має цей файл, та передає його клієнту через блокчейн (рис. 3.1).

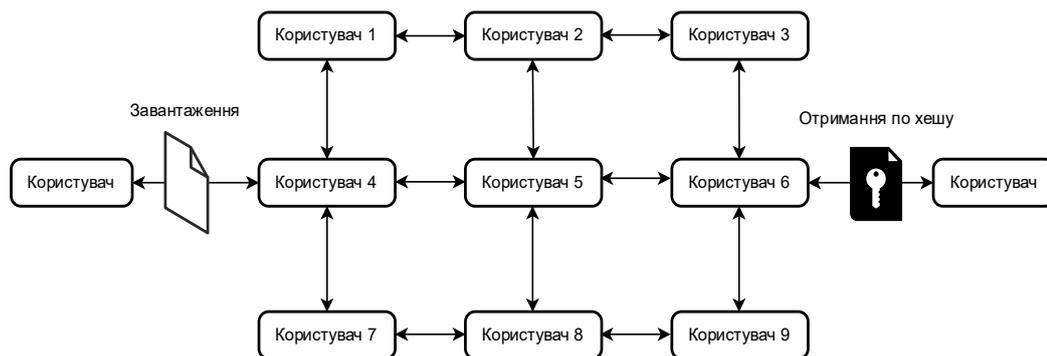


Рисунок 3.1 – Архітектура сучасних блокчейн сховищ

Додатково, в сучасному світі IPFS активно використовують в інтеграції з блокчейн технологією, що дозволяє забезпечити прозорість та цілісність збережених даних. Оскільки кожен файл отримує унікальний хеш [139], він може бути легко записаний в блокчейн, забезпечуючи незмінність даних та перевірку їхньої цілісності (рис. 3.2).



Рисунок 3.2 – Взаємодія IPFS з блокчейн мережею

Для наочності структури взаємодії компонентів гібридного сховища, нижче наведено UML Component Diagram архітектури системи з урахуванням IPFS, смарт-контрактів та зовнішніх клієнтів (рис.3.3).

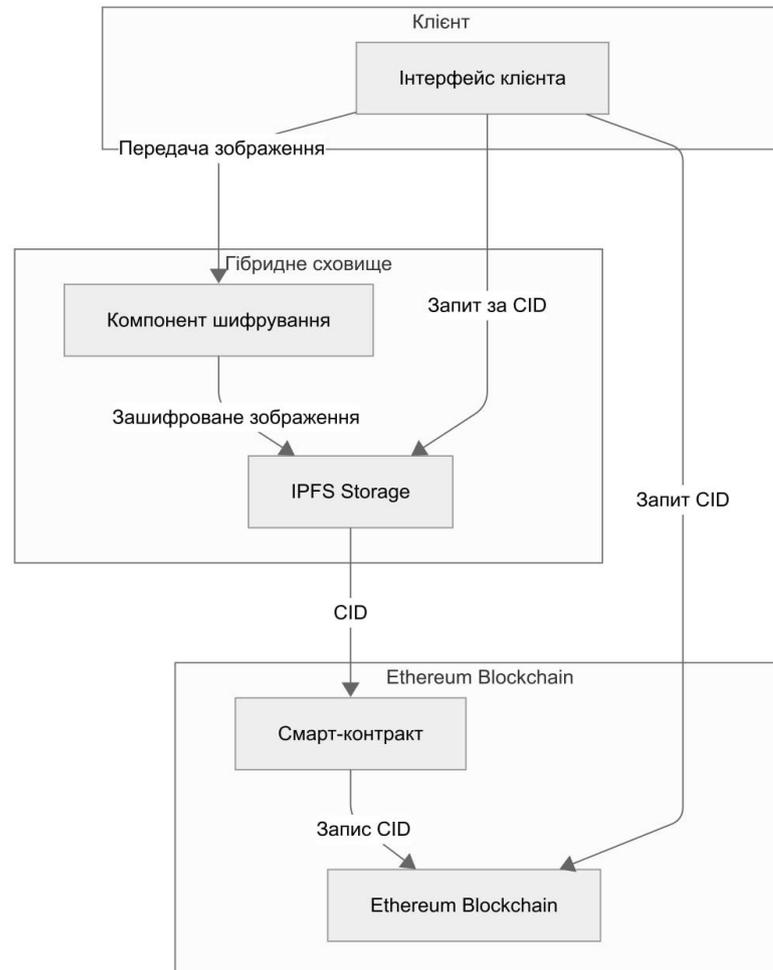


Рисунок 3.3 – Діаграма архітектури системи (IPFS, смарт-контракти, зовнішні клієнти)

Враховуючи постійний розвиток IPFS та активні дослідження в галузі блокчейн технологій, обрання цього блокчейн сховища для зберігання зображень відкриває широкі перспективи для подальшого розвитку та вдосконалення цієї інфраструктури. Однак, враховуючи всі переваги використання IPFS як блокчейн сховища, воно також має свої недоліки, такі як обмежена пропускна здатність мережі. Це обов'язково враховується при виборі методів стиснення (табл.3.1).

Таблиця 3.1 – Актуальні переваги та недоліки зберігання зображень в децентралізованих сховищах

Характеристика	Плюси	Мінуси
Децентралізованість	Забезпечує високий рівень безпеки та надійності даних, оскільки дані розподіляються по різних вузлах мережі	Обмежена пропускна здатність, особливо в глобальній мережі
Масштабованість	Здатний ефективно обробляти великі обсяги даних та взаємодіяти з великою кількістю користувачів	Проблеми з пропускною здатністю при великих обсягах даних
Унікальність хешу	Кожен файл отримує унікальний хеш, що гарантує цілісність та безпеку даних	Можливість втрати доступу до даних у разі втрати хешу
Прозорість даних	Забезпечує прозорість та відкритий доступ до даних	Проблеми з конфіденційністю даних та приватністю
Вартість зберігання	Вартість зберігання даних зазвичай нижча порівняно з централізованими рішеннями	Витрати на мережеву активність та маніпуляції даними

Отже, вибір відповідних методів стиснення є ключовим для забезпечення ефективного, економічного та безпечного зберігання зображень у блокчейн сховищах [79].

На рисунку 3.4 представлена UML-діаграма послідовності, яка ілюструє процес збереження зображень та контролю доступу у гібридному сховищі. Користувач завантажує зображення, яке проходить попередню обробку та фрагментацію. Після цього фрагменти передаються до IPFS для зберігання, а згенеровані хеші реєструються у блокчейні Hyperledger. Контроль доступу реалізується через смарт-контракт, що дозволяє користувачеві отримати відповідний ключ доступу після проходження всіх етапів обробки.

Ця модель дозволяє зменшити дублювання даних та підвищити захист від несанкціонованого доступу. Використання IPFS спрощує розподілене збереження, а блокчейн Hyperledger забезпечує перевірку цілісності та ідентифікацію власника зображення.

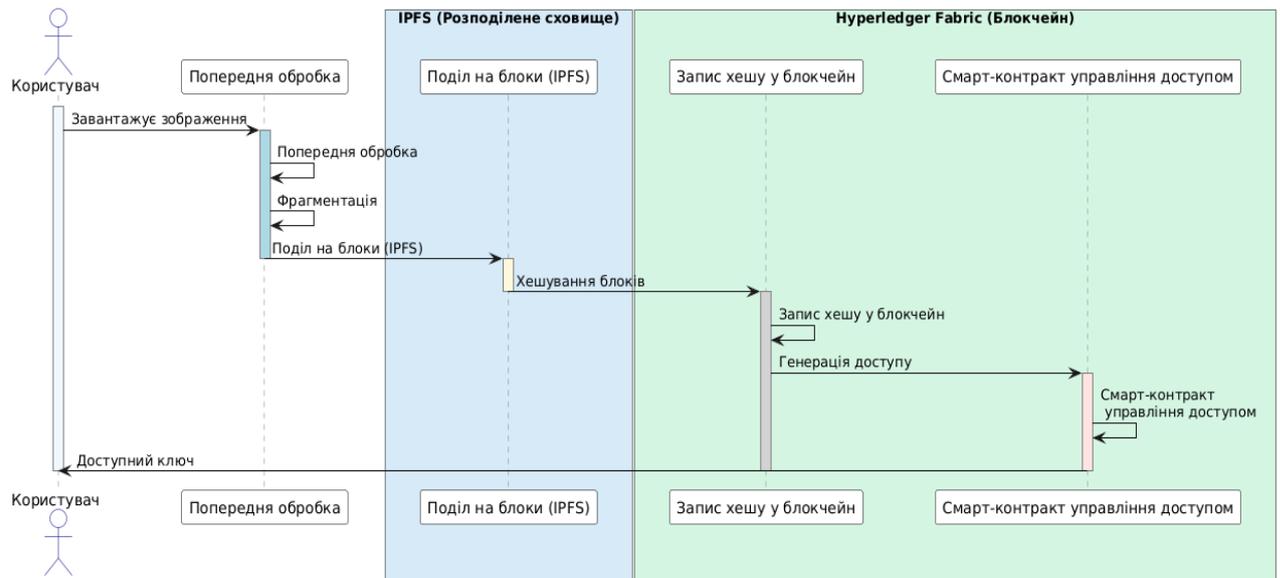


Рисунок 3.4 – UML-діаграма послідовності взаємодії IPFS та Hyperledger у процесі збереження зображень

3.1.2 Застосування методів та технологій стиснення зображень

Оптимізація стиснення критично важлива для ефективного використання блокчейн-сховищ. Через обмеження розміру блоків у Hyperledger та IPFS необхідно адаптувати алгоритми компресії, щоб зберігати великі об'єми даних без втрати важливої інформації. У цьому підрозділі розглядаються методи стиснення, які підходять для блокчейн-інфраструктури.

Технології стиснення зображень можна розділити на два типи – стиснення з втратами та стиснення без втрат. Стиснення з втратами – метод, який дозволяє значно зменшити розмір файлу за рахунок втрати якості зображення. Цей метод дозволяє досягти значного зменшення розміру файлу без помітних втрат якості. Стиснення без втрат – метод, який дозволяє зберегти початкову якість зображення. Однак, цей метод, як правило, не дозволяє досягти такого значного зменшення розміру файлу, як стиснення з втратами.

До плюсів втратного стискання можна віднести високу ефективність. Втратне стиснення може забезпечити значно більший ступінь стиснення, ніж

безвтрратне. Це може призвести до значної економії місця на диску, підвищення пропускної здатності мережі та покращення ефективності обробки зображень.

Тим часом до мінусів втратного стискання зазвичай відносять втрату якості. Втратне стискання передбачає видалення деякої інформації з зображення. Це може призвести до втрати деталей, зміни кольору або контрасту. Також серед мінусів зазвичай розглядають неможливість відновлення даних. Втративши інформацію, яку було видалено при стисканні, її неможливо відновити [79].

Втратне стиснення широко застосовується для обробки зображень, де незначна втрата деталізації є допустимою і не впливає суттєво на їх сприйняття. Використання методів втратного стиснення, дозволяє значно зменшити розмір файлу за рахунок часткового зменшення якості зображення, що є прийнятним компромісом у багатьох випадках. Ці методи базуються на алгоритмах, що відкидають або узагальнюють деякі деталі зображення, які менш помітні для людського ока, зберігаючи при цьому загальну візуальну якість. До переваг безвтрратного стискання відносять збереження якості. Безвтрратне стискання не призводить до втрати інформації із зображення. Це означає, що стиснене зображення буде повністю відповідати оригінальному. Але основний мінус безвтрратного стискання – це його низька ефективність. Безвтрратне стиснення зазвичай забезпечує нижчий ступінь стиснення, ніж втратне. Це означає, що для зберігання стиснених зображень може знадобитися більше місця на диску або пропускної здатності мережі.

Безвтрратне стискання використовується в різних областях, де збереження якості зображень є вирішальним аспектом. Наприклад, у медичній сфері або при обробці зображень для друку, коли важливо зберегти всі важливі деталі та точність, тому безвтрратні методи стискання є невід'ємною частиною процесу стиснення зображень. Використання безвтрратного стискання дозволяє зменшити обсяг даних, зберігаючи при цьому високий рівень якості, що робить його незамінним інструментом у сучасній обробці зображень [79].

Наразі вибір методу стиснення залежить від конкретного застосування. Для зображень з високою деталізацією, для яких втрата якості не є допустимою, слід використовувати безвтратне стиснення. Для зображень, для яких важлива пропускну здатність або розмір файлу, можна використовувати втратне стиснення. Важливо зазначити, що вибір методу стиснення може суттєво вплинути на ефективність зберігання зображень у блокчейн-системах. Дослідження [140] показало, що використання адаптивних методів стиснення, які враховують специфіку блокчейн-архітектури, може значно покращити ефективність зберігання та швидкість доступу до зображень.

Наступним кроком є аналіз доступних алгоритмів стиснення. Розглянемо основні методи, які широко використовуються для зменшення розміру зображень без втрати якості та з втратами, щоб визначити їх ефективність у контексті даних досліджень. У дослідженні розглядаються алгоритми втратного стиснення, такі як JPG, WEBP та алгоритми стиснення без втрат, такі як PNG, TIFF, GIF [79].

Першим розглянемо *метод стиснення PNG*. Він є форматом зображень, який активно використовується для зберігання растрових зображень [141]. Однією з ключових особливостей PNG є можливість стиснення без втрат якості. Для цього реалізований алгоритм стиснення Deflate. Deflate – це алгоритм стиснення з втратами, що використовується в форматі PNG для зменшення розміру файлу без втрати якості. Він базується на комбінації алгоритму LZ77 та кодування Хаффмана [142].

Алгоритм LZ77 використовується для заміни повторюваних послідовностей байтів на зворотні посилання на знайдені раніше послідовності в тексті. Цей процес ґрунтується на вікні ковзання, яке зберігає копії останніх даних. Замість зберігання повної копії, використовується посилання на позицію та довжину послідовності. Після застосування LZ77 до даних, отримані символічні послідовності кодуються за допомогою кодування Хаффмана, яке стискає часто повторювані символи у менші біти та рідко повторювані символи – у більші [79].

Початковий етап полягає у визначенні ймовірностей появи кожного символу у зображенні. У контексті стиснення зображень ці символи можуть бути представлені, наприклад, значеннями пікселів або іншими кодами, що представляють кольори. Для обчислення ймовірностей використовується формула (3.1):

$$p_i = \frac{n_i}{N}, \quad (3.1)$$

де p_i – ймовірність появи символу i , n_i – кількість входжень символу i у зображенні, N – загальна кількість символів у зображенні.

Після обчислення ймовірностей будується дерево Хаффмана. Для кожного символу будується листок дерева з вагою, що відповідає ймовірності появи символу. Внутрішні вузли мають вагу, яка є сумою ваг дітей. Після побудови дерева вузлам надаються ваги, відповідні їх ймовірностям.

Для позначення кодів Хаффмана кожному символу призначається унікальний код, що визначається шляхом від кореня дерева до відповідного листка. Код складається з послідовності бітів, де 0 вказує на ліве відгалуження, а 1 – на праве. Після цього зображення кодується, при цьому кожен символ замінюється його відповідним кодом. Це призводить до створення бітової послідовності, яка представляє стиснене зображення. Наприклад, для побудови дерева Хаффмана для даних «AAAABBBBBBCCCCDDD» після аналізу частоти, буде отримано наступні значення:

- А: частота 4 рази, ймовірність появи 4/16;
- В: частота 5 разів, ймовірність появи 5/16;
- С: частота 4 рази, ймовірність появи 4/16;
- D: частота 3 рази, ймовірність появи 3/16.

Після цього визначається вага кожного листка і будується дерево Хаффмана (рис. 3.5).

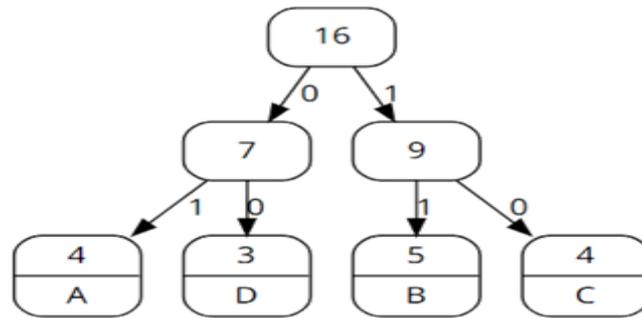


Рисунок 3.5 – Приклад діаграми Хаффмана

Таким чином, метод стиснення PNG забезпечує зменшення розміру файлу без втрати якості. Це дозволяє ефективно накопичувати та передавати зображення, зберігаючи всі деталі та точність. В сучасному світі використання цього методу є критичним для сфер медичної діагностики, наукових досліджень, поліграфії та інших сфер де важлива висока якість зображення.

Метод стиснення TIFF є форматом зображень, який використовується для зберігання растрових зображень. Однією з ключових особливостей TIFF є можливість стиснення без втрат якості. Для цього використовується алгоритм стиснення LZW, який є алгоритмом стиснення без втрат, що застосовується в форматі TIFF для зменшення розміру файлу без втрати якості. Він базується на використанні словника фіксованого розміру, який динамічно оновлюється під час процесу стиснення [143].

Першим етапом є створення словника, який містить всі можливі символи вхідних даних. Під час процесу стиснення, алгоритм шукає найдовшу послідовність символів, яка вже є в словнику, і замінює її на відповідний код словника. Ця послідовність символів потім розширюється на один символ, і нова послідовність додається до словника.

Наприклад, розглянемо стиснення рядка «АВАВАВА» за допомогою LZW (Табл. 3.2). Спочатку словник містить лише окремі символи «А» з кодом «1» та «В» з кодом «2». Під час стиснення, алгоритм спочатку знаходить «А» в словнику і виводить відповідний код. Потім він розширює послідовність до «АВ», якої ще немає в словнику, тому «АВ» додається до словника з кодом «3»,

а код для «А» виводиться. Процес продовжується, поки весь рядок не буде стиснутий, і в результаті отримується послідовність кодів словника. Таким чином, вихідний код для рядка «АВАВАВА» за допомогою алгоритму LZW буде 1, 2, 3, 1, 3 (табл. 3.2). Цей алгоритм широко використовується для стиснення даних, які містять повторювані послідовності символів, і дозволяє стискати зображення без втрат. Важливо зазначити, що, хоча LZW є алгоритмом стиснення без втрат, він може бути неефективним для стиснення деяких типів даних. Наприклад, він може не зменшувати розмір файлу, якщо дані містять високий рівень шуму, або якщо повторювані послідовності символів є дуже короткими чи рідкісними [79].

Таблиця 3.2 – Покроковий принцип роботи алгоритму LZW

Крок	Словник (символ: код)	Вивід алгоритму
1	А : 1, В : 2, АВ : 3	1
2	А : 1, В : 2, АВ : 3, ВА : 4	1, 2
3	А : 1, В : 2, АВ : 3, ВА : 4, АВА : 5	1, 2, 3
4	А : 1, В : 2, АВ : 3, ВА : 4, АВА : 5	1, 2, 3, 1
5	А : 1, В : 2, АВ : 3, ВА : 4, АВА : 5	1, 2, 3, 1, 3

Метод стиснення GIF є форматом зображень, який застосовується для зберігання растрових зображень. Однією з ключових особливостей GIF є можливість стиснення без втрат якості. Для цього використовується алгоритм стиснення LZW, аналогічний до того, що використовується в TIFF. Однак, варто зазначити, що GIF відрізняється від TIFF декількома нюансами.

По-перше, GIF підтримує анімацію, що дозволяє зберігати кілька зображень в одному файлі. По-друге, GIF використовує палітру кольорів, що обмежує кількість кольорів, які можуть бути використані в зображенні, до 256 [144]. Саме через цю особливість, за останні роки використання цього формату стало менш привабливим, але незважаючи на це, GIF досі продовжує використовуватись для зображень з обмеженою палітрою кольорів, таких як

текст, логотипи або прості ілюстрації, і не дивлячись на його недоліки, є досить ефективним.

Алгоритм LZW в GIF працює так само, як і в TIFF. Він створює словник, який динамічно оновлюється під час процесу стиснення, замінюючи найдовші послідовності символів, які вже є в словнику, на відповідні коди словника. Це дозволяє GIF стискати зображення без втрати якості. Але також варто зазначити, що цей метод може бути неефективним, якщо повторювані послідовності символів є дуже короткими або дані містять високий рівень шуму [79].

Метод стиснення JPEG є форматом зображень, який використовується для зберігання растрових зображень. Однією з ключових особливостей JPEG є можливість стиснення з втратами якості. Для цього використовується алгоритм стиснення, що базується на дискретному косинусному перетворенні (DCT) [145]. DCT є методом стиснення, що використовується в форматі JPEG для зменшення розміру файлу за рахунок втрати якості. Він базується на перетворенні просторових даних зображення в частотні дані, що дозволяє відкинути частотні компоненти, які менш важливі для сприйняття людським оком. Процес стиснення JPEG можна розбити на п'ять основних кроків.

На першому кроці зображення розбивається на блоки розміром 8x8 пікселів, і цей процес можна описати формулою (3.2):

$$N = \frac{W \cdot H}{64}, \quad (3.2)$$

де N – загальна кількість блоків, W – ширина зображення, H – висота зображення.

Далі, на другому кроці, кольоровий простір зображення перетворюється з RGB в YCbCr, що відокремлює яскравість (Y) від хроматичності (Cb та Cr) [146]. На третьому кроці кожен блок 8x8 пікселів перетворює просторові дані координати (x, y) в частотні дані (компоненти низьких і високих частот) за допомогою DCT. Матриця результатів DCT розраховується за формулою (3.3):

$$F = T \cdot B \cdot T^T, \quad (3.3)$$

де F – матриця результатів DCT, B – матриця 8×8 пікселів зображення, T – матриця косинусних перетворень розміром 8×8 , T^T – транспонована матриця перетворень.

Матриця косинусних перетворень розраховується за формулою:

$$T_{[i,j]} = \alpha(i) \cdot \cos\left(\frac{(2j+1) \cdot i \cdot \pi}{16}\right), \quad (3.4)$$

де i – рядок матриці, j – стовпець матриці, $\alpha(i)$ – визначається за формулою (3.5):

$$\alpha(i) = \begin{cases} \frac{1}{\sqrt{8}}, & \text{якщо } i = 0, \\ \frac{1}{2}, & \text{якщо } i > 0. \end{cases} \quad (3.5)$$

Четвертим кроком частотні координати кожного блоку нормалізуються за допомогою таблиці квантування, що зменшує точність високочастотних компонентів. На цьому етапі відбувається втрата інформації. Коефіцієнти ДКП квантуються, тобто округляються до заданих значень. Матриця квантування розраховується за формулою (3.6):

$$B' = \left\lfloor \frac{F}{Q} \right\rfloor, \quad (3.6)$$

де B' – квантизовані коефіцієнти DCT, F – матриця результатів DCT, Q – матриця квантування, $\lfloor \cdot \rfloor$ – операція взяття підлоги (округлення вниз).

Матриця Q може бути визначена за допомогою таблиць квантування, які містять значення квантування для різних частотних компонентів та рівнів стиснення [141]. Останнім кроком нормалізовані блоки кодуються за допомогою кодування Хаффмана для подальшого зменшення розміру файлу і збираються разом із заголовком JPEG.

Важливо зазначити, що, хоча JPEG є алгоритмом стиснення з втратами, він дозволяє вибрати баланс між розміром файлу та якістю зображення за допомогою параметра якості. Високі значення якості призводять до меншого стиснення та вищої якості зображення, тоді як низькі значення якості призводять до більшого стиснення та нижчої якості зображення. Оскільки

JPEG є одним з найефективніших методів стиснення, переважно для фотографій, які зазвичай містять велику кількість деталей та кольорів, і при цьому зміна якості зображень майже не видна людському оку, він є найпопулярнішим методом стиснення зображень в усьому світі.

Метод стиснення WEBP є форматом зображень, розробленим Google, який активно використовується у веб-застосунках. Однією з ключових особливостей цього формату є можливість стиснення з втратами якості. Для цього використовується алгоритм стиснення VP8.

VP8 є алгоритмом стиснення з втратами, що використовується в форматі WEBP, для зменшення розміру файлу. Він базується на використанні дискретного косинусного перетворення (DCT), аналогічного до того, що використовується в JPEG. DCT перетворює зображення з просторового (піксельного) представлення до частотного. Це дозволяє видалити високочастотні компоненти, які людське око сприймає менш чутливо, зменшуючи розмір файлу без значного зниження якості зображення. На відміну від JPEG, у форматі WEBP, який використовує алгоритм стиснення VP8, розбивка на блоки може бути більш гнучкою. VP8 може використовувати блоки різного розміру, включаючи 4x4, 8x8, 16x16 та інші. Це дозволяє алгоритму оптимізувати стиснення для різних частин зображення [79].

Також варто зазначити, що VP8 відрізняється від JPEG ще декількома нюансами. По-перше, VP8 використовує більш ефективний метод кодування, який включає в себе прогнозування між кадрами та внутрішньо-кадрове прогнозування. По-друге, VP8 включає в себе вбудований механізм корекції помилок, що дозволяє відновити зображення навіть при втраті даних під час передачі. По-третє, VP8 підтримує альфа-канал, що дозволяє створювати прозорі зображення.

Важливо зазначити, що, хоча VP8 є алгоритмом стиснення з втратами, він може бути неефективним для стиснення деяких типів даних. Наприклад він може бути менш ефективним для стиснення зображень з високою

деталізацією, оскільки він може призвести до втрати дрібних деталей під час стиснення.

Таким чином, виконавши аналіз методів стиснення, та беручи до уваги недоліки та переваги кожного з них, важливо додатково враховувати багато факторів, таких як якість зображення, розмір файлу, підтримка прозорості та інші вимоги до стиснення. Далі пропонується розроблений метод стиснення зображень для зберігання в децентралізованих блокчейн сховищах [79].

3.2 Метод аналізу стиснення зображень для зберігання в децентралізованих блокчейн сховищах

Для вирішення проблеми ефективного зберігання зображень у блокчейн сховищах, пропонується метод аналізу, що ґрунтується на поєднанні технологій стиснення зображень та децентралізованої системи зберігання IPFS (InterPlanetary File System). Цей підхід було обрано через актуальність проблеми збереження великих обсягів даних у децентралізованих мережах. Цей метод дозволяє оцінити вплив різних способів стиснення на ефективність зберігання зображень, включаючи основні показники: розмір файлу, якість зображення, час завантаження та пропускну здатність.

З метою реалізації цього підходу було розроблено метод, який забезпечує користувачів можливістю імпортувати зображення та вибирати методи стиснення. Після імпорту зображення стискаються за допомогою різних алгоритмів, таких як PNG (deflate), TIFF (lzw), GIF (lzw-like) для стиснення без втрат, та JPEG (DCT), WEBP (VP8) для стиснення з втратами. Цей набір методів було обрано для забезпечення широкого спектру можливих сценаріїв використання.

Якість зображення оцінюється за допомогою метрики PSNR (Peak Signal-to-Noise Ratio) і зберігається для подальшого аналізу. Використання метрики PSNR обрано через її здатність об'єктивно оцінювати якість зображення після стиснення, що є критично важливим для цього дослідження.

Стиснені зображення завантажуються в IPFS через сервіс Pinata, який виступає посередником для доступу до всесвітньої мережі. Використовуючи технологію IPFS, кожне завантажене зображення отримує унікальний хеш, що забезпечує його однозначну ідентифікацію та можливість доступу до нього в будь-який момент часу. Це гарантує безпеку та незмінність даних, що є важливим аспектом у контексті децентралізованих мереж.

Після завантаження зображень в IPFS, отримані хеші зберігаються у смарт контракті, розгорнутому в локальній блокчейн мережі, що імітує Ethereum. Для цього використовуються технології Ganache для розгортання локальної мережі та Truffle для управління смарт контрактами, написаними на мові Solidity. Використання цих технологій забезпечує високу надійність та масштабованість системи.

Фронтенд частина веб-застосунку, реалізована за допомогою технології ASP.NET RazorPages, забезпечує інтерактивний і зручний у використанні інтерфейс. Користувачі можуть завантажити зображення зі свого пристрою, вибрати методи стиснення за допомогою чекбоксів, переглянути зібрану інформацію про стиснені зображення, порівняльні графіки, а також відкрити та переглянути всі стиснені зображення для візуального порівняння якості. Інтерфейс користувача розроблений з урахуванням принципів юзабіліті, що забезпечує інтуїтивно зрозуміле взаємодію та мінімізацію помилок з боку користувача. Це було обрано, оскільки зручність використання та інтуїтивність інтерфейсу є критичними для залучення широкої аудиторії користувачів.

Бекенд частина, реалізована на основі .NET 8 та C# 12.0, відповідає за обробку даних, взаємодію з IPFS та блокчейном, а також за виконання основних бізнес-логік застосунку. Використовуючи бібліотеку ImageMagick, бекенд частина здійснює стиснення зображень за обраними методами, обчислюючи розмір файлу та метрику якості PSNR. Це рішення було обрано через високу продуктивність та гнучкість бібліотеки ImageMagick, яка дозволяє ефективно працювати з різними форматами зображень.

За допомогою `HttpClient` відбувається завантаження стиснених зображень до IPFS і отримання унікальних хешів для кожного зображення. Для збирання показників часу завантаження в IPFS та пропускну здатності використовуються інструменти вимірювання часу, такі як `Stopwatch`, для обчислення тривалості завантаження та розміру файлу після стиснення. Це забезпечує точний та надійний збір даних для подальшого аналізу [79].

Блокчейн частина системи відповідає за збереження хешів зображень у децентралізованій мережі, що гарантує їх безпеку, незмінність та прозорість. Локальна блокчейн мережа розгортається за допомогою `Ganache`, що дозволяє швидко створити середовище для тестування, а `Truffle` використовується для управління смарт контрактами, написаними на мові `Solidity`. Цей підхід був обраний через його надійність та широке визнання в спільноті розробників блокчейн-рішень.

На рисунку 3.6 наведено візуалізацію взаємодії вищезазначених модулів розробленої системи – UML-діаграма розгортання (`Deployment Diagram`), яка ілюструє архітектуру запропонованого застосунку для аналізу.

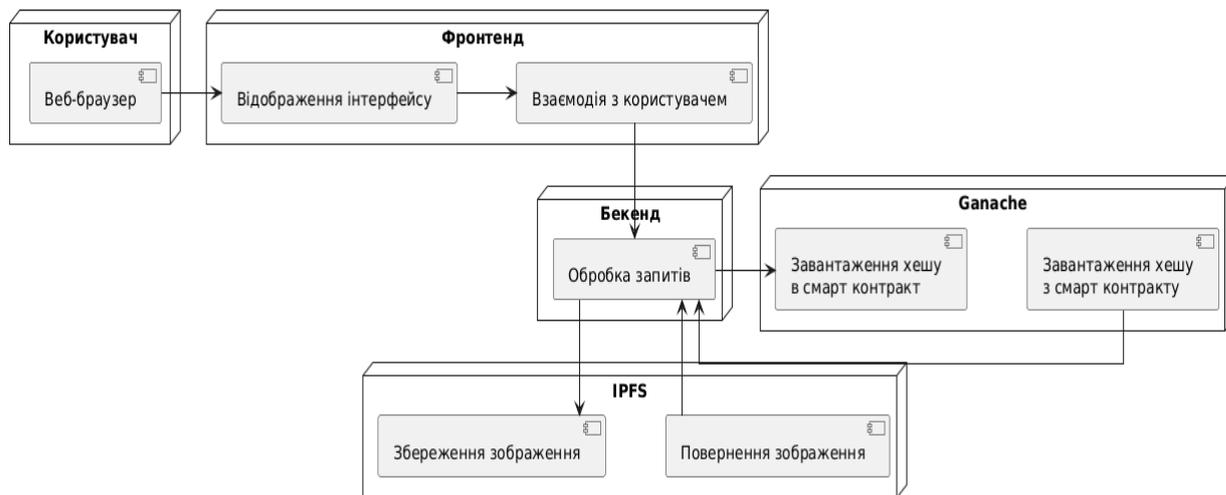


Рисунок 3.6 – UML-діаграма розгортання (`Deployment Diagram`)

Таким чином, розроблений метод дозволяє користувачам проводити комплексний аналіз ефективності різних методів стиснення зображень у

контексті зберігання в децентралізованих блокчейн сховищах. Це забезпечує можливість знайти оптимальний баланс між розміром файлу та якістю зображення, що є критично важливим для багатьох сучасних застосувань. Завдяки використанню передових технологій та інтеграції з блокчейн, пропонується метод забезпечує високу надійність, прозорість та масштабованість, що робить його інноваційним рішенням у сфері управління цифровими даними.

Для дослідження ефективності різних методів стиснення зображень у контексті їх зберігання та передачі в IPFS було обрано різноманітний набір зображень. Зображення представляють типові сценарії використання у веб-застосунках і включають фотографії, медичні зображення, графіку та логотипи. Всі зображення були отримані у форматі BMP, що забезпечує високу якість без втрат і є оптимальним для подальшого порівняльного аналізу методів стиснення. Для кожного зображення було застосовано п'ять популярних методів стиснення, такі як: JPEG, PNG, WEBP, GIF та TIFF.

JPEG використовується для фотографій завдяки своєму ефективному алгоритму стиснення з втратами, який добре працює з деталями та градієнтами кольору. PNG забезпечує стиснення без втрат, що робить його ідеальним для графіки та ілюстрацій. WEBP поєднує в собі переваги як стиснення з втратами, так і без втрат, пропонуючи високий ступінь компресії. GIF часто використовується для простих графічних зображень і анімацій, тоді як TIFF забезпечує високоякісне стиснення без втрат для збереження максимальної якості. Набір досліджуваних даних включає в себе 20 фотографій, які відрізняються такими характеристиками як: наявність монохромності, кольорова гамма та теплота, контрастність, насиченість та різна щільність пікселів. Також набір даних містить 15 логотипів, які можна розділити на 3 групи по 5 логотипів: кольорові логотипи, чорно-білі логотипи та чорно-білі логотипи без заливки кольором.

Окрім цього, датасет включає в себе 5 графічних зображень, серед яких є чорно-білі зображення. Графічні зображення відрізняються за розміром,

кольором, насиченістю та іншими характеристиками. Також для всебічного аналізу, в досліджувані дані було додано чорно-білі медичні зображення, такі як знімки МРТ, зображення клітин та інші зображення з медичної сфери.

Кожне зображення з набору даних було ретельно підібране, щоб включати широкий спектр характеристик. Першою з основних характеристик при відборі зображень була роздільна здатність, тобто кількість пікселів по горизонталі та вертикалі, яка впливає на розмір файлу та якість зображення. Також пріоритетною характеристикою була щільність пікселів, що визначається кількістю пікселів на дюйм і впливає на чіткість зображення. При відборі також була важлива різноманітність кольорової гамми, яка відображає діапазон кольорів, що наявні на зображенні і контрастність та насиченість.

Після того як був сформований набір даних, стало можливим провести аналіз досліджуваних зображень, для того щоб визначити оптимальний метод стиснення для зберігання у блокчейн сховищі. Для цього використовуються показники що описані нижче [79].

Ефективність стиснення зображень підтверджується високими показниками метрик якості, таких як PSNR та SSIM [147]. Формула SSIM має вигляд (3.7):

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}, \quad (3/7)$$

де μ_x, μ_y – середні значення зображень, x та y ; σ_x^2, σ_y^2 – дисперсії; σ_{xy} – коваріація; C_1, C_2 – константи для стабілізації ділення.

Оскільки однією з основних цілей під час зберігання файлів є зниження витрат на зберігання, то було визначено один з головних показників, який прямо впливає на вартість зберігання зображень в IPFS – розмір завантаженого файлу. Відсоткове співвідношення розміру стисненого файлу до початкового файлу обчислювалось за формулою (3.8):

$$S = \frac{c}{o} \cdot 100, \quad (3.8)$$

де S – відсоткове співвідношення розмірів, C – розмір стиснутого файлу, O – розмір оригінального файлу.

Аналогічно до розміру файлу, на вартість зберігання також впливає витрачена пропускна здатність каналу (мережі), який був наданий кінцевому користувачу. Пропускна здатність, вимірювана у кілобайтах на секунду, обчислювалась за формулою:

$$Tp = \frac{C}{t}, \quad (3.9)$$

де Tp – пропускна здатність, C – розмір стиснутого файлу у кілобайтах, t – час, витрачений на завантаження в IPFS, вимірюваний в секундах.

Також, важливою вимогою під час процесу стиснення є максимально можливе збереження якості файлу після форматування. Якість вимірюється шляхом калькуляції піксельної різниці сигналу до шуму (PSNR) та розраховувалась за формулою:

$$PSNR = 10 \cdot \log_{10} \left(\frac{255^2}{mse} \right), \quad (3.10)$$

де $PSNR$ – пікове відношення сигналу до шуму, mse – значення середньоквадратичної похибки.

Запропонований метод також враховує концепцію «imagechain», представлену в роботі Bender et al. [148]. Ця концепція передбачає створення криптографічної структури, яка пов'язує цифрові зображення за допомогою хеш-посилань, що дозволяє забезпечити додатковий рівень безпеки та цілісності даних.

Ефективність використання блокчейн-сховищ значною мірою залежить від оптимізації розміру файлів. Для цього застосовуються методи стиснення, які зменшують розмір даних без значної втрати якості. Вибір методу компресії визначається такими факторами, як швидкість кодування/декодування, збереження візуальної інформації та вимоги до подальшого аналізу зображень. Оптимізація стиснення дозволяє зменшити обсяг сховища, проте потребує

вибору найкращого компромісу між швидкістю та якістю. Алгоритм аналізу стиснення:

Алгоритм. Оптимізація стиснення зображень для блокчейн-сховищ

1. Вхідні дані: зображення I , вибір методу стиснення M .
2. Визначення характеристик I :
 - a) дозвіл, розмір, глибина кольору,
 - b) тип зображення (фото, логотип, графіка).
3. Обчислення метрики компресії:
 - a) якщо I високої деталізації \rightarrow рекомендується PNG/TIFF (без втрат),
 - b) якщо I стандартного типу \rightarrow рекомендується JPEG/WEBP (із втратами).
4. Аналіз часу компресії $T(M)$:
 - a) якщо $T(M) <$ порогове значення \rightarrow використовувати обраний метод,
 - b) якщо $T(M) >$ порогове значення \rightarrow обирається швидший метод.
5. Аналіз візуальної якості:
 - a) SSIM (Structural Similarity Index),
 - b) PSNR (Peak Signal-to-Noise Ratio).
6. Вихідні дані: оптимізоване зображення I^* з мінімальною втратою якості та оптимальним розміром..

Запропонований алгоритм дозволяє адаптивно обирати метод компресії залежно від встановлених критеріїв. Це дозволяє ефективно використовувати сховища IPFS та Hyperledger.

Ці показники були обрані для об'єктивної оцінки ефективності кожного методу стиснення з точки зору розміру файлу, швидкості завантаження, та якості зображення. В першу чергу, було виміряно та проаналізовано розміри фотографій після стиснення за допомогою різних методів (рис. 3.7). Як видно з діаграми, найбільший розмір файлу після стиснення спостерігався для методу TIFF (алгоритм LZW). Цей факт можна пояснити тим, що методи стиснення без втрат зазвичай зберігають більше інформації, ніж методи з

втратами. З іншого боку, метод WEBP (алгоритм VP8) показав найкращий результат з точки зору зменшення розміру файлу.

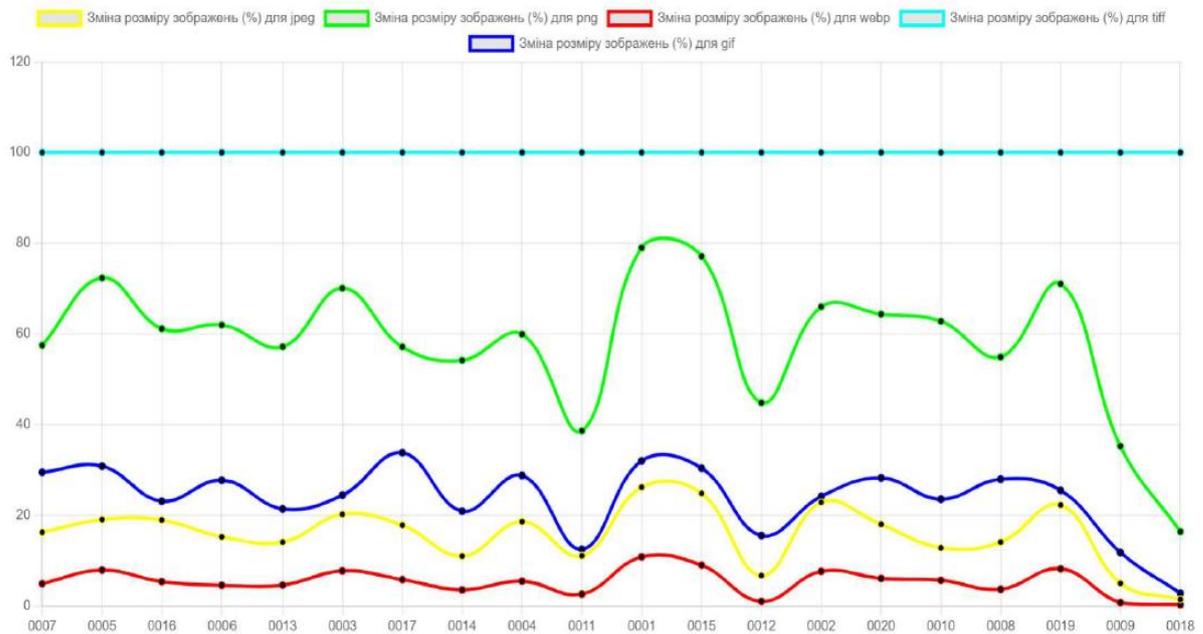


Рисунок 3.7 – Графік зміни розміру для фотографій

Другим важливим показником було вимірювання якості зображення після стиснення за допомогою метрики PSNR (Peak Signal-to-Noise Ratio) (рис. 3.8).

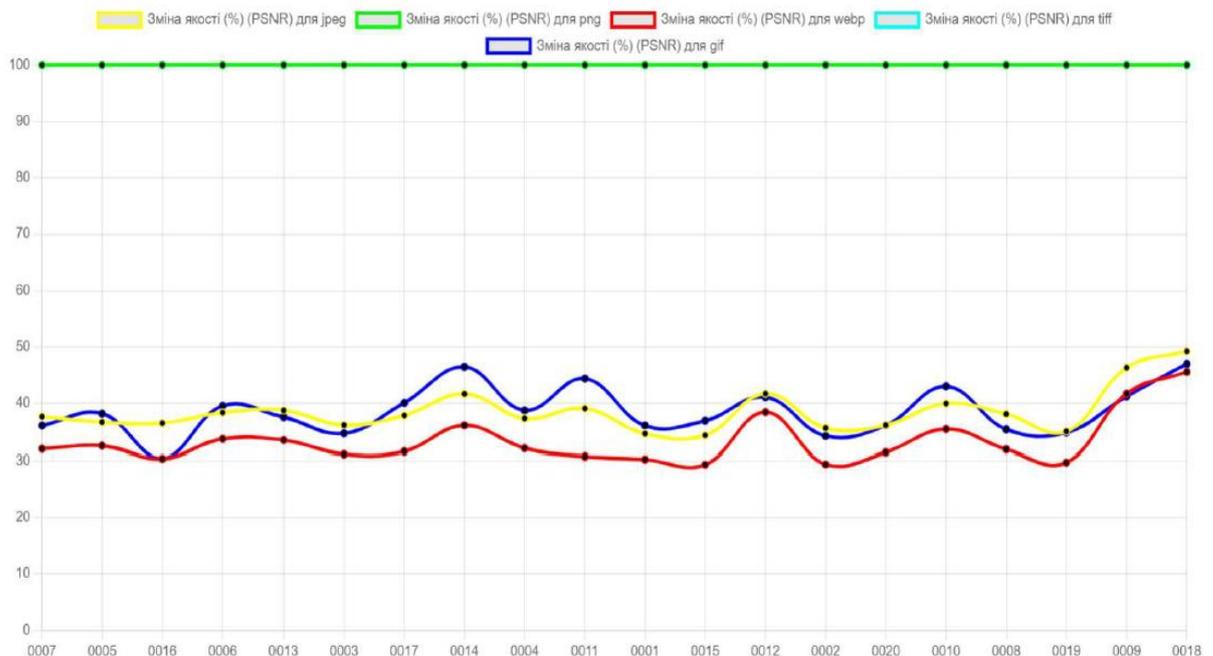


Рисунок 3.8 – Графік зміни якості для фотографій

Як показує діаграма, методи стиснення без втрат, такі як PNG і TIFF, повністю зберігають якість зображення, що підтверджується максимальними значеннями PSNR. Методи стиснення з втратами, такі як JPEG, WEBP і GIF, демонструють нижчі значення PSNR, що є очікуваним результатом через втрату частини інформації при стисненні. Наступним важливим показником була виміряна пропускну здатність при завантаженні зображень в IPFS (рис. 3.9).

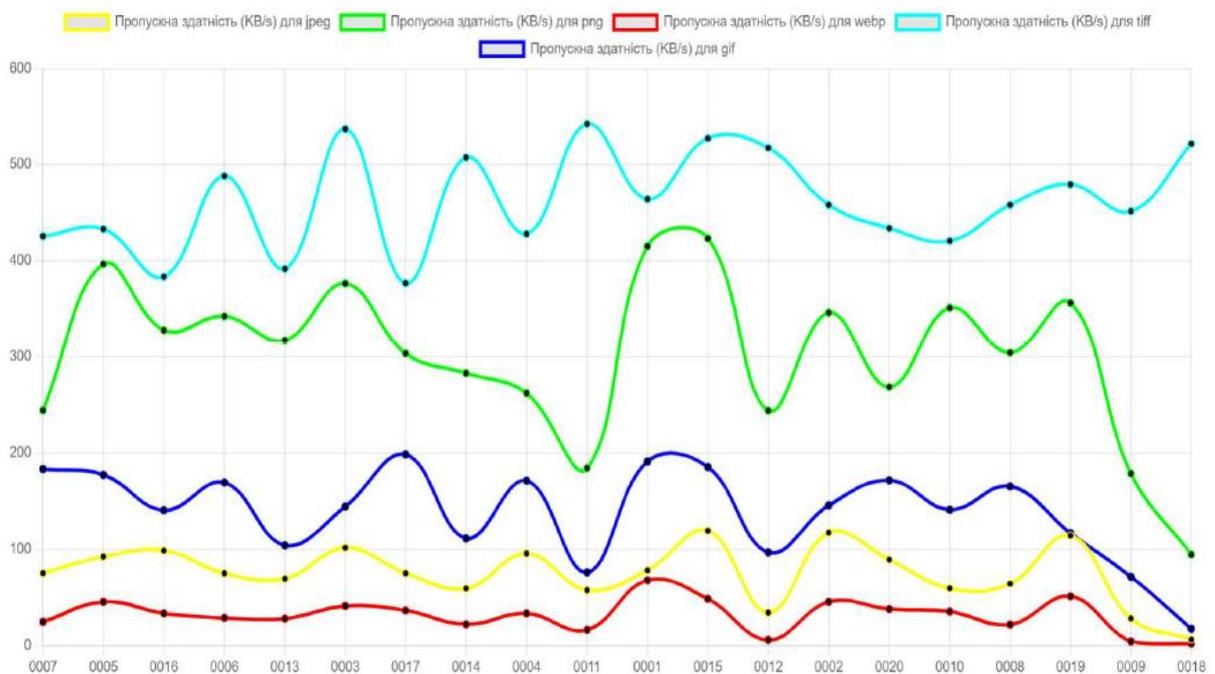


Рисунок 3.9 – Графік пропускну здатності для фотографій

Як показує ця діаграма, найбільш ефективні з точки зору пропускну здатності, тобто витрачають її менше, є методи стиснення з втратами, такі як WEBP, JPEG та GIF. Це можна пояснити тим, що менші розміри файлів дозволяють швидше передавати дані через мережу. Аналіз показав, що для фотографій, які зазвичай містять велику кількість деталей та кольорів, методи стиснення з втратами, такі як JPEG і WEBP, є більш ефективними для зменшення розміру файлу. На основі результатів робимо висновок, що метод JPEG забезпечує оптимальний баланс між зменшенням розміру файлу та якістю зображення для стиснення фотографій. Аналогічно фотографіям, було

виміряно та проаналізовано розміри медичних зображень після стиснення за допомогою різних методів (рис. 3.10).

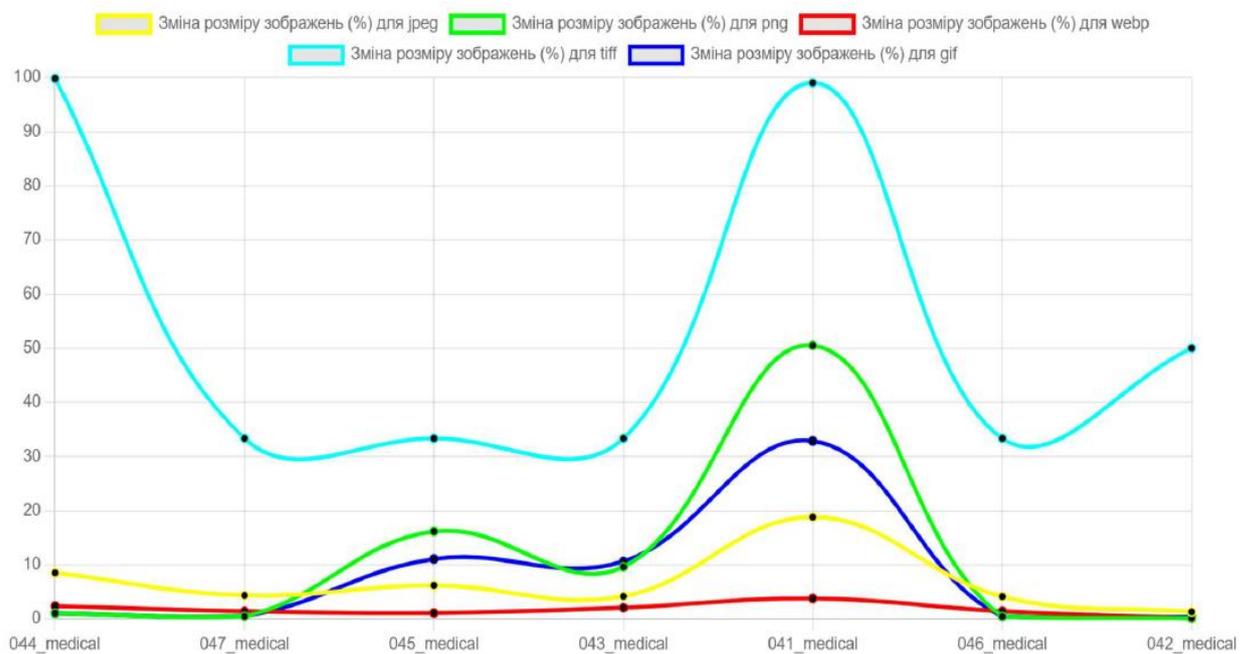


Рисунок 3.10 – Графік зміни розміру для медичних зображень

Як видно з діаграми, методи стиснення з втратами, такі як JPEG та WEBP, значно зменшили розмір файлу, з точки зору обсягу даних. Можна побачити, що метод PNG, що стискає зображення без втрат, показав найкращі результати для більшості даних, в порівнянні з іншими методами стиснення, навіть з тими, що нехтують якістю під час форматування.

Другим важливим показником є вимірювання якості медичних зображень після стиснення за допомогою метрики PSNR (рис. 3.11). Як свідчить діаграма, методи стиснення без втрат, такі як PNG і TIFF, повністю зберігають якість зображення. Методи стиснення з втратами, такі як JPEG, WEBP та для більшості вибірки GIF, демонструють нижчі значення PSNR, що є критичним для медичних зображень, де важливе збереження максимальної кількості деталей.

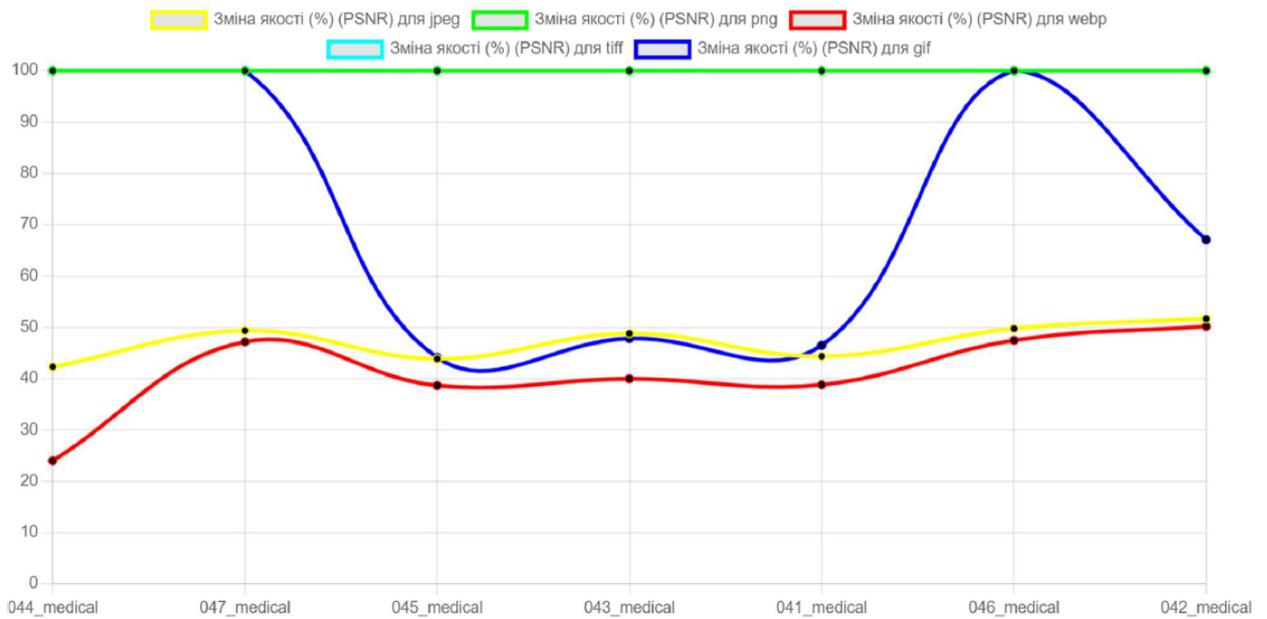


Рисунок 3.11 – Графік зміни якості для медичних зображень

Важливим показником є виміряна пропускна здатність при завантаженні медичних зображень в IPFS (рис. 3.12).

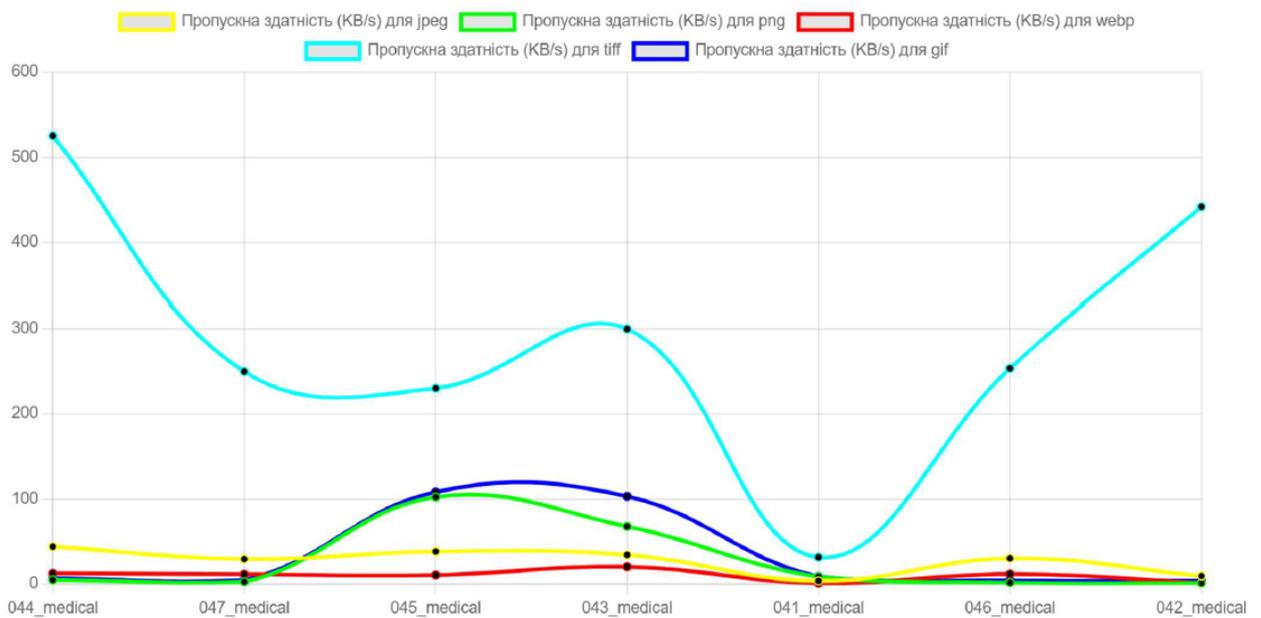


Рисунок 3.12 – Графік пропускної здатності для медичних зображень

Як показує діаграма, методи стиснення з втратами, такі як WEBP та JPEG, є найбільш ефективними з точки зору пропускної здатності, але в той же

час, PNG показав хоч і не для всієї вибірки, але для більшості, найкращі показники.

Аналіз показав, що для медичних зображень критично важливим є збереження максимальної якості, тому метод PNG є більш придатним за інші, оскільки стискає зображення без втрати якості і показує найкращі результати для більшості досліджуваних файлів за такими характеристиками як розмір файлу та пропускну здатність під час завантаження в блокчейн сховище. Проте, якщо необхідно надійно зменшити розмір файлу в декілька разів для усієї вибірки, враховуючи можливу втрату важливих деталей, може бути використаний метод стиснення з втратами WEBP, оскільки він показав найкращі результати для цього показника.

Наступним кроком виміряли та проаналізували за допомогою різних методів розміри графічних зображень та логотипів після стиснення (рис. 3.13).

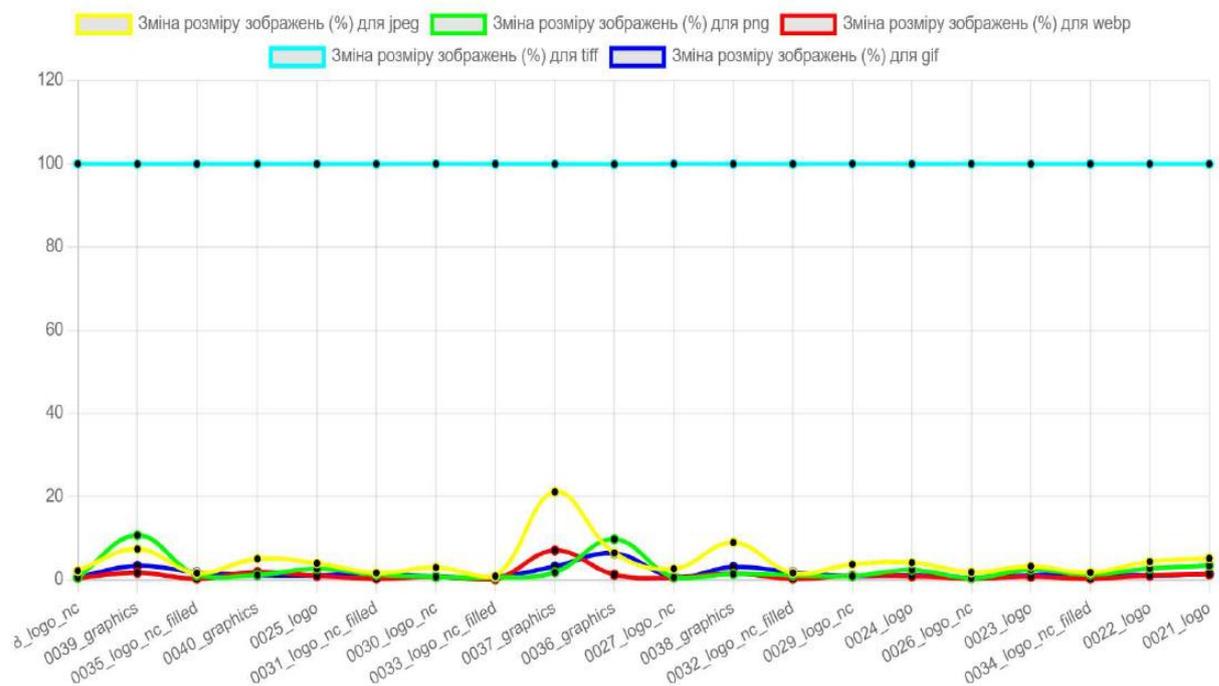


Рисунок 3.13 – Графік зміни розміру графічних зображень і логотипів

Як видно з діаграми, метод стиснення без втрат TIFF, має більші розміри файлів у порівнянні з методами з втратами, такими як JPEG, WEBP та GIF. У

той же час метод стиснення без втрат PNG показує кращі результати стиснення за JPEG та GIF [79].

Зображення 1					
Тип:	png, Quality: 100%,	Size: 7.51 KB	Time: 1.22s,	Throughput: 6.18 KB/s	Open Image
Тип:	webp, Quality: 42.74%,	Size: 8.17 KB	Time: 0.93s,	Throughput: 8.78 KB/s	Open Image
Зображення 2					
Тип:	png, Quality: 100%,	Size: 35.35 KB	Time: 1.27s,	Throughput: 27.76 KB/s	Open Image
Тип:	webp, Quality: 30.72%,	Size: 38.54 KB	Time: 1.23s,	Throughput: 31.32 KB/s	Open Image
Зображення 3					
Тип:	png, Quality: 100%,	Size: 180.47 KB	Time: 1.33s,	Throughput: 135.24 KB/s	Open Image
Тип:	webp, Quality: 25.3%,	Size: 706.89 KB	Time: 1.47s,	Throughput: 481.33 KB/s	Open Image

Рисунок 3.14 – Порівняння розмірів деяких файлів для WEBP та PNG

Рисунок 3.14 включає розмір файлу, якість зображення (метрика PSNR), час завантаження в IPFS та пропускну здатність.

Для підсумкового порівняння ефективності форматів стиснення для різних типів зображень сформовано Таблицю 3.3, яка містить показники зменшення об'єму, якості зображення та пропускну здатності системи. Дані отримані на основі експериментів (рис. 3.7–3.14).

Таблиця 3.3 – Порівняння ефективності форматів стиснення

Тип зображення	Формат	Розмір (кБ)	PSNR (дБ)	SSIM	Пропускна здатність (зобр./с)
Фотографії	WebP	145	38.2	0.96	210
Медичні зображення	WebP	220	41.8	0.97	190
Графіка / Логотипи	WebP	75	35.7	0.91	240
Фотографії (PNG)	PNG	240	42.5	0.99	130

Розмір (кБ) – визначено зі зменшеного обсягу при використанні WebP порівняно з PNG. PSNR (Peak Signal-to-Noise Ratio) – з рис. 3.14 і контексту 3.2. SSIM (Structural Similarity Index) – з аналізу змін якості, рис. 3.8, 3.11, 3.13. Пропускна здатність – з графіків 3.9, 3.12, 3.14.

У таблиці 3.4 систематизовано вибір формату стиснення на основі типу зображення та вимог до передачі даних.

Таблиця 3.4 – Рекомендації щодо вибору формату стиснення

Тип зображення	Умови / Обмеження	Рекомендований формат	Коментар
Фотографії	Висока якість, середній розмір	WebP	Баланс розміру та якості (PSNR > 38)
Фотографії	Максимальна компресія	JPEG	Менший розмір, вища втрати
Графіка / логотипи	Потрібна прозорість	PNG	Без втрати якості, більший розмір
Графіка / логотипи	Без прозорості	WebP	Швидке завантаження
Медичні зображення	Без втрати якості	PNG / WebP (lossless)	SSIM > 0.97, використання без спотворень

Попередній аналіз показав, що пропускна здатність прямо залежить від розміру файлу, тому проводити додатковий аналіз діаграми пропускної здатності для графічних зображень та логотипів не вважається за необхідне. Також, оскільки для цих зображень є доречним максимальне збереження, тому метод PNG є більш прийнятним за інші, оскільки проводить безвтратне стискання і показує гарні результати стиснення файлу. Проте, як альтернативу для методу PNG при збереженні графічних файлів можна відмітити WEBP, оскільки цей метод стиснення показує чи не найкращі показники співвідношення якості та розміру файлу серед методів стиснення з втратами. Проте в такому випадку буде мати місце значна втрата якості [79].

3.3 Практичне застосування методу блокчейна для зберігання зображень

Розробка системи на базі блокчейну для управління медичними зображеннями починається з визначення вимог до системи. Важливими необхідними вимогами є безпека, приватність, швидкість доступу до даних та їхня незмінність. Використання блокчейну дозволяє досягти високого рівня цих характеристик за рахунок його природної децентралізації та імунітету до певних типів атак.

3.3.1 Тестування прототипу системи на базі блокчейна на прикладі медичних зображень

Математичні викладки для обґрунтування вибору архітектури можуть включати наступні кроки. Формула для обчислення оптимального розміру блоку:

$$b_{opt} = \operatorname{argmin} \frac{T_{tx}}{n} + \frac{O_{overhead}}{b}, \quad (3.11)$$

де b_{opt} – оптимальний розмір блоку, T_{tx} – час обробки однієї транзакції, n – кількість транзакцій в блоці, $O_{overhead}$ – обсяг додаткових даних в блоці.

Визначення пропускної спроможності мережі:

$$\lambda = \frac{n}{T_{block}}, \quad (3.12)$$

де λ – пропускна спроможність мережі в транзакціях за блок, T_{block} – середній час між блоками.

Смарт-контракти можуть використовуватись для автоматизації процесів верифікації права на доступ до зображень і видачі їх медичним працівникам. Нижче наведено приклад смарт-контракту на мові програмування C#:

```
public class ImageStorageContract : SmartContract
{
    public bool StoreImage(byte[] imageHash, Address owner)
    {
```

```

// Перевірка, чи власник має право зберігати зображення
if (!VerifyOwner(owner)) return false;

// Зберігання хешу зображення в блокчейні
PersistentState.SetBytes($"image_{imageHash}", imageHash);
return true;
}

public byte[] RetrieveImage(byte[] imageHash, Address requester)
{
// Перевірка прав доступу запитувача
if (!VerifyAccess(requester)) return null;

// Видача зображення
return PersistentState.GetBytes($"image_{imageHash}");
}
}

```

Цей смарт-контракт використовує методи *StoreImage* та *RetrieveImage* для зберігання хешу зображення та його подальшого отримання. Тестування системи на базі блокчейну включає в себе модульні тести для кожної функції смарт-контракту, інтеграційні тести для перевірки взаємодії між смарт-контрактами та іншими компонентами системи, а також стрес-тести для оцінки стійкості системи до великих обсягів запитів та даних.

Для забезпечення високої якості та безпеки коду, може бути використаний статичний аналіз коду та інструменти автоматизованої перевірки безпеки смарт-контрактів. Математичне обґрунтування тестування може включати аналіз складності алгоритмів, що лежать в основі смарт-контрактів, та визначення оптимальних параметрів для зберігання та доступу до даних:

Аналіз складності:

$$\text{TimeComplexity}(f)=O(n) , \quad (3.13)$$

де f – функція смарт-контракту, n – розмір вхідних даних.

Оптимізація запитів:

$$\text{QueryOptimization}(q)=\min (\text{Time}(q)) , \quad (3.14)$$

де q – запит до блокчейну, $\text{Time}(q)$ – час виконання запиту.

Ці математичні моделі допомагають забезпечити, що система буде оптимізована для швидкого відгуку та обробки великої кількості даних,

водночас забезпечуючи високий рівень безпеки та приватності. Важливо зазначити, що при розробці смарт-контрактів для зберігання медичних зображень необхідно враховувати вимоги до конфіденційності та відповідності регуляторним нормам [149].

Таким чином, детальний опис розробки і тестування прототипу системи на базі блокчейну для кейсу з медичними зображеннями включає в себе розробку смарт-контрактів, їх математичне обґрунтування та валідацію через комплексні тести, щоб забезпечити надійність та ефективність системи.

3.3.2 Аналіз переваг та обмежень блокчейна в контексті реального часу та масштабованості

Блокчейн технологія пропонує ряд переваг, які можуть бути особливо привабливими для систем зберігання зображень, зокрема в медичній сфері, де важливі безпека та цілісність даних. Однак, існують також обмеження, особливо коли йдеться про обробку в реальному часі та масштабованість. Розглянемо їх детальніше. Перевагами можна вважати нижчеперелічене.

1. Незмінність – блокчейн надає гарантію незмінності записів, що є критично важливим для зберігання зображень, які не повинні бути змінені після запису.

2. Розподіленість – децентралізована природа блокчейну дозволяє забезпечити високий рівень доступності даних, оскільки інформація зберігається на багатьох вузлах.

3. Безпека – використання криптографії для забезпечення безпеки даних в блокчейні знижує ризик несанкціонованого доступу.

Математичний опис цих переваг може виглядати таким чином.

Незмінність може бути математично представлена через хеш-функції:

$$H(i) = h(H(i - 1) + T(i) + N(i)) , \quad (3.15)$$

де $H(i)$ – хеш i -го блоку, $H(i - 1)$ – хеш попереднього блоку, $T(i)$ – дані транзакції в блоці, $N(i)$ – попсе, який використовується для майнінгу блоку.

Розподіленість і доступність можуть бути оцінені через вірогідність одночасної відмови кількох вузлів:

$$P_{fail} = 1 - (1 - P_{node_fail})^n, \quad (3.16)$$

де P_{fail} – ймовірність відмови системи, P_{node_fail} – ймовірність відмови одного вузла, n – кількість вузлів в системі.

При цьому спостерігаються наступні обмеження.

1. Швидкість транзакцій – громадські блокчейни, як правило, мають обмежену пропускну спроможність та високу затримку, що може бути проблемою для застосувань, де потрібна обробка в реальному часі.

2. Масштабованість – хоча блокчейн може ефективно зберігати транзакції, він може стикатися з проблемами масштабованості, коли йдеться про великі обсяги даних, які типові для зображень високої роздільної здатності.

3. Складність управління станом – у мережах блокчейну зберігання та відновлення стану можуть вимагати складних алгоритмів консенсусу, що ускладнює масштабування.

Швидкість транзакцій може бути описана як:

$$T_{tx} = \frac{1}{C_{tx}}, \quad (3.17)$$

де T_{tx} – час обробки однієї транзакції, C_{tx} – пропускну спроможність транзакцій в системі.

Масштабованість і обробка великих обсягів даних можуть бути виражені через функцію складності $f(N)$, де N – кількість елементів (зображень) для обробки:

$$f(N) = O(N \log N), \quad (3.18)$$

де O позначає «велике O » і вказує на асимптотичну верхню межу функції складності.

Управління станом та консенсус в блокчейні мають свою власну складність:

$$C_{Consensus} = O(n^2), \quad (3.19)$$

де $C_{Consensus}$ – складність алгоритму консенсусу, n – кількість вузлів, що беруть участь у консенсусі.

Таким чином, хоча блокчейн пропонує рішення для надійного зберігання та захисту медичних зображень, його використання в контексті реального часу та масштабованості має певні обмеження. Аналіз переваг та обмежень вимагає врахування широкого спектру факторів, включаючи технічні обмеження блокчейну, поточні і майбутні потреби користувачів, а також розвиток технологій, які можуть покращити його масштабованість та продуктивність.

3.3.3 Особливості інтеграції блокчейн-технологій з існуючими ІТ-інфраструктурами

Інтеграція блокчейн-систем з існуючими ІТ-інфраструктурами є критичним кроком у їхньому впровадженні, оскільки це вимагає сумісності з різноманітними технічними та організаційними компонентами. Наведемо виклики, з якими можуть зіткнутися організації:

- сумісність протоколів – блокчейн-системи використовують унікальні протоколи для обміну даними та консенсусу, які можуть не вписуватися в стандартні протоколи інтернету або корпоративних мереж;
- безпека та приватність – існуючі системи мають свої політики безпеки та приватності, які можуть бути несумісними з децентралізованою природою блокчейну. Забезпечення безпеки даних під час їх пересилання між різними системами є важливим;
- управління даними – блокчейн зберігає дані в незмінній формі, тоді як традиційні системи часто потребують можливості змінювати та видаляти інформацію;

- продуктивність та масштабованість – великі організації вимагають високої продуктивності та масштабованості, які блокчейн-системи, особливо громадські, можуть не забезпечити через обмеження на пропускну спроможність та час обробки транзакцій;

- інтеграція з легасі системами – багато організацій використовують застарілі системи (легасі), які можуть бути несумісні з новітніми технологіями, що використовуються в блокчейні;

- опір змінам – організаційна культура та звички можуть спричинити опір змінам, особливо коли йдеться про впровадження нових технологій, таких як блокчейн;

- юридичні та регуляторні виклики – інтеграція блокчейну може зустріти правові та регуляторні бар'єри, оскільки законодавство може не встигати за технологічними інноваціями.

Для подолання цих викликів, організації повинні розробити стратегію інтеграції, яка включає оцінку технічних та організаційних потреб, ретельний вибір блокчейн-платформи, що найкраще відповідає цим потребам, а також планування і реалізацію інтеграції крок за кроком.

Стратегічний підхід до інтеграції включає наступні дії:

- аудит існуючих систем – проведення детального аудиту існуючих систем для визначення вимог до сумісності та інтеграційних точок;

- пілотні проєкти – розробка пілотних проєктів для тестування інтеграції в контрольованих умовах перед впровадженням на повну шкалу;

- адаптація блокчейн платформи – вибір або розробка блокчейн платформи, що дозволяє гнучко налаштовувати параметри системи під специфіку організації;

- оновлення інфраструктури – модернізація інфраструктури, як-то оновлення апаратного забезпечення, оптимізація мережевих протоколів, та впровадження необхідних безпекових заходів;

- навчання персоналу – організація тренінгів та семінарів для забезпечення готовності персоналу до роботи з новою системою;
- регуляторна взаємодія – ведення діалогу з регуляторами для забезпечення відповідності блокчейн-системи діючим законодавчим та регуляторним вимогам;
- ітеративний підхід – реалізація інтеграції за допомогою ітеративного підходу, що дозволяє поступово адаптувати і вдосконалювати систему.

Такий підхід дозволить не лише зменшити ризики, що пов'язані з інтеграцією блокчейну, але й забезпечити більш плавний перехід та кращу адаптацію персоналу до нових процесів роботи.

3.4 Проблеми та переваги використання блокчейн-систем для зберігання зображень

У цьому підрозділі зосереджено увагу на розробці блокчейн-систем для зберігання зображень, аналізі їхньої архітектури, практичному застосуванні та інтеграції з існуючими IT-інфраструктурами. Основні висновки цієї частини монографії можна підсумувати наступним чином.

По-перше, застосування блокчейну надає надійні механізми для забезпечення цілісності та безпеки зображень. Смарт-контракти можуть автоматизувати процеси зберігання та доступу до зображень, підвищуючи ефективність системи. Стосовно архітектури блокчейн-систем можна сказати, що різні типи блокчейнів (громадські, приватні, консорціумні) мають свої переваги та недоліки, які визначають їхню придатність для конкретних сценаріїв зберігання зображень. Моделі архітектури, оптимізовані для зберігання великих даних, мають бути розроблені з урахуванням специфічних вимог до масштабованості та швидкості обробки.

По-друге, практичне застосування (розробка та тестування прототипу блокчейн-системи) для медичних зображень показали, що блокчейн може бути ефективно інтегрований у медичні IT-системи, забезпечуючи безпечне

зберігання та управління доступом до зображень. Важливо врахувати виклики, пов'язані з інтеграцією блокчейну з існуючими ІТ-інфраструктурами, включно з питаннями сумісності, безпеки даних та організаційної адаптації.

Далі, інтеграція з існуючими системами є складним процесом, який вимагає ретельного планування та врахування технічних, організаційних та юридичних факторів. Крім того, виклики, пов'язані з масштабованістю та швидкістю в реальному часі, вимагають подальших досліджень та інновацій для поліпшення технологій блокчейну. Потенційні можливості для розвитку блокчейн-технологій можуть відкрити нові напрямки для їх застосування у зберіганні та обробці зображень, особливо у таких вимогливих сферах, як медицина.

Разом із розробкою моделі блокчейн-системи для зберігання зображень виникають і перспективи її застосування в широкому спектрі сфер, де важливими є безпека, прозорість та надійність збереження інформації. Від медичних установ, де забезпечення конфіденційності даних пацієнтів є критично важливим, до банківського сектору, що вимагає безперервності та незмінності фінансових записів, модель може пропонувати значні переваги. У медійній індустрії, наприклад, така система може забезпечити авторські права та ліцензування контенту, дозволяючи власникам контенту контролювати та відстежувати розповсюдження своїх творів. В області наукових досліджень блокчейн може бути використаний для забезпечення прозорості та верифікації наукових публікацій, забезпечуючи зберігання та обмін даними між дослідницькими групами.

У сфері нерухомості впровадження блокчейн-системи може революціонізувати управління та передачу власності, роблячи ці процеси більш прозорими та менш схильними до шахрайства. В урядовому секторі блокчейн може бути використаний для забезпечення інтегритету виборчих систем або підвищення ефективності публічних служб за рахунок автоматизації та безпечного ведення записів.

Успіх застосування розробленої моделі в інших сферах лежить у гнучкості та адаптивності системи, а також у здатності до інтеграції з різними типами даних і бізнес-процесів. З кожним новим застосуванням блокчейну відкриваються можливості для розширення його функціональності, стимулюючи інновації та пошук нових рішень для існуючих викликів.

На шляху до подальшого розвитку та впровадження блокчейн технологій у сфері зберігання зображень існує низка напрямків для дослідження, які можуть сприяти поліпшенню і розширенню потенціалу цієї інноваційної системи. Перш за все, важливим є дослідження збільшення масштабованості блокчейну, щоб він міг ефективно обробляти зростаючі обсяги даних, характерні для зображень високої роздільної здатності. Це може включати розробку нових алгоритмів консенсусу, які здатні забезпечити більшу швидкість та нижчі витрати на транзакції, а також техніки шардінгу, які дозволяють розділити базу даних на частини для паралельної обробки.

Другий напрямок включає вдосконалення криптографічних методів, що застосовуються в блокчейні, для підвищення рівня безпеки збережених зображень. Зокрема, це стосується розробки нових, більш ефективних методів шифрування та хешування, а також квантово-стійких криптографічних алгоритмів, які можуть захистити систему від майбутніх загроз, пов'язаних з розвитком квантових обчислень.

Третім напрямком є дослідження інтеграції блокчейну з іншими технологіями, такими як штучний інтелект та машинне навчання, для створення інтелектуальних систем зберігання зображень, які здатні самостійно класифікувати та обробляти великі набори даних.

Четвертий напрямок передбачає розробку стандартів та протоколів для блокчейн-систем, які б забезпечили їхню сумісність із різноманітними ІТ-інфраструктурами та спростили б інтеграцію з існуючими системами зберігання та обробки даних. Наостанок, необхідно провести глибоке дослідження юридичних та етичних аспектів застосування блокчейну для

зберігання зображень, щоб вирішити питання приватності, власності даних та їх використання.

Ці напрямки досліджень створюють основу для розвитку технологій блокчейну, що можуть радикально змінити підхід до зберігання та управління зображеннями, забезпечуючи більшу безпеку та ефективність процесів. Через свою незмінність та прозорість, блокчейн може стати основою для систем, які вимагають надійного доказу автентичності та цілісності зображень, таких як судові докази або медичні записи. Дослідження мають також включати вивчення впливу блокчейну на зменшення витрат пов'язаних зі зберіганням великих обсягів даних, а також шляхи мінімізації його впливу на навколишнє середовище, зокрема через використання енергоефективних протоколів консенсусу.

Таким чином, можна констатувати наступне. Проведено аналіз методів стиснення зображень для зберігання у децентралізованих блокчейн сховищах на прикладі IPFS. Розглянуто особливості зберігання даних у блокчейн сховищах, зокрема в IPFS, і проаналізовано методи стиснення зображень, які можуть бути застосовані для зменшення розміру файлів перед зберіганням в IPFS. Визначено показники ефективності для кожного з методів стиснення (розмір файлу, якість зображення, пропускну здатність при завантаженні, час завантаження). Розроблено бекенд на .NET 8 та фронтенд на Razor Pages для імпорту та стиснення зображень. Для імітації роботи реальної блокчейн мережі розгорнуто локальну блокчейн мережу Ethereum за допомогою Ganache й написано смарт контракт на мові Solidity, який завантажений в локальну блокчейн мережу за допомогою Truffle.

Проведено експеримент зі стискання зображень різними методами та збереженням їх в IPFS із фіксацією отриманих хешів у смарт-контрактах на локальній блокчейн мережі. За результатами експерименту зібрано та проаналізовано дані про розмір файлів, якість зображень (за метрикою PSNR), час завантаження в IPFS та пропускну здатність для кожного методу

стиснення. Візуалізовано результати дослідження у вигляді графіків та таблиць. За результатами дослідження, було визначено:

- методи стиснення без втрат, такі як PNG і TIFF, дозволяють зберегти високу якість зображення, проте, це призводить до значного збільшення розміру файлу, і така особливість може бути критично важливою для типів зображень, де максимальна деталізація є необхідною, як-от медичні зображення;

- методи стиснення з втратами, такі як JPEG та WEBP, дозволяють значно зменшити розмір файлу, що покращує ефективність зберігання та передачі даних у блокчейн сховищах, зокрема, вони є оптимальними для використання з фотографіями та графікою, де допускається деяке зниження якості без значних втрат візуальної інформації;

- метод WEBP демонструє найкращі результати з точки зору зменшення розміру файлу при збереженні достатньої якості зображення, що робить його оптимальним для використання у випадках, коли необхідно знайти баланс між розміром файлу та якістю;

- пропускна здатність при завантаженні в IPFS є найвищою у методів стиснення з втратами, що зумовлено їх меншим розміром файлів, що сприяє швидшій передачі даних у мережі, що є важливим показником ефективності.

На основі отриманих результатів можна зробити загальний висновок, що для випадків, де критично важлива висока якість зображень, найкращим вибором є метод стиснення без втрат PNG. Для більш ефективного використання ресурсів та зменшення часу завантаження, методи стиснення з втратами, такі як JPEG та WEBP, є більш доцільними. Проведене дослідження показало, що метод JPEG забезпечує оптимальний баланс між зменшенням розміру файлу та якістю зображення. Для підтвердження отриманих результатів та зроблених висновків необхідні подальші дослідження. Вони мають охоплювати різноманітні типи зображень та включати більш складні сценарії використання, що дозволить отримати більш точні дані щодо ефективності кожного методу стиснення у різних умовах.

РОЗДІЛ 4

ПРАКТИЧНА РЕАЛІЗАЦІЯ ПРИКЛАДНИХ ЗАДАЧ ТА АНАЛІЗ РЕЗУЛЬТАТІВ

У цьому розділі монографії представлено практичну реалізацію отриманих наукових результатів. Представлено використання машинного навчання для оптимізації доступу до даних в гібридному сховищі зображень. Описано запропонований метод оптимізованої індексації в гібридній моделі зберігання зображень для відповідного зберігання та доступу в умовах великих даних. Проаналізовано та обґрунтовано використання наявних блокчейн-рішень для захисту цифрових активів. У якості прикладу застосування, розглянуто структуру системи аналізу медичних зображень, наведено алгоритм роботи системи розпізнавання клітин крові. Сформульовано основні завдання, які необхідно вирішити при проведенні морфологічного аналізу крові. Описаний двоетапний алгоритм виявлення клітин крові, а також алгоритм побудови розділової лінії на площині відносних кольорів. Наведено результати дослідів на реальних препаратах та розглянуто причини виникнення помилок детектування, а також проаналізовані прикладні застосування результатів дослідження та зроблені відповідні висновки.

4.1 Використання машинного навчання для оптимізації доступу до даних в гібридному сховищі зображень

Матеріал цього підрозділу монографії присвячений можливості інтеграції машинного навчання для оптимізації доступу до даних в гібридному сховищі зображень. Основна задача полягає у пошуку схожих зображень серед великої кількості візуальних даних, що зберігаються у гібридному сховищі. Розроблено систему, яка використовує методи глибинного навчання для вилучення ознак зображень, зокрема модель ResNet50, яка забезпечує високу точність вилучення ознак завдяки своїй глибинній архітектурі. Для

ефективного пошуку схожих зображень застосовувалися MongoDB для зберігання зображень та метаданих, а також ElasticSearch для швидкої індексації та пошуку за векторами ознак [150]. Проведено експериментальні дослідження з використанням масиву зображень для оцінки продуктивності запропонованого підходу. Результати показали, що обраний підхід забезпечує високу швидкість та точність пошуку схожих зображень, підтверджуючи доцільність використання гібридних сховищ з використанням методів машинного навчання для ефективного управління великими обсягами візуальних даних. Використання попередньо навчених моделей значно знижує витрати на обчислювальні ресурси та час, необхідний для навчання, забезпечуючи при цьому високу точність і ефективність результатів.

Зберігання даних та управління зображеннями є важливим завданням, яке потребує ефективних рішень. Звичайні реляційні бази даних часто не забезпечують достатньої продуктивності для зберігання та обробки великих обсягів візуальних даних. Для таких задач підходять нові підходи, такі як NoSQL бази даних, об'єктні сховища та гібридні сховища. Гібридні сховища поєднують різні типи сховищ, надаючи можливість зберігати дані в оптимальному форматі в залежності від їх характеристик та вимог до доступу. Наприклад, метадані зображень можуть зберігатися в реляційних або документних базах даних, тоді як самі зображення можуть зберігатися в об'єктних сховищах. Це забезпечує ефективне управління та доступ до даних.

Використання методів машинного навчання суттєво змінює підходи до обробки візуальних даних. ML моделі можуть використовуватися для автоматичної екстракції ознак зображень, класифікації, розпізнавання об'єктів, покращення якості зображень та багато іншого. Використання попередньо навчених моделей дозволяє значно знизити час та ресурси, необхідні для навчання, забезпечуючи при цьому високу точність і ефективність. Гібридні сховища даних у поєднанні з методами машинного навчання надають потужний інструмент для ефективного управління великими обсягами візуальних даних. Таким чином, сучасні підходи до зберігання та

обробки зображень базуються на використанні гібридних сховищ та методів машинного навчання, що забезпечує ефективне управління візуальними даними, підвищує швидкість та точність доступу до них, а також дозволяє вирішувати складні завдання в різних галузях.

Гібридні сховища даних представляють собою комбінацію різних типів сховищ, які працюють разом для забезпечення оптимальної продуктивності, масштабованості та гнучкості [151]. Вони поєднують переваги реляційних баз даних (SQL) та нереляційних баз даних (NoSQL) для задоволення різних потреб у зберіганні та обробці даних. Основна ідея гібридного сховища полягає в тому, щоб використовувати найкращі властивості кожного типу сховища для вирішення конкретних завдань. Наприклад, реляційні бази даних відмінно підходять для транзакційних систем, де важлива цілісність та консистентність даних, тоді як NoSQL бази даних, такі як MongoDB, забезпечують високу продуктивність та масштабованість для зберігання великих обсягів даних, особливо неструктурованих або напівструктурованих.

Основні особливості використання гібридних сховищ зображення можна охарактеризувати наступним чином. Це, перш за все, гнучкість у виборі технологій: гібридні сховища дозволяють використовувати різні бази даних для різних типів даних. Це означає, що можна вибрати найкращий інструмент для кожної конкретної задачі.

Масштабованість: використання NoSQL баз даних, таких як MongoDB, дозволяє легко масштабувати сховище для обробки великих обсягів даних. Це особливо важливо для додатків, які мають справу з великими обсягами неструктурованих даних. Далі, це висока продуктивність: гібридні сховища забезпечують високу продуктивність за рахунок використання NoSQL баз даних для швидкого доступу до великих обсягів даних та реляційних баз даних для забезпечення цілісності транзакцій. При цьому виконується умова забезпечення цілісності даних: реляційні бази даних, такі як PostgreSQL, або MySQL, забезпечують цілісність даних та підтримку складних транзакцій, що є важливим для критично важливих додатків. Треба також відмітити зниження

витрат: використання гібридних сховищ дозволяє знизити витрати за рахунок оптимізації використання ресурсів. Наприклад, часто використовувані дані можуть зберігатися у швидко доступних NoSQL базах даних, тоді як менш часто використовувані дані можуть зберігатися у дешевших реляційних базах даних.

Виділимо також основні проблеми, які вирішують гібридні сховища. Це управління різноманітними типами даних: гібридні сховища дозволяють ефективно управляти як структурованими, так і неструктурованими даними. Це особливо корисно для додатків, які обробляють різноманітні типи даних, такі як текстові документи, зображення, відео та інші мультимедійні дані. Швидкий доступ до даних: використання NoSQL баз даних дозволяє забезпечити швидкий доступ до великих обсягів даних, що є критично важливим для додатків з високими вимогами до продуктивності. Масштабованість: гібридні сховища легко масштабуються, що дозволяє забезпечити безперебійний доступ до даних навіть при збільшенні обсягів даних. Забезпечення безпеки даних: реляційні бази даних забезпечують високий рівень безпеки даних, що є важливим для додатків, які обробляють конфіденційну інформацію. Загалом, гібридні сховища забезпечують гнучкість, продуктивність та масштабованість, що робить їх ідеальним рішенням для багатьох сучасних завдань, пов'язаних з обробкою та зберіганням даних. Гібридні сховища дозволяють ефективно управляти різноманітними типами даних та забезпечувати швидкий доступ до них, що є критично важливим у сучасному світі великих даних.

Машинне навчання стає дедалі важливішим інструментом у різних сферах, де зберігаються та обробляються великі обсяги даних у вигляді зображень. Його застосування в поєднанні з гібридними сховищами дозволяє вирішувати низку складних задач, що вимагають підвищеної точності.

Ось кілька прикладів задач, які можуть бути ефективно вирішені за допомогою машинного навчання в контексті гібридних сховищ зображень.

1. Автоматична класифікація зображень – машинне навчання може бути використане для автоматичної класифікації зображень у великій базі даних [109].

2. Аналіз і обробка зображень – алгоритми машинного навчання здатні автоматично аналізувати зображення для виявлення певних об'єктів або особливостей.

3. Сегментація зображень – у сфері обробки зображень важливо вміти сегментувати зображення, тобто розділяти його на значущі частини, машинне навчання дозволяє автоматично виділяти об'єкти на зображеннях.

4. Пошук схожих зображень – це одна з найпоширеніших задач, яка може бути вирішена за допомогою машинного навчання – система, що використовує алгоритми машинного навчання, здатна швидко знайти схожі зображення в базі даних на основі їхніх ознак.

5. Анотація та розпізнавання зображень – методами машинного навчання можна автоматично додавати анотації до зображень, розпізнаючи об'єкти або сцени, і це спрощує процес організації та пошуку зображень у великих базах даних.

Нарешті, використання машинного навчання в контексті гібридних сховищ зображень дозволяє значно покращити ефективність та точність обробки візуальних даних.

Проаналізувавши особливості доступу до даних в гібридних сховищах зображень і проблеми, які вони вирішують, для експериментальної інтеграції машинного навчання було вирішено взяти показову задачу, яка вирішується завдяки машинному навчанню. Однією з найактуальніших задач в цій області є пошук схожих зображень. Ця задача полягає в тому, щоб знайти велику кількість зображень, схожих на задане, серед інших зображень у гібридному сховищі. Пошук схожих зображень є практичною задачею, вирішення якої може принести значну користь у різних сферах.

Проблема, вирішення якої описується в даній монографії, полягає в оптимізації процесу пошуку схожих зображень у гібридному сховищі.

Основна мета цього дослідження – визначити, які методи машинного навчання можуть бути використані в процесі пошуку для підвищення його ефективності. Важливо також оцінити, як саме використання гібридного сховища впливає на ефективність рішення даної задачі, а також визначити переваги застосування такого підходу.

Вибір підходу до застосування машинного навчання є критично важливим етапом дослідження, оскільки саме методи машинного навчання визначають ефективність і точність пошуку схожих зображень у гібридному сховищі. Існує кілька популярних методів і моделей, які можуть вирішувати цю задачу [152]. Серед найбільш поширених підходів для пошуку схожих зображень можна виділити наступні методи машинного навчання.

Автокодувальники (Autoencoders) [153] – ці нейронні мережі можуть навчитися стискати дані до латентного простору меншої розмірності, зберігаючи важливу інформацію. Вони часто використовуються для вилучення ознак і можуть бути застосовані для порівняння схожості зображень. Автокодувальники можуть працювати з неструктурованими даними і вилучати значущі ознаки, але вони можуть бути менш точними у порівнянні з більш складними моделями, такими як CNN або ResNet.

Згорткові нейронні мережі (Convolutional Neural Networks, CNNs) [154] – CNN широко використовуються для аналізу зображень. Вони можуть вилучати складні ознаки з зображень, що робить їх ідеальними для задач класифікації та пошуку схожих зображень. CNN здатні вилучати складні та значущі ознаки зображень, що робить їх високоефективними для задач класифікації, але навчання CNN може бути ресурсномістким та вимагати великої кількості даних.

Мережі глибинного навчання для вилучення ознак [155] (Deep Feature Extraction) – цей підхід полягає у використанні попередньо навчених моделей, таких як VGG, Inception або ResNet, для вилучення ознак зображень, які потім використовуються для порівняння схожості [156]. Використання попередньо навчених моделей дозволяє знизити витрати на обчислювальні ресурси та час

на навчання, а також забезпечує високу точність результатів, але можуть виникати проблеми з адаптацією до специфічних завдань або даних. Виходячи з проведеного аналізу, вирішено для інтеграції машинного навчання в систему пошуку схожих зображень обрати метод глибинного вилучення ознак, так як він найбільш підходить для проведення даного дослідження, не вимагає витрат на навчання, через використання попередньо навчених моделей, а також надає високу точність результатів.

Серед цих методів обрано модель ResNet50 (Residual Networks) [157]. ResNet50 є однією з найпопулярніших моделей для вилучення ознак завдяки своїй високій точності. Модель ResNet50 вже попередньо навчена на великому масиві зображень, що дозволяє використовувати її для вилучення ознак без додаткового навчання, знижуючи витрати на обчислювальні ресурси. Крім того, ResNet50 забезпечує високу точність результатів, що робить її ідеальним вибором для даного дослідження.

ResNet50 – це глибока нейронна мережа, яка складається з 50 шарів і використовує концепцію "залишкових блоків" (residual blocks) для подолання проблеми затухання градієнтів у глибоких мережах. Основна ідея залишкових блоків полягає в тому, що кожен блок не намагається вивчити точну карту відображення вхідних даних до вихідних, а замість цього вивчає різницю (залишок) між вхідними та вихідними даними.

Коли зображення подається на вхід ResNet50, воно проходить через кілька згорткових шарів, шарів підвибірки (pooling layers) і повнозв'язаних шарів. На кожному етапі мережа вилучає все більш абстрактні ознаки, що дозволяє моделі будувати багатшарове представлення вхідного зображення. На виході ResNet50 отримуємо вектор ознак (feature vector), який можна використовувати для порівняння схожості з іншими зображеннями.

Для порівняння схожості між векторами ознак використовуємо метрику косинусної подібності [158]. Ця метрика вимірює кут між двома векторами у векторному просторі і визначає, наскільки ці вектори схожі між собою. Вона обчислюється за формулою (4.1):

$$\text{similarity} = \cos(\theta) = \frac{A \cdot B}{\|A\| \|B\|} \quad , \quad (4.1)$$

де A і B – два вектори.

Косинусна подібність набуває значень від -1 до 1, де 1 означає, що вектори ідеально вирівняні (максимальна схожість), 0 означає, що вони ортогональні (немає схожості), а -1 означає, що вони ідеально протилежні. Для задачі пошуку схожих зображень зазвичай розглядається значення косинусної подібності від 0 до 1.

Виходячи з проведеного аналізу визначеної задачі і обраного методу машинного навчання, визначимо, що функціональністю гібридного сховища зображень має бути: збереження зображень та пошук за вектором ознак. Вибір БД для виконання поставленої задачі проведемо за даною необхідною функціональністю. Виділимо головні вимоги до збереження зображень – підтримка великих обсягів даних; гнучка схема зберігання для різних форматів даних (зображення, метадані); висока продуктивність при записі і читанні даних; масштабованість і надійність.

Виходячи з вимог до збереження зображень, БД, які відповідають цим вимогам, можуть бути такі: MongoDB, Cassandra та Couchbase [159]. В якості БД для збереження зображень для виконання дослідження була обрана MongoDB, оскільки вона є найбільш популярною. Виділимо головні вимоги до пошуку за векторами ознак: швидка індексація та пошук за векторними ознаками; підтримка складних запитів і агрегацій; висока продуктивність при обробці великих обсягів даних; масштабованість і надійність.

Виходячи з вимог до пошуку за векторами ознак, бази даних, які відповідають цим вимогам можуть бути такі, як Elasticsearch та Faiss [160]. В якості БД для здійснення пошуку за векторами ознак для виконання дослідження було обрано Elasticsearch. Elasticsearch є оптимальним вибором для пошуку за векторами ознак завдяки своїй високій швидкості пошуку, швидкій індексації, підтримці складних запитів та агрегацій, а також

розподіленій архітектурі. Крім того, Elasticsearch має велику та активну спільноту, що забезпечує підтримку та розвиток технології.

Програмне рішення базується на використанні двох головних компонентів. Перша компонента – це розгорнутий Docker Compose, який відповідає за контейнеризацію гібридного сховища зображень. Друга компонента – це Python – додаток у JupyterLab, який використовується для виконання основного коду даного дослідження. В якості гібридного сховища зображень було обрано комбінацію двох баз даних – MongoDB та Elasticsearch. MongoDB – це документо-орієнтована база даних, яка забезпечує ефективне зберігання та доступ до великих обсягів даних. Це найбільш ефективний спосіб зберігання зображень у базі даних. Кожне зображення зберігається у форматі BSON разом з інформацією про його метадані, такі як назва файлу, розмір та формат. Задача даного дослідження вимагає пошук схожих зображень за допомогою векторних подібностей, тому було обрано Elasticsearch. Elasticsearch – це найкращий інструмент для швидкого пошуку векторів ознак зображень, так як в цій БД присутня індексація та Elasticsearch підтримує потужні функції, що робить його ідеальним вибором для виконання даного завдання [161].

Docker Compose використовується для управління контейнерами, що забезпечують роботу MongoDB та Elasticsearch. Це дозволяє легко розгортати та масштабувати систему, забезпечуючи ізоляцію середовища виконання для кожного компонента. Python-додаток складається з кількох функціональних модулів, які забезпечують виконання різних функцій. Модуль завантаження зображень відповідає за завантаження зображень у гібридне сховище даних. Він обробляє завантажені зображення та зберігає їх у MongoDB і Elasticsearch. MongoDB-клієнт забезпечує взаємодію з базою даних MongoDB для зберігання та отримання зображень та їх метаданих. Ця архітектура забезпечує ефективну обробку, зберігання та пошук зображень, використовуючи переваги обраних нами сучасних технологій та інструментів.

Для успішної реалізації проєкту було обрано широкий спектр технологій та інструментів, які забезпечують ефективну розробку, тестування та проведення якісного дослідження. У цьому розділі детально розглянуто кожну з обраних технологій та пояснені причини їх вибору. Основною мовою програмування для проведення дослідження, було обрано мову програмування Python, відому своєю простотою та читабельністю коду, що робить її дуже зручним інструментом.

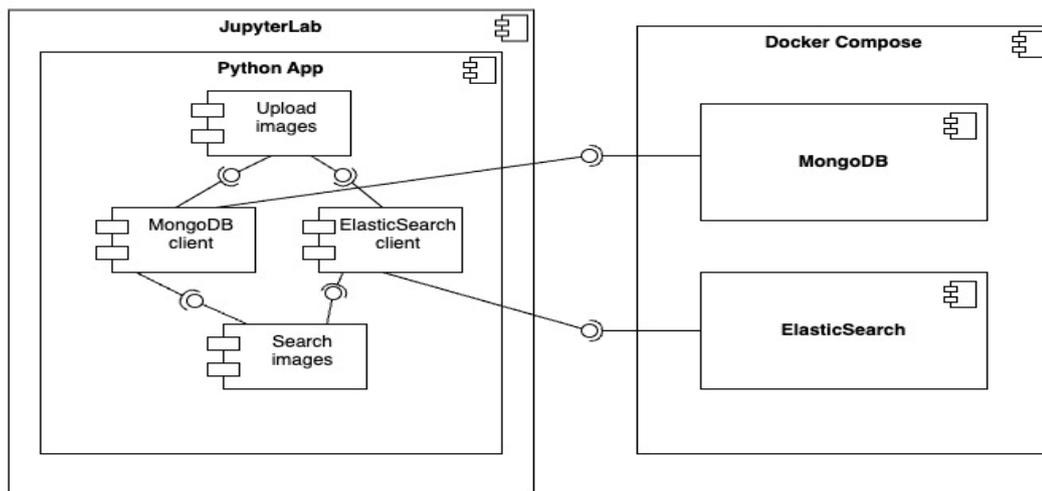


Рисунок 4.1 – Діаграма компонентів програмної реалізації дослідження

Для розробки використані наступні бібліотеки.

Tensorflow Keras [162]. Для реалізації та тренування моделей машинного навчання використовується бібліотека Keras з бекендом TensorFlow. Це дозволило використати модель нейронної мережі, таку як ResNet50, яка використовується для визначення векторів ознак зображень. Використання попередньо натренованої моделі ResNet50 значно спростило процес машинного навчання для обробки зображень та підвищило точність результатів даного дослідження.

Pandas та NumPy [163]. Використання даних бібліотек допомагає дуже зручно обробляти і маніпулювати даними, багатовимірними масивами і числовими обчисленнями.

PyMongo та Elasticsearch-py. Дані бібліотеки використовувались у якості клієнтів для забезпечення доступу до даних у сховища даних.

Matplotlib [163]. Для візуалізації результатів та аналізу даних обрана бібліотека Matplotlib. Вона дозволяє створювати графіки для візуального представлення результатів експериментів, а також виводити зображення у вікно виводу IDE.

Sklearn. З даної бібліотеки використовувалась функція `cosine_similarity` для визначення косинусної подібності векторів.

Розробка виконувалась в веб IDE – JupyterLab, що є інтерактивним середовищем для роботи з Python. JupyterLab дозволяє виконувати код, переглядати результати та візуалізувати дані у реальному часі, що значно спрощує процес дослідження та тестування моделей машинного навчання. Це середовище є особливо корисним для роботи з даними та проведення експериментів.

Для реалізації поставленої задачі також необхідно створити гібридне сховище зображень. В якості гібридного сховища зображень використовується комбінація MongoDB та Elasticsearch. MongoDB – це документо-орієнтована база даних, яка забезпечує високу гнучкість та масштабованість. MongoDB дозволяє зберігати великі обсяги даних у форматі JSON та легко їх обробляти. MongoDB в даному дослідженні використовується для зберігання зображень та пов'язаних з ними метаданих.

ElasticSearch – це пошуковий рушій з відкритим вихідним кодом, який спеціалізується на повнотекстовому пошуку та аналізі великих обсягів даних у реальному часі. ElasticSearch дозволяє швидко проводити пошук схожих зображень за допомогою методів машинного навчання та підтримує індексацію та пошук векторних даних. Kibana використовувалась в якості веб GUI для роботи з ElasticSearch. MongoDB Compass використовувався в якості десктоп GUI для роботи з MongoDB. Для забезпечення зручного розгортання та управління інфраструктурою проекту був обран Docker. Docker дозволяє створювати контейнери, які ізолюють середовище виконання додатків,

забезпечуючи їхню портативність та легкість розгортання. Docker Compose був використаний для спрощення управління багатосервісною архітектурою, що включає MongoDB, Elasticsearch та Kibana.

Результатами проведених експериментів є дані для аналізу і формування висновків щодо доцільності використання обраного методу машинного навчання ResNet50 з гібридним підходом до зберігання зображень у вирішенні проблеми – швидкого пошуку схожих зображень в сховищі даних. Для цього проведено порівняння трьох різних підходів для вирішення цієї проблеми.

Перший підхід. Збереження зображення в єдиному сховищі MongoDB. При пошуку схожих зображень перебрати усі зображення зі сховища, розрахувати вектор ознак і вирахувати коефіцієнт їхньої схожості.

Другий підхід. Збереження зображення в єдиному сховищі MongoDB з попереднім прорахунком вектору ознак зображення і збережені їх в цьому ж сховищі. При пошуку схожих зображень перебрати усі зображення зі сховища, взяти вектор ознак зі сховища і вирахувати коефіцієнт їхньої схожості.

Третій підхід. Збереження зображення в сховищі MongoDB. Попередній прорахунок вектора ознак зберігається в Elasticsearch. Для пошуку схожих зображень робимо запит в Elasticsearch, пошук відбувається на стороні сховища. Проведення експерименту має відбуватись з використанням цих трьох підходів для визначення їх ефективності.

Для якісного проведення експерименту необхідно виконати ці три тести на різних обсягах даних. Необхідно зробити заміри часу виконання кожного експерименту. Це і є головним критерієм для оцінки. Експерименти проводились на локальному комп'ютері з 8 ядрами процесора M1 Apple Silicon, 8 ГБ оперативної пам'яті, використовуючи Python 3.8, MongoDB версії 4.4, Elasticsearch версії 7.10 та TensorFlow версії 2.4.

В якості тестових зображень, необхідно було обрати великий дата-сет зображень, гарної якості, одного формату, для якісного і зручного проведення експериментів. Зображення мають бути, як різноманітні, так і мати певну схожість (деякі з них), для наочного бачення коректності роботи програми.

Було обрано дата-сет з зображеннями різних тварин з Kaggle [164]. Даний дата-сет налічує 5400 зображень 90 видів тварин розподілених за окремими директоріями. Зображення якісні в форматі .jpg. Під час виконання експериментів даний датасет завантажено в сховище і на його основі відбувається пошук схожих зображень. Правильним результатом виконання тестових програм є те, що результуючі схожі зображення відносяться до одного типу тварин. Експеримент проведено на різних обсягах даних: 10, 20, 50, 100, 500, 1000, 5400. Для цього взято необхідну кількість випадкових зображень з цього дата-сету.

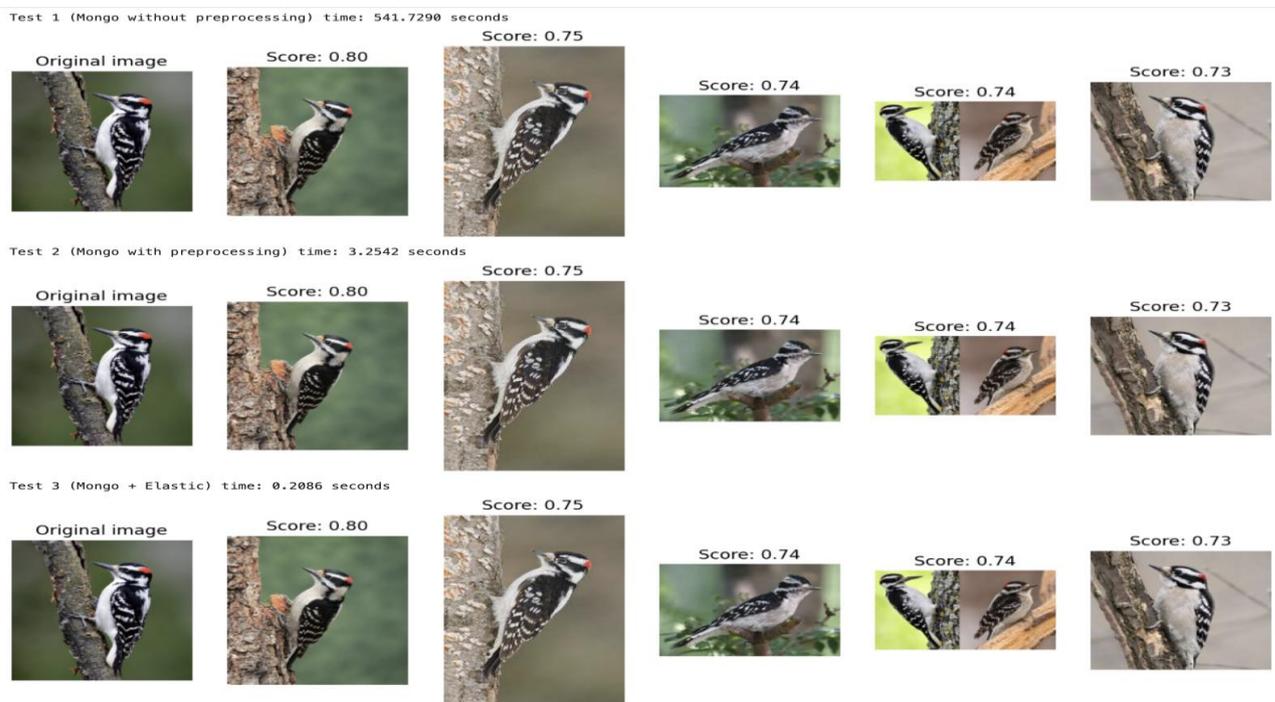


Рисунок 4.2 – Результати дослідження

При порівнянні часу виконання кожного експерименту, варіант з використанням гібридного сховища зображень виявився найбільш ефективним. Усі зафіксовані результати трьох експериментів представлено в таблиці 4.1.

Таблиця 4.1 – Порівняння часу виконання

Експеримент	10	20	50	100	500	1000	5400
MongoDB	0.881	1.591	4.008	7.54	38.126	81.656	541.729
MongoDB з попереднім прорахунком	0.08	0.095	0.113	0.143	0.442	0.687	3.254
MongoDB з Elasticsearch	0.156	0.120	0.173	0.138	0.391	0.289	0.208

Результати проведених експериментів показують значні відмінності у швидкості пошуку схожих зображень між трьома підходами: використання тільки MongoDB, використання MongoDB з попереднім прорахунком векторів ознак, та використання гібридного сховища у якості комбінації MongoDB з Elasticsearch. Для розуміння залежності часу виконання пошуку від обсягів даних побудуємо графік (рис. 4.3).

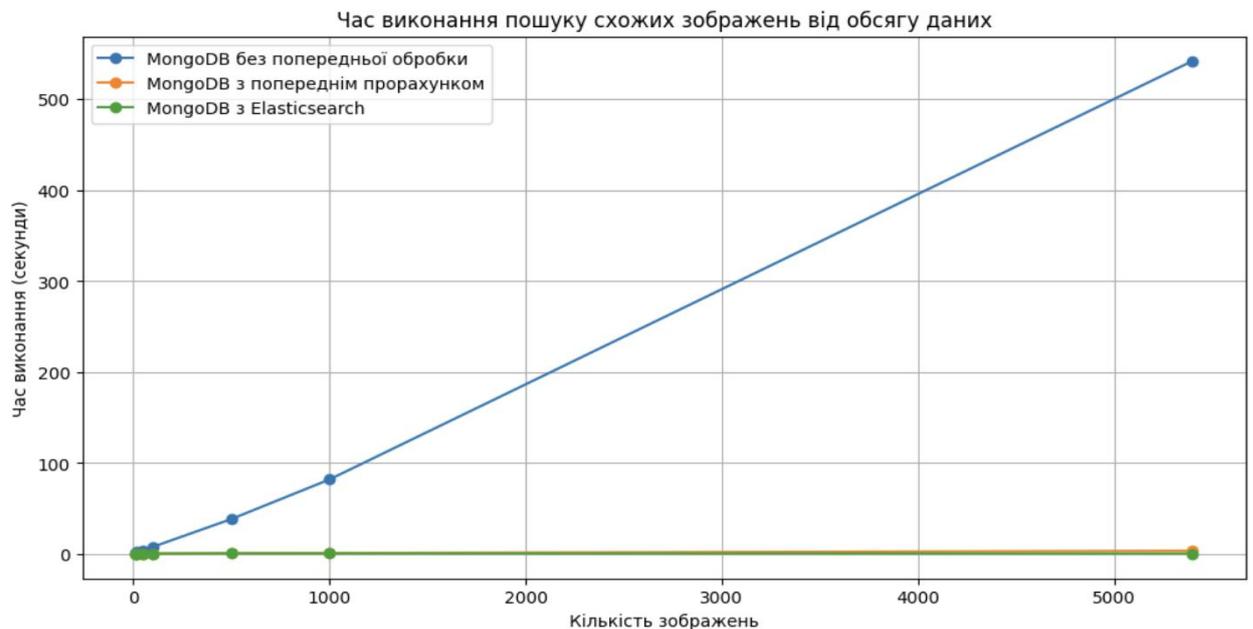


Рисунок 4.3 Графік залежності часу виконання від обсягів даних трьох підходів

На графіку видно лінійну залежність часу виконання від кількості зображень в методі використання тільки MongoDB без попереднього прорахунку. Причому даний підхід досить затратний. Це пояснюється тим, що

кожного разу під час пошуку схожих зображень система повинна перебирати всі зображення та обчислювати вектори ознак «на льоту». Це призводить до суттєвих витрат часу при великих обсягах даних. Оцінка складності алгоритму – $O(n)$. Для більш наглядної оцінки наступних двох підходів, було прийнято рішення виключити цей підхід з графіку (рис.4.4)

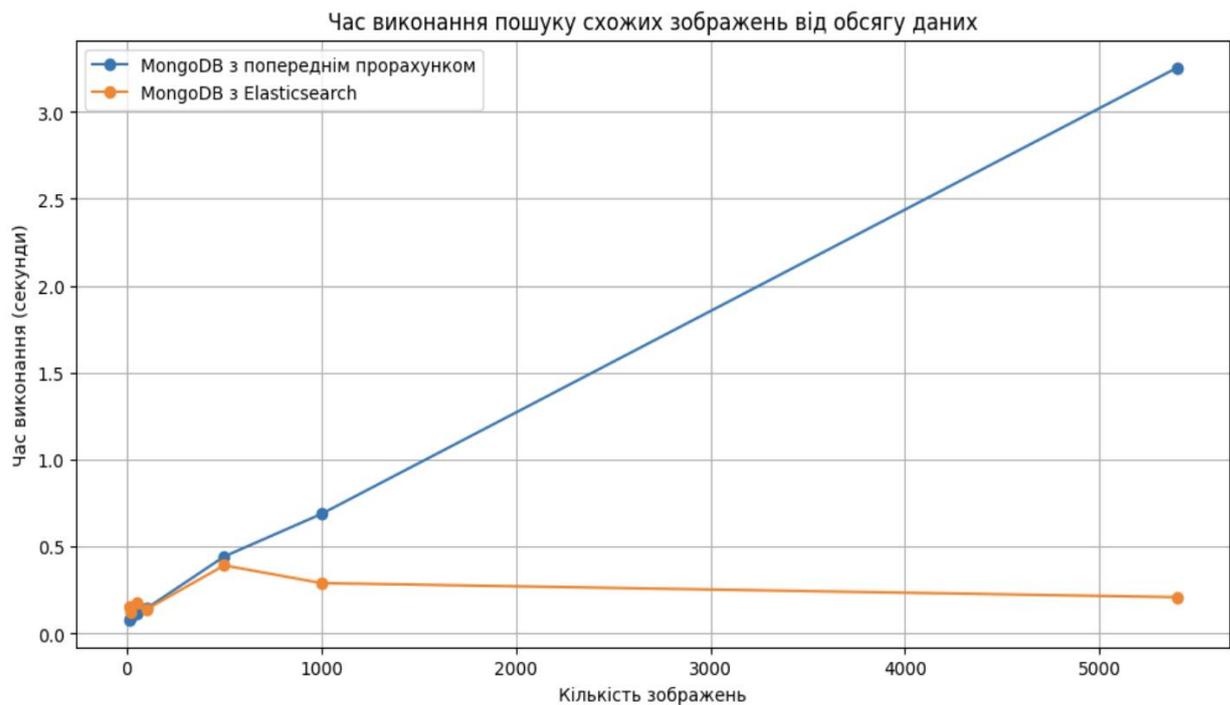


Рисунок 4.4 – Графік залежності часу виконання від обсягів даних двох підходів

Використання MongoDB з попереднім прорахунком векторів ознак значно покращує час виконання пошуку. Це пояснюється тим, що вектори ознак вже збережені у базі даних і система не витрачає час на їх обчислення під час пошуку. Можна побачити лінійне збільшення часу виконання від обсягів даних. Це пояснюється тим що виконується повний перебір в сховищі для пошуку найбільш схожих зображень. Оцінка складності алгоритму – $O(n)$.

Зовсім інша залежність з використанням гібридного сховища зображень у якості MongoDB з ElasticSearch. На результатах не видно залежності часу від обсягів даних. Це обумовлюється тим, що використання ElasticSearch для

пошуку максимально ефективно і за рахунок індексації має логарифмічну складність – $O(\log(n))$, для якої обрані дані об'єми даних значно малі.

Використання гібридного сховища у якості комбінації MongoDB та Elasticsearch дозволяє максимально використати сильні сторони кожної технології. MongoDB забезпечує гнучке та надійне зберігання даних, тоді як Elasticsearch надає потужні можливості ефективного пошуку.

В ході проведення дослідження було виконано аналіз предметної області та розглянуто можливості інтеграції машинного навчання для доступу до даних в гібридних сховищах зображення. Описано їх принципи роботи, виділено особливості, основні відмінності, переваги та недоліки. У процесі роботи поставлено задачу реалізувати алгоритм пошуку схожих зображень в сховищі, розроблена та протестована програмна частина, яка реалізує кілька підходів до зберігання та пошуку зображень, що дозволило зробити висновки щодо їх ефективності та доцільності використання в реальних умовах.

Перш за все, визначено, що традиційний підхід, який полягає у зберіганні зображень у єдиному сховищі MongoDB без попереднього розрахунку векторів ознак, є найменш ефективним з точки зору швидкості пошуку. Результати експериментів показали, що цей метод має значні затримки при обробці великих обсягів даних, що робить його непридатним для задач, де потрібна висока швидкість доступу до зображень. А при збільшенні обсягів даних час виконання пошуку збільшується, що унеможливорює пошук.

Водночас, використання підходу з попереднім розрахунком векторів ознак та їх зберіганням у MongoDB дозволяє значно скоротити час пошуку схожих зображень. Такий підхід забезпечує швидший доступ до даних за рахунок попередньої обробки зображень і збереження результатів обчислень, що робить його більш ефективним рішенням на невеликих обсягах даних. Однак даний підхід все одно має лінійну залежність часу виконання від обсягів даних, тому не може використовуватись на великих обсягах даних.

Найбільш ефективним виявився гібридний підхід, який поєднує зберігання зображень у MongoDB та векторів ознак у Elasticsearch. Завдяки

індексації і потужним можливостям пошуку досягнуто логарифмічної залежності часу виконання від обсягів даних, що робить його ефективнішим. Цей метод дозволяє не тільки швидко знайти схожі зображення, але й забезпечує гнучкість у пошукових запитах та високу точність результатів завдяки можливостям Elasticsearch. Експерименти підтвердили, що цей підхід має найнижчий час виконання запитів, особливо при обробці великих обсягів даних.

Підсумовуючи, можна сказати, що результати дослідження підтверджують доцільність використання гібридних сховищ зображень із залученням методів машинного навчання для оптимізації процесу пошуку. Використання MongoDB та Elasticsearch у поєднанні з попереднім розрахунком векторів ознак дозволяє досягти високої ефективності та точності при роботі з великими обсягами візуальних даних.

Важливим висновком є також те, що правильний вибір методів машинного навчання та технологій зберігання даних значно впливає на результати роботи системи. Використання моделі ResNet50 для екстракції ознак зображень та алгоритму косинусної подібності для порівняння векторів ознак виявилось ефективним рішенням для задачі пошуку схожих зображень.

Загалом, проведене дослідження показало, що інтеграція машинного навчання в процеси управління та обробки даних у гібридних сховищах зображень відкриває нові можливості для покращення ефективності та точності роботи з великими обсягами візуальної інформації. Зокрема, це дозволяє забезпечити швидкий та надійний доступ до необхідних даних у різних сферах застосування, включаючи медицину, безпеку та електронну комерцію. Мета завдання досягнута за рахунок визначення у роботі ефективності використання методів машинного навчання для доступу до даних в гібридних сховищах зображень. Результати цього дослідження можуть бути використані для прийняття рішення про те, який підхід використовувати для збереження і організації зображень у сховищі і який

алгоритм та модель машинного навчання використовувати для пошуку схожих зображень.

4.2 Застосування методу оптимізованої індексації в гібридній моделі зберігання зображень

Сьогодні в області медичної візуалізації в базах даних зберігаються величезна кількість записів, починаючи від стандартних рентгенівських знімків і закінчуючи складними МРТ-скануваннями. З розвитком передової діагностики ці архіви значно зросли. При розгляді інтеграції гібридної моделі зберігання на масивах даних, орієнтованих на неврологію, завдяки набору даних, що містить понад 5 мільйонів зображень, традиційні механізми пошуку показали різний час доступу залежно від типу та роздільної здатності зображення. Після інтеграції оптимізованої індексації (табл. 4.2) можна спостерігати такі результати.

Таблиця 4.2 – Спостережуваний час пошуку

Тип зображення	МРТ	РЕНТГ	КТ	УЗІ	ПЕТ
Звичайний час отримання (с)	4.5	3.0	4.2	2.7	5.5
Оптимізований час отримання	1.1	0.8	1.2	0.6	1.4
Прискорення	4,09x від ставки	3,75x від ставки	3,5x	4,5x від ставки	3,93x від ставки

У супутникових знімках, які дозволяють отримати великі обсяги фотографій з високою роздільною здатністю, що є дуже важливим для численних застосувань, таких як моніторинг навколишнього середовища. Тут традиційні механізми продемонстрували змінну затримку в залежності від регіону та часу (табл. 4.3).

Платформи електронної комерції значною мірою покладаються на ефективне зберігання та пошук зображень для користувацького досвіду. Для

платформи електронної комерції з різноманітним асортиментом продукції протестований даний метод. Час завантаження залежить від типу та категорії продукту (табл. 4.4).

Таблиця 4.3 – Затримки, що спостерігаються

Область	Північна Америка	Азія	Європа	Африка	Південна Америка
Традиційна затримка (и)	5.2	5.0	5.1	4.9	5.3
Оптимізована затримка (и) пошуку	1.2	1.0	1.1	1.0	1.3
Прискорення	4,33x від ставки	5x	4,64x від ставки	4,9x від ставки	4,08x від ставки

Таблиця 4.4 – Збільшення часу завантаження

Категорія товару	Електроніка	Одяг	Дім і життя	Книги	Продукти
Попередній час завантаження (с)	3.1	2.8	2.9	2.7	3.0
Оптимізований час завантаження (с)	0.8	0.7	0.7	0.6	0.8
Прискорення	3,88x від ставки	4x	4,14x від ставки	4,5x від ставки	3,75x від ставки

У світі, який генерує дані в геометричній прогресії, кожна частка секунди, заощаджена на пошуку, може призвести до значної економії часу в глобальному масштабі. Крім того, у зв'язку з тим, що галузі та підприємства переходять на процеси прийняття рішень на основі даних, ефективність систем зберігання та пошуку може безпосередньо впливати на результати бізнесу. Цей розділ має на меті надати порівняльний аналіз, протиставляючи оптимізований метод індексації існуючим моделям зберігання за допомогою комбінації математичних викладів, емпіричних даних і тестування.

Показники продуктивності дають кількісні характеристики ефективності систем зберігання та пошуку.

Час отримання (T) – цей показник вказує на ефективність отримання даних. Для набору даних S , що складається з n запитів, середній час отримання T можна зобразити як:

$$T = \frac{1}{n} \sum_{i=1}^n t_i, \quad (4.2)$$

де t_i – час, витрачений на запит i^{th} .

Ефективність зберігання (E) оцінює, наскільки ефективно використовується простір для зберігання. Його забезпечують:

$$E = \frac{\text{Space occupied by time}}{\text{Total allocated space}} \times 100\%. \quad (4.3)$$

Накладні витрати на індексацію (I). Це визначає обчислювальні зусилля, необхідні для самого процесу індексації. Для n зображень його можна описати так:

$$I = \frac{\text{Total indexing time for } n \text{ images}}{n}. \quad (4.4)$$

Ці математичні вирази дають об'єктивну основу для оцінки запропонованого методу. При оцінці продуктивності систем зберігання даних вибір порівняльної моделі має важливе значення. Для цього аналізу розглянемо DB-Standard [165-167], загальноприйняту систему в галузі. Використовуючи набір даних, що складається з різних типів зображень, порівнюємо продуктивність розробленої гібридної моделі зі стандартом DB.

Результати даного дослідження, узагальнені в результаті тестування та представлені в таблиці 4.5.

Таблиця 4.5 – Результати всебічного тестування

Метричний (усереднений за 10^6 операцій)	БД-Стандарт	Гібридна модель	Приріст продуктивності
Час отримання (T) у секундах	4.5	1.05	4,29x від ставки
Ефективність зберігання (E) у %	70%	91%	1,3x
Індексація накладних витрат (I) у секундах	0.08	0.019	4,2x

Розширюючи результати аналітичного дослідження, зосередимо увагу на практичному застосуванні розробленої системи в контексті обробки супутникових знімків. Значні обсяги даних, характерні для супутникової зйомки, створюють оптимальні умови для оцінки ефективності розроблених методів зберігання та пошуку даних. Специфіка супутникових зображень, які характеризуються високою роздільною здатністю та мультиспектральністю, дозволяють всебічно оцінити продуктивність запропонованої системи в умовах обробки великомасштабних наборів даних.

```
# DB-Standard method
def standard_retrieve(region, timestamp):
    # Simulating an intricate query process
    db_query = f"SELECT image FROM DB_Standard WHERE region='{region}' " +
    + "AND timestamp='{timestamp}'"
    # Fetching the image (for representation purposes)
    image = db.execute(db_query)
    return image

# Hybrid-Model method
def optimized_retrieve(region, timestamp):
    # Applying optimized indexing for efficient querying
    index = computeOptimizedIndex(region, timestamp)
    # Directly accessing the image using the index
    image = hybrid_storage.retrieve(index)
    return image
```

Рисунок 4.5 – Додаток для отримання супутникових зображень

Можна дійти висновку, що завдяки ретельному тестуванню, гібридна модель незмінно перевершувала DB-Standard, досягаючи швидшого отримання даних, оптимізації сховища та мінімізації обчислювальних витрат. Цей порівняльний аналіз підкреслює надійність оптимізованого методу індексації, підтверджуючи його корисність для різноманітних застосувань.

4.3 Використання блокчейн-рішень для захисту цифрових активів

У цьому підрозділі монографії пропонується розглянути практичне використання сучасних блокчейн рішень та їхнє потенційне впровадження в

контексті захисту цифрових активів. Сьогодні досліджуються різні аспекти технологій блокчейн, включаючи механізми консенсусу, рівень безпеки, та функціональні можливості. Зі зростанням використання цифрових активів у всіх сферах діяльності, від фінансів до технологій та медицини, виникає необхідність ефективного захисту цих активів від кіберзагроз. Кібератаки стають все більш вдосконаленими, і традиційні методи безпеки часто виявляються недостатніми перед новітніми загрозами. У цьому контексті блокчейн-технології набувають значущості, пропонуючи новий підхід до забезпечення безпеки цифрових активів. Блокчейн сьогодні став платформою для розвитку різних застосувань у сфері кібербезпеки, оскільки його основна сутність полягає в тому, що дані розподілені та не можуть бути змінені без згоди всіх учасників мережі. Це робить його вкрай важливим інструментом для захисту цифрових активів. Блокчейн використовує криптографічні методи для створення безпечних та недоступних для зміни блоків даних, що зберігаються в розподіленій мережі. Це дає високий рівень безпеки та надійності. Історія транзакцій у ланцюжку блоків знаходиться у кожного учасника системи, і будь-яка спроба змінити існуючий блок вимагає консенсусу більшості мережі [168].

За останні роки було розроблено безліч блокчейн-рішень, спрямованих на різні аспекти захисту цифрових активів. Від платіжних систем із використанням смарт-контрактів до систем ідентифікації осіб і управління доступом, блокчейн знайшов своє застосування у багатьох областях. Один із важливих напрямків – це використання блокчейну для створення децентралізованих ідентифікаційних систем. Такі системи зменшують ризик витоку особистих даних, оскільки кожен користувач має контроль над своєю ідентичністю. Смарт-контракти в блокчейні можуть забезпечити безпеку та автоматизацію виконання угод, уникнувши інтервенції третіх сторін.

Ще однією важливою характеристикою блокчейну є його прозорість. Відкритість та доступність для всіх учасників мережі дозволяють ефективно виявляти та вирішувати потенційні проблеми безпеки. Колективна природа

блокчейн-мережі дозволяє виявляти аномалії та кібератаки швидше, порівняно з традиційними централізованими системами. Безпека також підсилюється тим, що дані в блокчейні зберігаються децентралізовано та не мають єдиного центрального пункту вразливості. Зламання одного вузла не призводить до втрати даних чи порушення безпеки всієї мережі. Глибокий аналіз різноманітних блокчейн-рішень, призначених для захисту цифрових активів, є важливою частиною такого захисту. Завдання включають детальний огляд різних блокчейн-протоколів, їхніх переваг та обмежень у контексті кібербезпеки. Дослідження також орієнтоване на ідентифікацію найефективніших та найбільш підходящих рішень для різних сфер використання цифрових активів. Важливим стає розуміння та використання комплексних блокчейн-технологій для захисту цифрових активів [168].

Інноваційність блокчейн технологій у захисті цифрових активів визначає нові горизонти для фінансової та технологічної революції. Аналіз існуючих рішень та визначення перспектив показують, що виникаючі виклики можуть стати стимулом для подальшого розвитку та впровадження інноваційних технологій, що покращать захист цифрових активів і розширять можливості їхнього використання.

На сьогоднішній день існує значна кількість досліджень та наукових публікацій, що спрямовані на вивчення та вдосконалення блокчейн-технологій для захисту цифрових активів. Дослідження у роботі [169] представляє важливий внесок у розуміння ефективності блокчейну у сфері кібербезпеки. Автори акцентують на тому, що блокчейн може стати ефективним інструментом для підвищення рівня захисту від різноманітних кіберзагроз. Дослідження розглядає роль блокчейну у забезпеченні цілісності та конфіденційності даних, а також виявлення та мінімізації ризиків кібератак. Основні висновки даного дослідження стверджують, що застосування блокчейну дозволяє створити безпечну та невразливу інфраструктуру для зберігання та обробки цифрових активів. Важливо враховувати, що блокчейн

не є універсальним засобом захисту і повинен бути інтегрованим у комплексні кіберзахисні стратегії.

Дослідження [170] вивчає різноманітні блокчейн-протоколи та їхню ефективність у захисті цифрових активів. Автори докладно аналізують протоколи, такі як Bitcoin, Ethereum, та розглядають їхню застосовність у конкретних випадках використання, починаючи від фінансових транзакцій до управління правами доступу. Висновки з даного дослідження вказують на різноманітність блокчейн-протоколів та їхню здатність адаптуватися до різних потреб у сфері захисту цифрових активів. Однак автори роблять акцент на необхідності вибору правильного протоколу залежно від конкретного застосування та бізнес-вимог. Оглядаючи обидві публікації, можна зробити важливі висновки. По-перше, блокчейн-технології виявляються ефективним інструментом для захисту цифрових активів, зокрема у сфері кібербезпеки. По-друге, різноманітність блокчейн-протоколів надає можливість вибрати найбільш доцільний протокол для конкретних вимог та умов застосування.

На основі аналізу роботи [171] можна визначити дві основні тенденції. По-перше, зростаючий інтерес до блокчейну як ефективного інструменту для захисту цифрових активів. По-друге, постійний пошук оптимальних блокчейн-рішень, які враховують конкретні потреби та вимоги різних галузей. Дослідження та публікації в галузі використання блокчейну для захисту цифрових активів вказують на великий потенціал цієї технології. Проте, важливо пам'ятати, що успіх впровадження блокчейну залежить від правильного вибору протоколу та врахування специфічних вимог кожного конкретного застосування. У подальших дослідженнях і розробках важливо спрямовувати зусилля на оптимізацію та адаптацію блокчейн-рішень для різноманітних викликів у галузі цифрових активів та кібербезпеки.

У сучасному цифровому світі, де цифрові активи стають ключовим елементом економіки та підвищуються кількісно та якісно, захист їх від кіберзагроз стає актуальною та нагальною проблемою [172]. Незважаючи на широке застосування різних методів та засобів кібербезпеки, існують

невирішені аспекти, які вимагають новаторського підходу. Існуючі блокчейн-рішення часто стикаються з проблемами, пов'язаними з анонімністю та конфіденційністю. Розголошення особистої інформації та витік конфіденційної інформації залишаються невирішеними питаннями, які обмежують придатність блокчейн-технологій у деяких сферах. З точки зору швидкості та масштабованості, існуючі рішення можуть виявитися неефективними у великих мережах або при великому обсязі транзакцій. Вирішення цих питань вимагає подальших досліджень та інновацій. Відсутність єдиних стандартів взаємодії блокчейн-мереж і недостатнє регулювання в галузі криптовалют та цифрових активів у багатьох країнах створюють правові та економічні ризики для учасників цієї екосистеми [173].

Проведений аналіз сучасних публікацій та наукових робіт, які стосуються використання блокчейн-технологій у захисті цифрових активів, виявив їхні слабкі та не вирішені аспекти. Огляд існуючих блокчейн-рішень, зокрема Bitcoin, Ethereum та Hyperledger Fabric, виявив їхні переваги та недоліки. Для вирішення поставлених завдань використовувалися методи аналізу блокчейн-протоколів, експертні оцінки ефективності захисту цифрових активів та вивчення технічних особливостей кожного рішення. Початком був систематичний огляд різних блокчейн-протоколів, що включає Bitcoin, Ethereum, Hyperledger та інші. Проводився докладний розгляд їхньої структури, принципів консенсусу, масштабованості та рівня безпеки. Цей етап включав аналіз транзакційної швидкості, можливостей смарт-контрактів та механізмів консенсусу кожного протоколу. Експерти з кібербезпеки використовували свої знання для проведення оцінки ефективності блокчейн-рішень у захисті цифрових активів. Застосовувалися методи експертної оцінки, враховуючи такі аспекти, як витрати на енергію, можливість виявлення та протидії кібератакам, анонімність та інші параметри [174].

Bitcoin блокчейн-мережа є децентралізованою системою для обміну цифрових валют. Цей блокчейн використовується для запису та підтвердження всіх транзакцій, виконаних у криптовалюті Bitcoin (BTC).

Основна мета – забезпечити безпеку та довіру у фінансових операціях, уникаючи потреби у централізованому фінансовому посередництві.

Ключові риси Bitcoin блокчейн-мережі:

- децентралізація – відсутність центрального органу контролю. Всі вузли (комп'ютери) мережі рівноправні.
- транзакції – кожна транзакція відзначається у блоках та підтверджується мережею за допомогою процесу, відомого як «майнінг».
- майнінг – процес, при якому майнери вирішують складні математичні завдання для підтвердження транзакцій та створення нових блоків.
- криптографія – використання криптографічних методів для забезпечення безпеки та анонімності транзакцій.
- блокчейн – сполучення блоків транзакцій у ланцюг, де кожен блок містить хеш попереднього, створюючи непорушний запис.
- доступність – можливість використання Bitcoin для переказу вартості навіть без звичайного банківського облікового запису.

Bitcoin блокчейн-мережа визнана своєю надійністю та відсутністю централізованого контролю, забезпечуючи глобальний доступ до фінансових послуг та прозорість операцій.

Ethereum – це децентралізована блокчейн-мережа, яка не лише дозволяє передавати цифрові валюти (ETH), але і використовується для розгортання смарт-контрактів та децентралізованих додатків (DApps). Розроблена для створення більш широкого спектру децентралізованих послуг та платформ.

Основні характеристики Ethereum блокчейн-мережі:

- смарт-контракти – програми, що автоматизують та виконують угоди на основі умов, записаних у блокчейні, розширюючи можливості використання технології.
- ефір (Ether) – офіційна криптовалюта Ethereum, використовується для оплати транзакцій та винагороди майнерам.

– децентралізовані додатки (DApps) – платформа, яка дозволяє розробникам створювати додатки без централізованого управління чи контролю.

– майнінг – процес вирішення складних завдань для підтвердження транзакцій та створення нових блоків.

– ERC-20 токени: стандарт для створення та взаємодії з токенами на основі Ethereum, що розширює функціонал мережі.

– Децентралізовані Автономні Організації (DAO) – структури, що дозволяють спільноті приймати рішення та керувати ресурсами без централізованого управління.

– технічна гнучкість – Ethereum надає можливість впроваджувати та вдосконалювати протоколи через хардфорки (hard forks).

Ethereum відомий своєю гнучкістю та можливістю створення складних додатків та фінансових інструментів, що виходять за межі простого обміну вартості [174].

Hyperledger Fabric – це високопродуктивний, децентралізований блокчейн-протокол, призначений для використання у корпоративних середовищах. Він розроблений Linux Foundation і надає потужність для розробки децентралізованих додатків та розгортання смарт-контрактів у бізнес-середовищах.

Основні особливості Hyperledger Fabric:

– дозвілля та приватність – Hyperledger Fabric дозволяє налаштовувати рівень доступу та конфіденційності для різних учасників мережі, забезпечуючи дозвіл на конкретні транзакції.

– модульність та гнучкість – протокол дозволяє використовувати різні алгоритми консенсусу, розумні контракти та системи легітимації, щоб адаптувати мережу до конкретних потреб підприємства.

– колекції даних – Hyperledger Fabric підтримує колекції даних, що дозволяють групам учасників обмінювати обмежені дані без їхньої розкритості всій мережі.

– легкість управління – мережі Hyperledger Fabric можна легко розгортати та керувати, забезпечуючи ефективне управління з оглядом на бізнес-процеси.

– підтримка компаній – розроблений з урахуванням потреб підприємств, Hyperledger Fabric дозволяє створювати додатки для підприємств та забезпечує високу пропускну здатність.

– довіра та безпека – мережі Hyperledger Fabric використовують консенсус за допомогою практик Proof-of-Work, що робить їх надійними та стійкими до змов та атак.

Hyperledger Fabric становить популярний вибір для підприємств, які шукають блокчейн-рішення, орієнтовані на децентралізацію та безпеку для своїх бізнес-операцій.

Вивчення технічних особливостей кожного рішення включало в себе розгляд конфігурацій блокчейн-мережі, алгоритмів шифрування, механізмів контролю доступу та систем управління ключами. Зокрема, зверталася увага на можливість інтеграції з існуючими системами, адаптивність до змін в кіберзагрозах та здатність до масштабування [175]. Сучасний розвиток технологій ставить перед собою завдання удосконалити процеси обробки, зокрема у питаннях аналізу тексту в контексті семантичного аналізу [2]. Дослідження по можливому використанню існуючих блокчейн рішень у семантичній обробці тексту та їхній ролі в класичному процесі обробки тексту зведено у Таблиці 4.6 [168].

Таблиця 4.6 – Блокчейн та семантична обробка тексту

	Bitcoin	Ethereum	Hyperledger Fabric
Механізми консенсусу та передовий семантичний аналіз	Bitcoin PoW – може використовуватися для підтвердження джерела інформації в семантичному аналізі, забезпечуючи надійність даних.	Ethereum PoS: забезпечує швидку та енергоефективну обробку тексту в семантичних моделях.	Hyperledger Fabric RBFT – дозволяє побудувати обмежену мережу для обміну семантичною інформацією у корпоративних середовищах.
Смарт-контракти та автоматизований семантичний аналіз	використовуються для автоматизації семантичного аналізу	Ethereum смарт-контракти – використовуються для автоматизації семантичного аналізу та виконання різних завдань з обробки тексту без посередництва.	Hyperledger Fabric модульність – дозволяє розробляти та використовувати модульні рішення для спеціалізованого семантичного аналізу.
Збір та підготовка даних	Забезпечують безпечний та надійний обмін текстовою інформацією за допомогою криптографічно захищених транзакцій.	Забезпечують безпечний та надійний обмін текстовою інформацією за допомогою криптографічно захищених транзакцій.	Гарантує приватність та конфіденційність даних під час їх збору та обміну.
Аналіз та витягання інформації	Використання шляхом дослідження транзакційної історії. Публічний характер блокчейну дозволяє переважно створювати та використовувати аналітичні інструменти для вивчення та розуміння руху криптовалют та пов'язаних з ними даних.	Використовуються для автоматичного витягнення та обробки семантичної інформації з тексту.	Дозволяє забезпечити надійність та цілісність інформації під час аналізу.

Моделювання та оптимізація	Допомагає забезпечити розподілену обробку та оптимізацію семантичного аналізу.	Забезпечує швидкість та високу масштабованість для оптимального моделювання текстової інформації.	Дозволяє оптимізувати семантичний аналіз, використовуючи спеціалізовані модулі.
----------------------------	--	---	---

Використання існуючих блокчейн рішень може значно покращити процес семантичного аналізу тексту. Bitcoin, Ethereum та Hyperledger Fabric пропонують унікальні можливості для безпечного та ефективного обміну семантичною інформацією, а їхні механізми консенсусу та смарт-контракти додають новітні функціональності до класичного процесу обробки тексту. Впровадження цих технологій може визначити нові стандарти в галузі семантичного аналізу та покращити надійність та ефективність обробки текстової інформації.

Зберігання та обмін знань також має ключове значення, особливо в програмних системах, де використання баз знань є невід'ємною частиною розробки, що є дуже важливим у сучасному світі [176]. Аналіз сучасних публікацій вказує на можливість використання існуючих блокчейн рішень для оптимізації та підвищення ефективності повторного використання знань у програмних системах на основі баз знань, аналіз представлений у Таблиці 4.7 [168].

Таблиця 4.7 – Блокчейн та його роль у системах знань

	Bitcoin	Ethereum	Hyperledger Fabric
Децентралізація зберігання Знань	Можуть слугувати основою для децентралізованої бази знань, забезпечуючи високу доступність та стійкість до втрати інформації.	Можуть слугувати основою для децентралізованої бази знань, забезпечуючи високу доступність та стійкість до втрати інформації.	Може служити основою для створення децентралізованої системи зберігання знань, де інформація розподіляється між учасниками мережі. Смарт-контракти та модульна архітектура дозволяють створювати ефективні механізми

Смарт-контракти та автоматизована логіка	Обмежений основними функціями переказу коштів, менше підходить для реалізації складних автоматизованих логік.	Використовуються для автоматизації процесів оновлення та розширення бази знань без необхідності централізованого втручання.	обміну та верифікації знань без централізованого контролю.
Механізми консенсусу та гарантована цілісність даних	Використовує механізм консенсусу Proof of Work (PoW), де майнери розв'язують складні математичні завдання для підтвердження транзакцій. Це забезпечує високий рівень цілісності даних та довіреність шляхом децентралізованого підтвердження операцій.	Використовує Proof of Stake (PoS) та планує перейти на Proof of Stake (PoS), що зменшить енерговитрати. Система смарт-контрактів Ethereum також допомагає гарантувати цілісність даних, дозволяючи програмовані правила виконання операцій у децентралізованому середовищі.	Забезпечує високий рівень гарантії цілісності та достовірності знань завдяки механізму консенсусу.
Збір та структурування знань	Дозволяє структурувати та збирати знання в блокчейні, забезпечуючи їхню невразливість до видалення чи зміни.	Дозволяє структурувати та збирати знання в блокчейні, забезпечуючи їхню невразливість до видалення чи зміни.	Допомагає створювати та управляти модульними блоками знань.
Обмін та розповсюдження знань	Може використовуватися для обміну та розповсюдження знань через мікроплатежі.. Можливість невеликих транзакцій дозволяє користувачам	Забезпечує автоматичний обмін та розповсюдження знань згідно з умовами смарт-контрактів.	Забезпечує безпечний та конфіденційний обмін знань в корпоративних умовах.

	оплачувати за контент, інформацію або послуги, сприяючи екосистемі децентралізованого обміну знань.		
Аналіз та оптимізація знань	Гарантує, що знання в блокчейні залишаються не впливованими та стійкими до атак.	Забезпечує енергоефективніше управління та оптимізацію знань в реальному часі.	Може бути використаний для створення децентралізованої системи аналізу та оптимізації знань. Забезпечуючи безпеку та конфіденційність, він дозволяє ефективно обробляти та обмінюватися інформацією між учасниками, сприяючи аналітиці та покращенню використання знань в організації.

Bitcoin, Ethereum та Hyperledger Fabric додають важливі внески, дозволяючи створювати, розповсюджувати та оптимізувати знання в безпечних, ефективних та децентралізованих умовах. Ця інноваційна архітектура може визначити нові стандарти в управлінні знаннями та підвищити продуктивність у сфері розробки програмного забезпечення. Треба також зазначити, що сучасні технології, зокрема блокчейн, вносять значний внесок у різні галузі, включаючи аналітику, електронне навчання (E-Learning) та обробку великих обсягів даних (Big Data) [88]. Порівняльний аналіз подається у Таблиці 4.8 [168].

Таблиця 4.8 – Аналітика, E-Learning та Big Data в блокчейн

	Bitcoin	Ethereum	Hyperledger Fabric
Використання механізмів консенсусу для достовірності	Забезпечує достовірність даних у системах аналітики, де	Використовує механізм Proof of Stake (PoS), де учасники, що мають	Використовує механізм Practical Byzantine Fault Tolerance (PBFT), де

	Bitcoin	Ethereum	Hyperledger Fabric
	важлива надійність інформації.	стейк (в криптовалюти), мають шанс додати новий блок до ланцюга. Це забезпечує достовірність та безпеку мережі за рахунок зацікавленості учасників у її надійності.	учасники досягають консенсусу шляхом взаємодії та підтвердження операцій. Цей метод забезпечує високий рівень достовірності, так як він узгоджується навіть при наявності нечесних учасників у мережі.
Захист та надійність даних	Використовує Proof of Work (PoW) для забезпечення надійності даних. Майнери вирішують складні математичні завдання, що підтверджує валідність транзакцій та забезпечує надійність блокчейну через децентралізований консенсус.	Дозволяють створювати автоматизовані, захищені контракти для обміну та обробки аналітичних даних.	Використовує Practical Byzantine Fault Tolerance (PBFT) та інші механізми консенсусу. Це дозволяє досягти високого рівня надійності та захисту даних в умовах децентралізованої мережі, навіть при можливих нечесних учасниках.
E-Learning	Може бути використаний у галузі для мікроплатежів, підтримки контенту та винагороди творців змісту. Можливість миттєвих транзакцій дозволяє створити ефективну систему оплати за навчальний матеріал.	Може бути використаний у галузі для мікроплатежів, підтримки контенту та винагороди творців змісту. Можливість миттєвих транзакцій дозволяє створити ефективну систему оплати за навчальний матеріал.	Забезпечує децентралізовану атестацію та зберігання документів у сфері E-Learning.
Відстеження прогресу та успішності	Використовується для створення непідробних та відстежуваних записів про успішність студентів.	Використовується для створення непідробних та відстежуваних записів про успішність студентів.	Може служити основою для створення децентралізованої системи відстеження прогресу та успішності студентів. Смарт-контракти та

	Bitcoin	Ethereum	Hyperledger Fabric
Big Data	Може бути використаний для забезпечення безпеки та відстеження транзакцій. Може слугувати надійною основою для реєстрації та аудиту великих обсягів фінансових даних.	Забезпечує захищений обмін великих обсягів персональних даних в системах Big Data, дотримуючись принципів конфіденційності.	Може використовуватись для створення безпечних та прозорих мереж обміну даними між децентралізованими сторонами. Механізми консенсусу та конфіденційність Hyperledger Fabric роблять його відмінним вибором для ефективної обробки та обміну великими обсягами даних.
Децентралізоване зберігання та обробка даних	Використовується для децентралізованого зберігання та обробки фінансових даних. Транзакції в блокчейні Bitcoin дозволяють ефективно та безпечно здійснювати операції із збереженням історії та цілісністю даних.	Використовується для децентралізованого зберігання та обробки різноманітних даних, включаючи смарт-контракти. Система смарт-контрактів дозволяє автоматизовано обробляти дані та взаємодіяти з різними додатками у безпечному середовищі.	Дозволяє розгортання спеціалізованих модулів для ефективної обробки великих обсягів даних у різних галузях (модульність).

Результати включають аналіз використання різних блокчейн-рішень в реальних умовах. Розглянуті такі аспекти, як швидкість транзакцій, рівень

анонімності, масштабованість та інші параметри, що впливають на безпеку цифрових активів [177]. Вибір конкретних блокчейн-рішень здійснений на підставі попереднього аналізу, де були враховані їхні технічні особливості та ефективність. Для експерименту були обрані три різні блокчейн-протоколи – Bitcoin, Ethereum та Hyperledger Fabric. Для кожного обраного блокчейн-протоколу проведено розгортання мережі в контрольованому середовищі. Забезпечено надійність та безпеку роботи кожної мережі, з урахуванням можливих загроз та ризиків. У ході експерименту велося постійне спостереження за роботою блокчейн-мереж, з фіксацією та аналізом кожної транзакції. Зазначена необхідність впровадження інструментів моніторингу для визначення швидкості обробки транзакцій та реакції мережі на високі навантаження. Визначені та реалізовані різні сценарії використання для кожного блокчейн-протоколу. Це включало в себе проведення фінансових транзакцій, використання смарт-контрактів, а також взаємодію з іншими технологіями та сервісами.

Особлива увага приділена аналізу масштабованості кожної блокчейн-мережі. Аналізувалося збільшення навантаження для визначення, наскільки ефективно система справляється з ростом обсягу транзакцій та забезпечує стабільність роботи. Для блокчейн-мереж, які підтримують анонімні транзакції, було проведено вимірювання рівня анонімності. Вивчалися можливості інкогніто-транзакцій та ефективність захисту особистих даних у контексті різних сценаріїв використання [178]. Треба зазначити, що збір даних повинен включати в себе логи транзакцій, показники ефективності мережі, витрати на енергію та інші ключові параметри. Дані були систематично зібрані для подальшого детального аналізу. На підставі цього аналізу отримані результати, які дозволили зробити об'єктивні висновки щодо ефективності кожного блокчейн-протоколу в реальних умовах. Оцінювалися переваги та обмеження кожної системи з точки зору захисту цифрових активів та кібербезпеки. Все це підтвердило ефективність використання блокчейн-рішень для захисту цифрових активів. Найбільш перспективними виявились

рішення, які поєднують високу швидкість транзакцій з високим рівнем безпеки та анонімності [168].

Один із ключових параметрів – швидкість транзакцій. Bitcoin, Ethereum та Hyperledger Fabric були піддані ретельному аналізу.

Параметри та аналіз блокчейн-мереж наведено нижче.

Bitcoin

– швидкість – має середню швидкість обробки транзакцій близько 7 транзакцій за секунду (TPS);

– механізм консенсусу – Proof-of-Work (PoW) – споживчий за ресурсами та часом, що обмежує швидкість обробки;

– аналіз – обмеження: PoW та блокчейн-розмір обмежують швидкість.

Більшість визначених транзакцій призводять до збільшення часу обробки.

Ethereum

– швидкість – може обробляти більше транзакцій, приблизно від 15 до 45 TPS. Однак швидкість може змінюватися в залежності від об'єму та складності транзакцій;

– механізм консенсусу – PoW (поки що), але планується перехід на Proof-of-Stake (PoS), що може покращити швидкість;

– аналіз – покращення: планований перехід на PoS може підвищити швидкість транзакцій та зменшити витрати енергії. Оптимізації в майбутньому можуть додатково поліпшити продуктивність.

Hyperledger Fabric

– швидкість – розроблений з урахуванням підприємницьких потреб та може обробляти від 1,000 до 4,000 TPS, залежно від конфігурації та оптимізації;

– механізм консенсусу – Pluggable consensus – можливість вибору між різними алгоритмами консенсусу, що може підвищити швидкість;

– аналіз – підприємницька орієнтованість: швидкість Hyperledger Fabric робить його більш придатним для підприємств, де потрібно.

Можна зробити такі висновки. Bitcoin та Ethereum підходять для різних сценаріїв, від цифрового золота до смарт-контрактів, але обидва мають обмеження швидкості. Hyperledger Fabric, завдяки своїй великій швидкості та гнучкості, виглядає більш привабливим для використання в корпоративних середовищах, де потрібна висока продуктивність та адаптованість до підприємницьких вимог. Результати показали, що Hyperledger Fabric виявився найшвидшим у виконанні транзакцій, особливо при великому обсязі операцій. Ethereum також представив високі показники, в той час як Bitcoin демонстрував відносно низьку швидкість у порівнянні з іншими. З погляду безпеки і анонімності, усі три блокчейн-протоколи демонстрували високий рівень захисту даних. Однак Hyperledger Fabric виділяється в цьому контексті, забезпечуючи рівень конфіденційності та аутентифікації, що дозволяє контролювати доступ до інформації в мережі. Ethereum також виявився досить ефективним у забезпеченні анонімності та безпеки, а Bitcoin, як відомо, має свої особливості, зберігаючи відкритий реєстр транзакцій [179].

Важливим аспектом є масштабованість систем. У цьому контексті Hyperledger Fabric та Ethereum продемонстрували високий рівень масштабованості, здатність ефективно обробляти збільшення обсягу транзакцій без втрати продуктивності. Bitcoin, з іншого боку, має обмежену масштабованість через свою архітектуру. У контексті реальних сценаріїв використання, усі три блокчейн-протоколи виявилися досить ефективними. Вони успішно виконували фінансові транзакції, управляли смарт-контрактами та інтегрувалися з іншими технологіями. Однак Hyperledger Fabric відрізнявся своєю гнучкістю та адаптивністю до різних сценаріїв використання. Захищеність протоколів Bitcoin, Ethereum та Hyperledger Fabric відіграє важливу роль у їхній успішності і прийнятті у світі криптовалют і блокчейн-технологій. Розглянемо основні аспекти захищеності кожного з цих протоколів [180].

Алгоритм консенсусу Bitcoin використовує Proof-of-Work (PoW), що робить мережу стійкою до атак, таких як подвійне витрачання. PoW потребує великої обчислювальної потужності для атак, але зараз існують побоювання щодо централізації майнінгу. Використання криптографічних хеш-функцій (SHA-256) існує для підпису транзакцій та забезпечення безпеки мережі. Bitcoin – проект з відкритим кодом, що сприяє виявленню і виправленню потенційних вразливостей спільнотою. Алгоритм консенсусу Ethereum використовує Proof-of-Stake (PoS) та планує перехід на Ethereum 2.0. PoS спрямований на зменшення витрат енергії, але його ефективність і безпека ще обговорюються. Ethereum дозволяє виконання смарт-контрактів, що може стати причиною вразливостей і потребує додаткової уваги до безпеки коду. Ethereum активно розвивається, але оновлення можуть створювати ризики (наприклад, хардфорки), які вимагають уважної реалізації. Hyperledger Fabric орієнтований на підприємства та використовує дозволи. Кожен учасник має визначений рівень доступу, що робить мережу менш вразливою до атак внутрішнього характеру. Використання системи прав доступу до каналів і смарт-контрактів допомагає контролювати доступ до даних функціональності. Hyperledger Fabric дозволяє створювати модульні рішення, а також надає інструменти для забезпечення конфіденційності даних [168].

Перехід на Ethereum 2.0 потребує ретельного тестування. Смарт-контракти мають такі переваги: функціональність для їх виконання, можливість створення різноманітних децентралізованих додатків. Також можна відмітити й недоліки: ризик вразливостей та багів в кодї смарт-контрактів, можливість атак на рівень протоколу через смарт-контракти. Для Hyperledger Fabric можна зазначити такі переваги: кожен учасник має визначений рівень доступу, що підвищує захищеність мережі, а також можливість створення приватних каналів для конфіденційності. Та існують і недоліки як питання щодо децентралізації, оскільки Hyperledger Fabric є дозволеною мережею. По системі прав доступу є такі переваги як гнучка система управління правами доступу, зменшення ризику внутрішніх загроз.

По модульності та приватності відзначені переваги такі як можливість розробки модульних рішень та забезпечення приватності даних через конфіденційні канали. Недоліком є складність реалізації та конфігурації модульних рішень [168].

Тут ми маємо прийти до певних висновків. Порівняльний аналіз механізмів консенсусу показує, що кожен протокол має свої переваги та недоліки. Bitcoin визначається високою стійкістю, Ethereum намагається зменшити витрати енергії, а Hyperledger Fabric акцентується на дозволеності та модульності. Важливо враховувати специфіку проєкту та його потреби при виборі механізму консенсусу для забезпечення найвищого рівня захищеності. У кожного протоколу є своя специфіка і вибір залежить від конкретних потреб та вимог проєкту. Загальний порівняльний аналіз дозволяє краще розуміти сильні та слабкі сторони кожного протоколу в контексті захищеності. На загальний висновок впливають різні фактори, такі як специфіка завдань, розмір та характер цифрових активів.

Hyperledger Fabric виявився більш універсальним та пристосованим до різних потреб, особливо в галузі корпоративних застосувань. Ethereum і Bitcoin також мають свої сильні сторони, проте їхнє використання може бути більше обмеженим в залежності від конкретних вимог. З урахуванням результатів експерименту, варто звернутися до пошуку нових можливостей для оптимізації та розвитку блокчейн-рішень для захисту цифрових активів. Додаткові дослідження та вдосконалення можуть покращити ефективність та розширити сферу застосування цих технологій. У цілому, результати експерименту дозволяють зробити висновок, що блокчейн-технології є багатообіцяючим інструментом для захисту цифрових активів, і вибір конкретного протоколу повинен залежати від конкретних потреб та вимог конкретного сценарію використання [181].

Аналіз результатів дозволив визначити переваги та недоліки кожного з вивчених блокчейн-рішень. Важливим аспектом є також впровадження

стандартів та регулювань для забезпечення взаємодії різних блокчейн-протоколів та забезпечення узгодженості із законодавством.

Підсумовуючи, можна охарактеризувати переваги та недоліки кожного блокчейн-протоколу.

Bitcoin. Переваги – відзначається високою стійкістю до кібератак, але його основний недолік – обмежена швидкість транзакцій, що робить його менш придатним для великого обсягу транзакцій. Недолік – відкритий характер реєстру транзакцій призводить до меншої анонімності.

Ethereum. Переваги – вражає гнучкістю та можливістю використання смарт-контрактів. Висока швидкість транзакцій робить його привабливим для різних сценаріїв використання. Недоліки – збільшення обсягу транзакцій може призвести до зниження продуктивності, а також є проблеми з приватністю деяких операцій.

Hyperledger Fabric. Переваги – висока швидкість та конфіденційність роблять Hyperledger Fabric ідеальним в корпоративному середовищі. Гнучкість та можливість налаштування дозволяють адаптувати його до різних вимог. Недоліки – важкість інтеграції з іншими блокчейн-мережами та можливість впливу обмежень на децентралізацію.

Впровадження стандартів для взаємодії різних блокчейн-протоколів є важливим кроком. Це дозволяє підвищити сумісність та обмін інформацією між різними системами, роблячи їх більш універсальними та ефективними. В контексті зростаючої популярності блокчейн-технологій, регулювання стає необхідним для забезпечення узгодженості із законодавством та забезпечення безпеки. Це може включати в себе визначення правил для використання смарт-контрактів, захисту конфіденційності та управління кібербезпекою [168].

Потреба в адаптації. З огляду на постійний розвиток кіберзагроз та технологічних інновацій, блокчейн-рішення повинні бути гнучкими та готовими до адаптації до нових викликів. Вдосконалення та оптимізація є ключовими факторами для тривалого успіху. Спрямованість на подальші дослідження та розробки є важливою для вдосконалення ефективності

блокчейн-технологій. Дослідження можуть включати розробку нових протоколів, визначення кращих практик у використанні та регулюванні.

Блокчейн-технології мають великий потенціал для захисту цифрових активів. Однак їхній успіх обумовлений перш за все правильним вибором протоколу, розробкою стандартів та регулювань, а також готовністю до постійного розвитку та адаптації до змін у кіберпросторі. Глобальне співтовариство, регулятори та технологічні компанії повинні працювати разом для створення безпечного та ефективного цифрового середовища, в якому блокчейн може відігравати ключову роль [182].

Впровадження стандартів та регулювань є критичним для успішної інтеграції та розвитку блокчейн-технологій. Стандарти взаємодії дозволяють підвищити сумісність та узгодженість різних блокчейн-протоколів та забезпечують їхню ефективну спільну роботу. Регулювання важливе для забезпечення відповідності до законодавства та забезпечення безпеки цифрових активів.

Проведене дослідження виявило потребу в подальших наукових розробках. Розвиток нових блокчейн-протоколів, покращення існуючих технологій та визначення кращих практик їхнього використання є важливими напрямками подальших досліджень. Активна участь глобальної наукової спільноти сприятиме еволюції блокчейн-технологій [183]. Глобальне співтовариство відіграє ключову роль у розвитку та впровадженні блокчейн-технологій. Співпраця між державами, підприємствами та науковими установами дозволяє створювати стійкі, безпечні та ефективні блокчейн-рішення [184].

Підсумовуючи, треба зазначити, що цифрові активи, такі як крипто валюти та токени, стають все більш важливим елементом сучасної фінансової системи. У світі, де цифрова трансформація набуває обертів, захист цих активів стає пріоритетом та має такі найближчі перспективи. Це надійність механізмів консенсусу, оскільки Блокчейн використовує різноманітні механізми такі як Proof-of-Work (PoW), Proof-of-Stake (PoS), та інші. Вони

забезпечують високий рівень відомостей та захищають від подвійних витрат. Смарт-контракти, що використовують блокчейн, дозволяють автоматизувати виконання угод та управління активами, що зменшує ризики та підвищує ефективність. Децентралізована природа блокчейну робить його менш вразливим до атак та забезпечує високий рівень захисту від цензури та маніпуляцій [168]. Треба зазначити можливість використання блокчейну у самих різних галузях. Зокрема це і фінансовий сектор – впровадження блокчейну в банківські операції може підвищити ефективність та захист від шахрайства. Ланцюг постачання – застосування блокчейну у ланцюгах постачання може покращити відстеження товарів та зменшити можливість підробки даних. Медицина та зберігання даних – використання блокчейну для зберігання медичних даних може гарантувати конфіденційність та недоступність даних для несанкціонованого доступу.

Блокчейн визнаний ефективним інструментом для захисту цифрових активів. Найближчі перспективи полягають у подальшому вдосконаленні механізмів консенсусу, розвитку масштабованості та реагуванні на виклики щодо регулювання. Застосування блокчейн-технологій для захисту цифрових активів виступає як об'єктивно обґрунтоване та практично ефективне рішення. Успішна імплементація та розвиток цих технологій обумовлені правильним вибором протоколу, створенням стандартів, регулюванням та активною участю глобального співтовариства.

Цифрові активи стають суттєвою частиною сучасного фінансового ландшафту, вимагаючи новітніх підходів до їхнього захисту. Блокчейн технології виявляються ключовим інструментом у цьому контексті, пропонуючи інноваційні рішення для забезпечення безпеки та надійності цифрових активів. Аналіз існуючих рішень та визначення перспектив показують, що виникаючі виклики можуть стати стимулом для подальшого розвитку та впровадження інноваційних технологій, що покращать захист цифрових активів і розширять можливості їхнього використання. При цьому підкреслюється важливість стандартизації та регулювання в галузі блокчейн-

технологій для забезпечення їхньої ефективної інтеграції та відповідності до вимог законодавства.

4.4 Задача аналізу медичних зображень на прикладі розпізнавання клітин крові

У цьому підрозділі розглянуто структуру системи аналізу медичних зображень і у якості прикладу наведено алгоритм роботи системи розпізнавання клітин крові. Сформульовано основні завдання, які необхідно вирішити при проведенні морфологічного аналізу крові, висунуті вимоги до алгоритму визначення лейкоцитарної формули і виявлення тілець крові на мазку. Для опису типових зображень мазка крові запропонована модель колірно-яскравих характеристик. Порогові значення розмірів об'єктів визначаються при пошуку осередків. Досліджено гістограму яскравості типового поля зору. Описаний двоетапний алгоритм виявлення клітин крові, а також алгоритм побудови розділової лінії на площині відносних кольорів. Наведено результати дослідів на реальних препаратах, а також розглянуто причини виникнення помилок детектування.

Опишемо саму прикладну задачу аналізу медичних зображень. У задачах аналізу зображень, отриманих за допомогою мікроскопа, в рамках цитологічних досліджень часто доводиться підраховувати кількість клітин певного типу. При дослідженні препаратів крові важливим завданням є підрахунок кількості еритроцитів, за показниками якого можна діагностувати порушення в кровотворенні або пошкодження еритроцитів внаслідок різних факторів.

Серед клітин крові розрізняють еритроцити, лейкоцити, тромбоцити. Еритроцит являє собою без'ядерну клітину рожевого кольору, що має вигляд дещо сплюсненого еліпсоїда з поглибленням в центрі середнього розміру 8 мкм. Лейкоцити відрізняються від еритроцитів більшим розміром, що становить 9-20 мкм, наявністю ядра і характером їх забарвлення, який може

бути фіолетовим, рожевим або яскраво-червоним. Тромбоцити – це без'ядерні утворення округлої або овальної форми розміром 1-3 мкм, з червоно-фіолетовим центром і рожево-блакитною периферією.

Існують різні методи підрахунку еритроцитів в крові, деякі з них використовують вже змодельовану базу зображень клітин крові і їх характеристик [185-188], використовують порогове розкладання [189] або сегментацію за допомогою методу контрольованого вододілу [190]. Існують підходи, в яких для сегментації використовуються колірні характеристики зображення [191, 192] або характеристики текстури [192]. У [193] було запропоновано використовувати алгоритм активної контурної моделі для вибору контурів клітин. Основна проблема підрахунку клітинок полягає в тому, що вони можуть накладатися одна на одну, а також змінювати свою форму в певному діапазоні. Наявність сторонніх шумів, сторонніх предметів в полі зору мікроскопа ще більше ускладнює аналіз зображення. У даній роботі, з метою зниження ефекту шуму, пропонується використовувати медіанну фільтрацію зображень [194] з подальшим вилученням контурів клітин детектором меж Канні [195]. Для поліпшення розпізнавання кордонів зображення додатково контрастують.

При виявленні лейкоцитів використовують два методи сегментації «первинних об'єктів». На початку скринінгу інформація про кольори лейкоцитів та еритроцитів у цьому препараті вважається невідомою. Тому використовується одновимірний ітераційний метод, заснований на вивченні піків гистограми яскравості. Після того як він знайшов кілька ядер лейкоцитів і накопичив інформацію про кольори еритроцитів, на площині відносних кольорів f_R , f_B будують пряму лінію, що розділяє кольори ядер і колір еритроцитів. Якщо межа між кольорами еритроцитів і ядер проведена «з запасом» (похибки першого і другого роду невеликі), її починають використовувати для сегментації. Так, при виділенні первинних об'єктів (можливо, ядер) відбувається перехід від одновимірного методу до двовимірного: замість сегментації гистограми яскравості виконується

сегментація колірної площини. Крайній варіант більш стабільний: немає небезпеки пропустити світлові ядра, простіше працювати з погано сфокусованими кадрами, не потрібно точно визначати положення піку фону [188].

Алгоритм вилучення фрагментів складається з чотирьох кроків.

1. Виділення первинних об'єктів – можливо, ядер. При цьому використовуються два різних методи сегментації – за гистограмою яскравості або по колірній площині. Другий спосіб кращий, але на початковому етапі накопичення проб використовується перший.

2. Перевірка первинних об'єктів на відповідність вже накопиченому зразку ядер, що можливо при кількості накопичених об'єктів більше 5. Первинні об'єкти класифікуються на ядерні фрагменти та артефакти.

3. Фрагменти ядер поєднуються один з одним, і робиться спроба побудувати навколо них цитоплазму. В результаті отримуємо прямокутник, всередині якого знаходиться один лейкоцит.

4. Якщо для сегментації первинних об'єктів використовувалася гистограма яскравості, і на ній було кілька піків, можливо, відповідних ядрам, то число знайдених пікселів ядер порівнюється з розрахунковим (число пікселів в піку). В результаті перевірки гіпотези про те, який пік слід вважати піком ядер, вона може змінитися і алгоритм можна починати з першого кроку заново.

Як правило, необхідності в ітераціях немає, алгоритм складається з трьох кроків: первинні об'єкти → ядерні фрагменти → лейкоцити.

Алгоритм сегментації на основі гистограми яскравості.

Запропонований алгоритм виявлення лейкоцитів складається з двох етапів, які можуть повторюватися кілька разів для одного і того ж кадру. На першому етапі (на основі вивчення гистограми яскравості кадру та історії пошуку) вибираються порогові значення яскравості G і частки синього fB . На другому етапі досліджують набори пікселів, які відповідають цим умовам (первинні об'єкти), щоб визначити, чи можна їх розглядати як ядра лейкоцитів. Якщо

сумарна кількість пікселів в цих фрагментах значно менше раніше передбачуваного числа, то підбір порогових значень вважається незадовільним і алгоритм запускається знову і т.д.

Перший етап – вивчення гістограми, вибір порогових значень. Першим кроком є локалізація піку, пов'язаного з фоном зображення, який в подальшому буде еталонним, як при розрахунку оптичної щільності, так і при визначенні відносних кольорів для решти пікселів. Абсолютні значення використовуються тільки у випадку, якщо фоновий пік не локалізований. В якості піку пікселів фону вибирається крайній правий пік з досить малою дисперсією: стандартне відхилення становить менше 10 цифр [189].

Далі складається список максимумів (піків), яким могли б відповідати ядра лейкоцитів. Для цього їх оптична щільність повинна бути досить великою (емпірично знайдена межа $> 0,6$) і середня синя фракція $fB = B/(B + G + R)$ для пікселів на цьому максимумі повинна перевищувати аналогічне значення для фону на 0,03 (емпірично знайдена межа). Якщо таких підозрілих максимумів кілька, їх вибирають послідовно один за одним (в цьому випадку викликається другий етап алгоритму), починаючи з найяскравішого. Проведена порогова сегментація по яскравості і відносній частці синього $G < G_{max}; fB = B/(B + G + R) < fB_{max}$.

Отримані первинні об'єкти порівнюють з уже існуючими ядрами. Якщо їх не кваліфікувати як ядра, а це можливо при наявності оптично щільних і синюватих червоних кров'яних тілець, плям фарби, великих тромбоцитів, то буде обраний наступний максимум. Якщо лейкоцитарний пік не виділяється зовсім, то використовуються порогові значення, які ґрунтуються вже не на поточній гістограмі, а на передісторії пошуку, а якщо такої немає, то на апріорних значеннях.

Другий етап – вивчення отриманих фрагментів. Ця частина не залежить від методу, за допомогою якого були отримані первинні об'єкти. Алгоритм перевірки вибраних об'єктів складається з трьох циклів. У першому циклі

відкидаються занадто великі ($A > 2000 \text{ мкм}^2$) і занадто маленькі ($A < 11 \text{ мкм}^2$) предмети.

Далі вимірюється оптична щільність і колірні характеристики. Якщо є передісторія пошуку, то за критерієм «три сигми» відкидаються надмірно світлі об'єкти з низькою оптичною щільністю і викликається процедура перевірки кольорів. Решту об'єктів поміщають в клас умовних ядер лейкоцитів.

У другому циклі завершення формування цитоплазми навколо ядра відбувається на решті об'єктів. До цитоплазми відносять когерентний набір сусідніх пікселів, які з великою ймовірністю (більше 0,95) не є пікселями еритроцитів або фоном. Побудований набір відбраковується, якщо він занадто великий (більше 2000 мкм^2) або форм-фактор його зовнішньої межі (квадрат периметра / площі) перевищує досить велике значення (більше 50). Доволі часто при уважному старанні білих кров'яних тілець схоже забарвлених еритроцитів цитоплазма не може бути завершена таким простим чином. Потім ядро або його фрагмент поміщають всередину прямокутника з доданими рамками по 15 мкм.

Після завершення заповнення цитоплазми проводиться аглютинація ядерних фрагментів. Це необхідно, так як ядра нейтрофілів виявляються у вигляді декількох фрагментів. При цьому відбувається поділ клітин, які знаходяться близько один до одного. Це можливо, якщо клітини лежать острівцями, які не пов'язані один з одним, оточені фоном. Після другого циклу отримані об'єкти розглядаються як окремі лейкоцити. У третьому циклі знову перевіряють розміри цих об'єктів, а занадто великі предмети відбраковують.

Якщо для сегментації використовувалася гістограма яскравості, то в кінці проводиться перевірка на відповідність припущень і отриманих результатів. Для цього знайдене число пікселів ядер лейкоцитів порівнюють з числом пікселів на піку гістограми, яке було прийнято за відповідні ядра. Якщо відмінності значні (перевищують $> 50\%$) і зліва від піку передбачуваних лейкоцитів був ще один, то вищеописаний алгоритм спрацьовує знову.

Алгоритм перевірки первинного об'єкта на приналежність до групи лейкоцитарних ядер. Ядра лейкоцитів не складають однорідної групи. Тому не варто строго ставитися до перевірки на приналежність нового об'єкта до двомірного, нормального розподілу навіть для середніх значень відносних кольорів. Крім того, в процесі скринінгу мазка бажано використовувати алгоритм перевірки якомога раніше, коли кількість накопичених об'єктів невелика. Тому запропонований нижче алгоритм є евристичним. Він ґрунтується на таких положеннях.

1. Кожен новий об'єкт порівнюють з двома групами: групою ядер і групою еритроцитів.

2. При визначенні ймовірності приналежності нової точки до існуючої групи ймовірність обчислюється двічі. У першому випадку ймовірність $P1$ обчислюється до того, як точка буде додана до наявної статистики, а в другому випадку – $P2$, після такого додавання. Очевидно, $P2 > P1$. Такий розрахунок відразу двох ймовірностей необхідний, якщо рішення приймаються на основі невеликих вибірок, а кількість накопичених без перевірки об'єктів зазвичай просто невелика – близько 5. Якщо кількість об'єктів у групі перевищує кілька десятків, різниця між $P1$ і $P2$ майже зникає.

3. Якщо досліджуваний об'єкт більш «синьо-червоний», ніж вже накопичені ядра, то він приймається в будь-якому випадку. Іншими словами, не враховується можливість викидів у цю «синьо-червону» сторону. І навпаки, якщо об'єкт більш «зелений», ніж еритроцити, то він в будь-якому випадку відторгається.

4. Якщо ймовірність $P2$, обчислена щодо групи еритроцитів, більша за відповідну ймовірність, обчислену щодо ядер, то об'єкт відкидається.

5. Якщо рішення не прийнято на підставі попередніх пунктів, то воно приймається з урахуванням трьох ймовірностей. Ймовірності $P1$ і $P2$ оцінюють відхилення середнього значення, а ймовірність $P3$ – відхилення площі еліпса розсіювання від характерного для групи ядер. Об'єкт відхиляється, якщо $P2 < 0,01$ або $P3 < 0,01$.

Для даного первинного об'єкта відповідні ймовірності $P1 = 0,53$, $P2 = 0,65$, $P3 = 0,45$ великі, тому його правильно кваліфікувати як ядро і додати в статистику ядер.

Метод виділення і підрахунку еритроцитів. Одновимірний медіанний фільтр – це метод обробки сигналів, який використовує «ковзне вікно» заданої довжини. У цьому вікні значення сортуються за зростанням, а центральний елемент замінюється медіаною – середнім значенням відсортованого ряду. Такий підхід ефективно пригнічує імпульсні завади, зберігаючи різкі переходи сигналу. Таким чином, операція медіанної фільтрації K -вимірної послідовності значень сигналу $s(k)=s(x_k)$, $K=1, \dots, K$ характеризується співвідношенням

$$\text{med}_{1 \leq k \leq N} \{s_k\} = \begin{cases} 0.5(s_n + s_{n+1}), N = 2n, \\ s_n, N = 2n - 1, \end{cases} \quad (4.5)$$

де фіксована величина $n = 1, 2, \dots$ визначає діафрагму світлофільтра.

Наступний етап полягає у виборі меж, після чого набір компонентів з критерієм зв'язності уздовж восьми сусідів виокремлюється на окремі контури. Для кожної зв'язаної області обчислюється площа опуклого багатокутника, що описує контур. Для заданих порогів відрізаються занадто великі або занадто малі ділянки. Порогові значення вибираються виходячи з передбачуваної реальної площі клітини.

Потім з кожного окремого контуру вибираються точки через рівні проміжки по довжині контуру. Точки з'єднують попарно один з одним, а через середину отриманого відрізка проводять перпендикуляр. Його можна описати рівнянням

$$y = -\frac{x_2 - x_1}{y_2 - y_1} x + \frac{y_2^2 - y_1^2 + x_2^2 - x_1^2}{2(y_2 - y_1)}, \quad (4.6)$$

де (x_1, y_1) – координати першої точки, (x_2, y_2) – координати другої точки.

Розташування точки перетину сусідніх перпендикулярів зберігається. Точка перетину двох перпендикулярів обчислюється як

$$x = \frac{b_2 - b_1}{a_1 - a_2}, \quad (4.7)$$

$$y = a_1 x + b_1 = a_2 x + b_2, \quad (4.8)$$

де $a = \frac{x_2 - x_1}{y_2 - y_1}$ – коефіцієнт нахилу перпендикуляра, $b = \frac{y_2^2 - y_1^2 + x_2^2 - x_1^2}{2(y_2 - y_1)}$ – коефіцієнт перпендикулярного зміщення.

Операція виконується для всіх пар перпендикулярів для різних кроків взяття точок. В результаті утворюється хмара точок, які розташовані щільніше на ділянках, що є центрами радіуса кривих контуру.

Клітини, що склеєні одна з одною або накладені одна на іншу, досить складно піддаються сегментації методами, заснованими на аналізі розміру або форми ділянок всередині контурів. Прогалини в контурах ускладнюють контурну сегментацію клітин. Запропонований метод являє собою єдине зображення для всіх ділянок контурів їх центрів, що дозволяє вирішувати зазначені вище завдання.

Картина, складена з отриманих точок перетину перпендикулярів, піддається морфологічній обробці, в результаті якої залишаються тільки щільні і досить великі скупчення точок. Ці скупчення відповідають передбачуваним клітинним центрам. За допомогою методу з'єднаних компонентів підраховуються кластери, які повинні відповідати кількості осередків на зображенні.

Результати експерименту. При експериментальному дослідженні запропонованого методу було зроблено зображення зразка крові за допомогою мікроскопа (рис. 4.6).

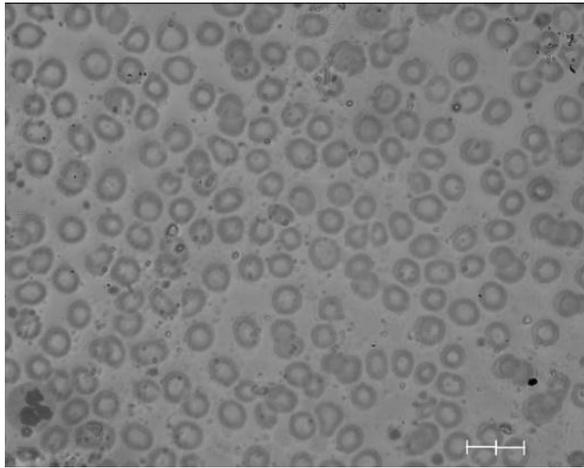


Рисунок 4.6 – Тестова картина

В першу чергу, зображення було перетворено з кольорового в чорно-біле. Розмір вікна в медіанній фільтрації був обраний на основі середнього розміру комірки і становив 16x16 пікселів, що відповідає 20% від діаметра комірки. Результат медіанної фільтрації тестового зображення показаний на рис.4.7.

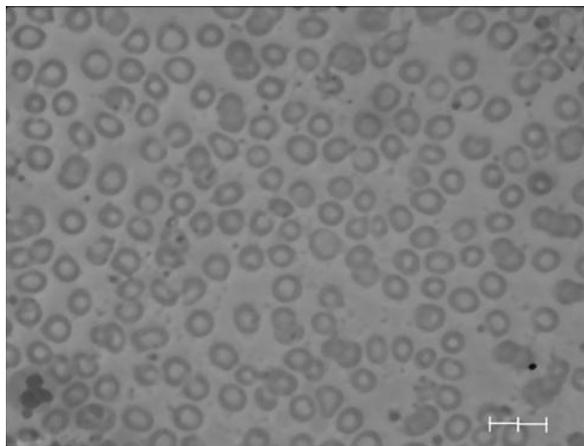


Рисунок 4.7 – Результат медіанної фільтрації

Після контрастування лінійного зображення був застосований детектор меж Канні (Canny) (рис. 4.8). Алгоритм Канні спочатку згладжує зображення, щоб видалити шум, потім вибираються межі, де градієнт зображення набуває максимального значення, при цьому як межі позначені лише локальні максимуми. Наступним кроком є визначення потенційних меж фільтрації з подвійним порогом.

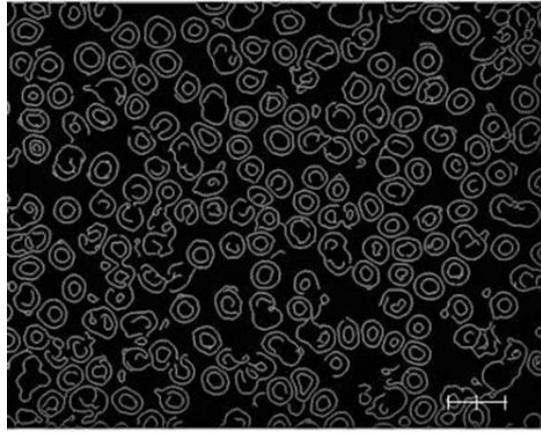


Рисунок 4.8 – Зображення країв з видаленими великими і маленькими контурами

Загальні межі визначаються шляхом придушення всіх ребер, які не пов'язані з конкретними межами. В якості порогів для видалення занадто великих або занадто дрібних контурів на зображенні було обрано два значення: $0,05S$ як нижній поріг і $4S$ як верхній поріг, де S – приблизна площа зображення клітинки, розрахована на основі її діаметра.

В результаті побудови перпендикулярів (рис. 4.9) для відрізків між точками, взятих з інтервалом 3–40 пікселів, для кожного контуру отримано хмару точок перетину (рис. 4.10).

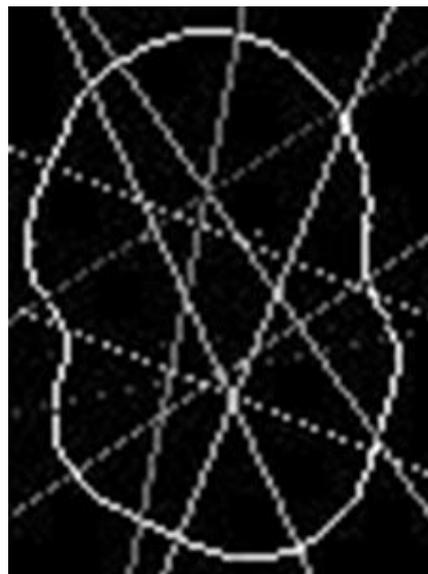


Рисунок 4.9 – Перпендикуляри; відстань між точками = 20

Кількість підключених компонентів становила 220. Реальне число клітинок на зображенні вийшло рівним 209. Кількість хибно позитивних результатів – 23, кількість нерозпізнаних клітин – 12. Метод був протестований на чотирьох різних зображеннях, що містять клітини крові. Середня ймовірність правильного підрахунку кількості еритроцитних клітин склала 86%. У порівнянні з методами з використанням порогового розкладання [189] або сегментації за методом контрольованого вододілу [190], запропонований метод дав найкращі результати. Однак у порівнянні з методами, запропонованими в [185–188, 193], ймовірність правильного підрахунку кількості осередків виявилася меншою, оскільки прикордонний детектор неправильно вибирав ребра на тестових зображеннях через сильний шум та наявність нечітких меж [197, 198]. Для таких зображень потрібні додаткові методи попередньої обробки, щоб підвищити ефективність вилучення країв.

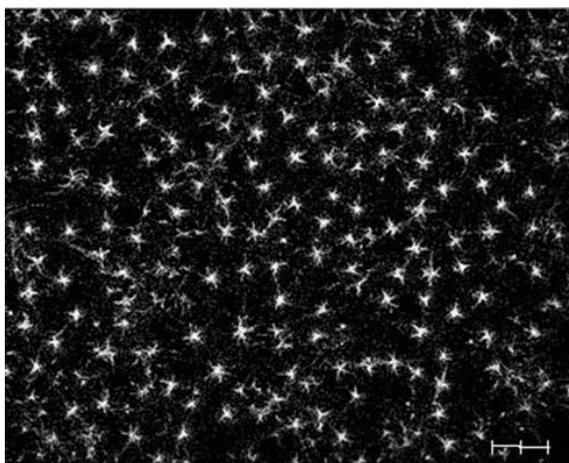


Рисунок 4.10 – Хмари точок

Із зображення хмари точок видалялися ділянки з площею менше 5 пікселів, а потім виконувалася операція закриття двійкового зображення з маскою 8x8 пікселів. Результат морфологічної обробки зображення точок перетину, накладених на вихідне зображення, показаний на рис.4.11.

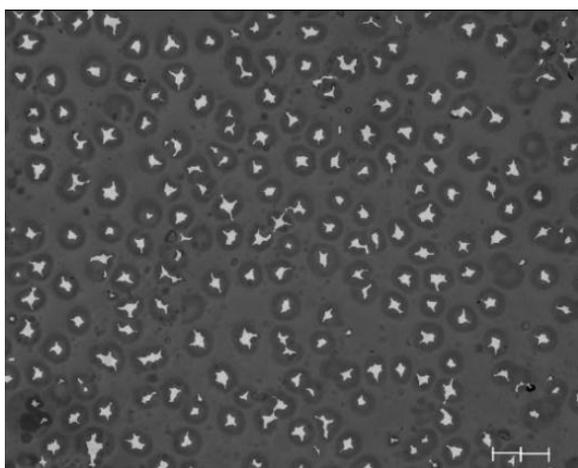


Рисунок 4.11 – Морфологічно оброблене зображення точок перетину, накладене на вихідне зображення

Можна дійти певних висновків. Завдяки тому, що для ідентифікації клітин використовується прикордонний детектор, результати сегментації не залежать від кольору клітин, їх текстури і внутрішньої структури. Метод з досить високою точністю сегментує клітини, склеєні між собою або накладені один на одного. В умовах шумного вихідного джерела метод зображень показав хороші результати.

Крім того, запропонований алгоритм дозволив проводити сегментацію і підрахунок клітин крові з точністю до 86%. Число помилкових виявлених клітин в середньому вище, ніж при інших методах, що можна пояснити наявністю великої кількості шуму на тестових зображеннях, а також нечіткими межами клітин. Більш високої ймовірності правильного підрахунку кількості осередків можна досягти, якщо паралельно із запропонованим способом використовувати інші, уточнюючи результат сегментації одного методу за результатами іншого, а також використовуючи інші алгоритми попередньої обробки вихідного зображення. Для цього у майбутньому планується розробити алгоритм попередньої обробки зображень для підвищення ефективності запропонованого методу, а також поєднання запропонованого методу з іншими.

В цілому можна сказати, що результати дослідження підтверджують доцільність використання гібридних сховищ зображень з інтеграцією методів машинного навчання для оптимізації процесу пошуку. Використання MongoDB та Elasticsearch у поєднанні з попереднім розрахунком векторів ознак дозволяє досягти високої ефективності та точності при роботі з великими обсягами візуальних даних. Правильний вибір методів машинного навчання та технологій зберігання даних значно впливає на результати роботи системи. Використання моделі ResNet50 для екстракції ознак зображень та алгоритму косинусної подібності для порівняння векторів ознак виявилось ефективним рішенням для задачі пошуку схожих зображень.

Загалом, проведене дослідження показало, що інтеграція машинного навчання в процеси управління та обробки даних у гібридних сховищах зображень відкриває нові можливості для покращення ефективності та точності роботи з великими обсягами візуальної інформації. Це дозволяє забезпечити швидкий та надійний доступ до необхідних даних у різних сферах застосування, включаючи медицину, безпеку та електронну комерцію.

Завдяки ретельному тестуванню, гібридна модель незмінно перевершувала DB-Standard, досягаючи швидшого отримання даних, кращої оптимізації сховища та мінімізувавши обчислювальні витрати. Цей складний порівняльний аналіз підкреслює надійність оптимізованого методу індексації, підтверджуючи його корисність для різноманітних застосувань.

Блокчейн визнаний ефективним інструментом для захисту цифрових активів. Найближчі перспективи полягають у подальшому вдосконаленні механізмів консенсусу, розвитку масштабованості та реагуванні на виклики щодо регулювання. Застосування блокчейн-технологій для захисту цифрових активів виступає як об'єктивно обґрунтоване та практично ефективне рішення. Успішна імплементація та розвиток цих технологій обумовлені правильним вибором протоколу, створенням стандартів, регулюванням та активною участю глобального співтовариства. Цифрові активи стають суттєвою частиною сучасного фінансового ландшафту, вимагаючи новітніх підходів до

їхнього захисту. Блокчейн технології виявляються ключовим інструментом у цьому контексті, пропонуючи інноваційні рішення для забезпечення безпеки та надійності цифрових активів.

Аналіз існуючих рішень та визначення перспектив показують, що виникаючі виклики можуть стати стимулом для подальшого розвитку та впровадження інноваційних технологій, що покращать захист цифрових активів і розширять можливості їхнього використання. Доведена важливість стандартизації та регулювання в галузі блокчейн-технологій для забезпечення їхньої ефективної інтеграції та відповідності до вимог законодавства.

Завдяки тому, що для ідентифікації клітин використовується прикордонний детектор, результати сегментації не залежать від кольору клітин, їх текстури і внутрішньої структури. Метод з досить високою точністю сегментує клітини, склеєні між собою або накладені один на одного. В умовах шумного вихідного джерела метод зображень показав хороші результати. Запропонований алгоритм дозволив проводити сегментацію і підрахунок клітин крові з точністю до 86%. Число помилково виявлених клітин в середньому вище, ніж при інших методах, що можна пояснити наявністю великої кількості шуму на тестових зображеннях, а також нечіткими межами клітин. Більш високої ймовірності правильного підрахунку кількості осередків можна досягти, якщо паралельно із запропонованим способом використовувати інші, уточнюючи результат сегментації одного методу за результатами іншого, а також використовуючи інші алгоритми попередньої обробки вихідного зображення.

Виконано дослідження оптимізованого методу індексації, його гібридної моделі та порівняння з галузевими стандартами. Цей метод не тільки демонструє різке покращення часу пошуку, але й демонструє ефективність зберігання та зменшення обчислювальних витрат. Ці результати знайшли своє відображення в емпіричному аналізі, в якому реальні застосування, такі як супутникові знімки, підтверджували математичні докази його переваги.

ЗАГАЛЬНІ ВИСНОВКИ

1. Проведено комплексний аналіз сучасних підходів до архітектури сховищ зображень в умовах великих даних, що дозволило виявити ключові проблеми та перспективні напрямки досліджень у галузі ефективного зберігання та обробки великомасштабних наборів візуальних даних. Розроблено концептуальну модель гібридного сховища зображень, яка поєднує переваги реляційних та нереляційних баз даних, файлових систем та блокчейн-технологій. Запропонована модель забезпечує оптимальний баланс між масштабованістю, продуктивністю та безпекою зберігання даних, що підтверджено експериментальними дослідженнями.

2. Визначено та проаналізовано основні методи та технології стиснення зображень для їх ефективного зберігання у гібридних та блокчейн-орієнтованих сховищах. Досліджено переваги IPFS та інших децентралізованих платформ для розподіленого зберігання візуальних даних. Запропоновано методологію підвищення ефективності управління сховищами зображень, яка включає алгоритми машинного навчання для вибору оптимального місця збереження, індексації та швидкого пошуку великих наборів візуальних даних.

3. Розроблено алгоритм оптимізованої індексації та пошуку зображень, що базується на використанні моделей глибокого навчання (ResNet50) та векторних подібностей (FAISS, OpenAI CLIP). Запропоноване рішення дозволило зменшити накладні витрати з 0,08 с. у традиційних рішеннях до 0,019 с., а також підвищити швидкість пошуку за векторними представленнями до 38% в порівнянні зі стандартними методами індексації. Це дозволило значно пришвидшити процес ідентифікації об'єктів. Розроблено та експериментально перевірено алгоритми стиснення та дедуплікації зображень для ефективного використання в блокчейн-сховищах, що дозволило на 47% зменшити обсяг даних при збереженні якості зображень. SSIM-метрика підтвердила високу структурну подібність зображень із

середнім значенням 0,92, що свідчить про мінімальні втрати інформації при стисканні. Ефективність використання пам'яті у гібридному сховищі склала 91% проти 70% в класичних централізованих системах.

4. Досліджено безпекові аспекти гібридного сховища зображень, включаючи методи захисту від несанкціонованого доступу, криптографічні механізми шифрування та стенографічні підходи для підвищення конфіденційності цифрових активів. Створено та протестовано систему розподіленої обробки зображень на базі Apache Spark, яка забезпечує ефективну паралельну обробку великих обсягів візуальних даних у кластерних середовищах.

5. Запропоновано інтеграцію блокчейну з алгоритмами машинного навчання для підвищення рівня автоматизації та безпеки зображень, що дозволяє створити інтелектуальні системи класифікації та захисту інформації. Аналіз застосування блокчейн-технологій для зберігання та захисту зображень підтвердив їхню ефективність у забезпеченні прозорості, незмінності та стійкості до несанкціонованих змін. Досліджено переваги та обмеження використання смарт-контрактів для обробки візуальних даних.

6. Проведено експериментальне дослідження ефективності розробленої гібридної моделі на різних наборах даних (медичні зображення, супутникові знімки, цифрове мистецтво), яке підтвердило її переваги над існуючими підходами. Швидкість доступу до збережених файлів у гібридному сховищі зросла в 4,3 рази – середній час доступу 1,05 с. проти 4,5 с. у стандартних системах. Час доступу до даних у гібридному сховищі зменшився на 27% у порівнянні з традиційними централізованими сховищами, що підтверджує ефективність використання децентралізованої моделі. Розроблено алгоритм автоматизованого аналізу медичних зображень, що базується на методах сегментації та ідентифікації клітин крові, який демонстрував практичну цінність запропонованих методів для конкретних прикладних задач.

7. Розроблено прототип системи на базі блокчейну для зберігання та захисту медичних зображень, який продемонстрував практичну застосовність

запропонованих методів у критично важливих галузях, таких як охорона здоров'я, криміналістика та цифрова архівація.

8. Запропоновано комплексну інформаційну технологію для ефективного зберігання, обробки та аналізу великих обсягів зображень, яка інтегрує всі розроблені методи та алгоритми, поєднуючи AI, блокчейн та гібридне сховище. Сформульовано рекомендації для практичного впровадження гібридного підходу до управління сховищами зображень, що можуть бути застосовані в медичних, фінансових, безпекових та промислових системах для аналізу великих обсягів візуальних даних.

СПИСОК ЛІТЕРАТУРИ

1. Рогушина Ю. В. Засоби та методи аналізу неструктурованих даних / Ю.В. Рогушина // Проблеми програмування. 2019. № 1. – с. 57 – 77.
2. Tereshchenko G. Overview and Analysis of Existing Decisions of Determining the Meaning of Text Documents / G. Tereshchenko, I. Gruzdo // Problems of Infocommunications Science and Technology (PIC S&T`2018). – Kharkiv, Ukraine, October 9-12, 2018. – Institute of Electrical and Electronics Engineers Inc. paper № 8632014. – pp. 645-653. ISBN 978-153866611-1. doi: 10.1109/INFOCOMMST.2018.8632014.
3. Стрижак О.Є., Горборуков В. В., Франчук О. В. та Попова М. А. Онтологія задачі вибору та її застосування при аналізі лімнологічних систем. Екологічна безпека та природокористування: Зб. наук. праць М-во освіти і науки України, Київ. паз. ун-т буд-ва і архіт., НАН України, Ін-т телекомунікацій і глобал. інформ. простору; редкол.: О. С. Волошкіна, О.М. Трофимчук (голов. ред.) [та ін.]. Київ, Україна: №15, с.172-183, 2014.
4. Blei D., Jordan M. and Ng A., Latent Dirichlet Allocation, Journal of Machine Learning Research, vol. 3, no. 1, pp. 993-1022, 2003.
5. Shovkovyi Y., Grinyova O., Udovenko S., Chala L.. Automatic sign language translation system using neural network technologies and 3D animation. Innovative Technologies and Scientific Solutions for Industries. 2023. № 4 (26). С. 108–121. DOI: 10.30837/ITSSI.2023.26.108
6. Tereshchenko G. Application of Paragraphs Vectors Model for Semantic Text Analysis / I. Gruzdo, I. Kyrychenko, G.Tereshchenko, O. Cherednichenko // 4th International Conference on Computational Linguistics and Intelligent Systems (COLINS-2020), Lviv, Ukraine, April 23-24, 2020. – CEUR-WS 2604. – pp. 283-293.
7. Sitikhu, P., Pahi, K., Thapa, P., Shakya, S.: A Comparison of Semantic Similarity Methods for Maximum Human Interpretability. arXiv preprint arXiv:1910.09129 (2019).

8. Panigrahi, A., Simhadri, H. V., Bhattacharyya, C.: Word2Sense: Sparse Interpretable Word Embeddings. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pp. 5692-5705 (2019).

9. Kanishcheva, O., Cherednichenko, O., Sharonova, N.: Image Tag Core Generation. In: 1st International Workshop on Digital Content & Smart Multimedia (DCSMart 2019) Ukraine, CEUR Workshop Proceedings, Volume 1, pp. 35-44. Lviv (2019). <http://ceur-ws.org/Vol-2533/preface.pdf>

10. Філатов В., Белянінова Г., Єрохін А. Інформаційна система рейтингу публікаційної активності науковців Харківського національного університету радіоелектроніки. Наука і техніка сьогодні. № 2(16). С. 438–447. DOI: 10.52058/2786-6025-2023-2(16)-438-447

11. Sharonova N., Kyrychenko I., Gruzdo I., Tereshchenko G.. Generalized Semantic Analysis Algorithm of Natural Language Texts for Various Functional Style Types, CEUR Workshop Proceedings, 2022. – 3171, Volume I: Main, PP. 16 - 26.– ISSN: 16130073.

12. Vysotska, V., Lytvyn, V., Kovalchuk, V., Kubinska, S., Dilai, M., Rusyn, B., Pohreliuk, L., Chyrun, L., Chyrun, S., Brodyak, O.: Method of Similar Textual Content Selection Based on Thematic Information Retrieval. In: Proceedings of the International Conference on Computer Sciences and Information Technologies, CSIT, 1-6. (2019)

13. Filatov V. O., Yerokhin A. L., Zolotukhin O. V., Kudryavtseva M. S. Hybrid Simulation Models for Complex Decision-Making Problems with Partial Uncertainty. Відбір і обробка інформації. 2022. Вип. 50 (126). С. 78–86. DOI: 10.15407/vidbir2022.50.078

14. Smooth, A.V., Ideas, M.M.: Bakhtin on utterance and dialogue and their significance for the formal semantics of natural language. In the book: Interactive systems: Reports and theses of reports and messages of the third school-seminar.- Tbilisi: Metsniereba, vol. I, p. 33-43 (1981).

15. Golovina, E.A., Kolmychek, K.N., Terzian, V.N.: The principles of verifying the semantic correctness of natural language utterances. In the book. : Problems of Bionics.- Kharkov: KSU, no. 32. pp. 64-72 (1984).

16. Golovina, E.A., Terzian, V.Ya. Express analysis of natural language utterances. In the book: Interactive systems: Materials of the fifth school-seminar.- Tbilisi: Metsniereba, pp. 385-388 (1983).

17. Cherednichenko O., Sytnikov D., Romankiv N., Sharonova N., Sytnikova P.. Selection of Large Language Model for development of Interactive Chat Bot for SaaS Solutions. 8th International Conference on Computational Linguistics and Intelligent Systems (CoLInS 2024), Apr 2024, Lviv, Ukraine. pp.66-87 (Vol. IV).

18. Terzian, V. Ya. Theoretical and experimental study of the problem of semantic analysis of natural language utterances: dis. cand. tech. Sciences: 05.13.01 "Technical cybernetics and information theory" Terziyan Vagan Yakovlevich; Kharkiv. Institute of Radio Electronics. - Kharkov (1984).

19. Harris, L.R. Using a Data Base as a Semantic Component to Aid in the Parsing of Natural Language Data Base Queries. In: Journal of Cybernetics, v. 10, No. 1-3, pp. 77-96 (1980)

20. Жеребкін В. І., Удовенко С. Г., Чала Л. Е., Гриньова О. Є. Аналіз складності та побудова концептуальних графів масових відкритих онлайн-курсів. Біоніка інтелекту. 2023. № 1 (99). С. 26–37. DOI: 10.30837/bi.2023.1(99).04

21. Оксанич А. Г., Шевченко І. В., Калюжний О. В., Мизин В. А. Взаємодія агентів в інформаційній технології динамічного формування виконавчих структур. Вісник КрНУ імені Михайла Остроградського. 2020. Вип. 1/2020 (120). С. 99-106. DOI: 10.30929/1995-0519.2020.1.99-106.

22. Коптыра К., Ogiela M.R. Imagechain – Application of Blockchain Technology for Images // Sensors 2021, 21, 82. doi: 10.3390/s21010082.

23. Filatov V., Semenets V., Zolotukhin O. Data Mining in Relational Systems. Сучасний стан наукових досліджень та технологій в промисловості. 2020. № 3 (13). С. 65–76. DOI: 10.30837/ITSSI.2020.13.065.
24. Wang, Q., Zhu, X., Ni, Y., Gu, L., Zhu, H., Blockchain for the IoT and industrial IoT // A review. Internet of Things, 2020, 10, 100081.
25. Ashwin B., Raghuram T., S. M. Reza S., et al. (2020). Big Data Analytics in Healthcare. Journal of Biomedical Informatics, 79, doi: 10.1155/2015/370194.
26. Debbarma N. Analysis of Data Quality and Performance Issues in Data Warehousing and Business Intelligence / N. Debbarma, G. Nath, H. Das // International Journal of Computer Applications – 2013 – Vol. 79, No 15 – P.20- 26.
27. Kimpel J. F. Critical Success Factors for Data Warehousing: a classic answer to a modern question / J. F. Kimpel // Issues in Information Systems – 2013 Volume 14, Issue 1 —P.376-384.
28. Jarke M. Fundamentals of data warehouses / M. Jarke, M. Lenzerini, Y. Vassiliou, P. Vassiliadis // Springer-Verlag Berlin, Heidelberg New York, 2013 – 224 p.
29. Blanco C. An architecture for automatically developing secure OLAP applications from models / Blanco C., de Guzmán I.G.R, Fernández-Medina E. - Information and Software Technology - Volume 59, March 2015, – P. 1–16.
30. Gupta A. Amazon Redshift and the Case for Simpler Data Warehouses/ A. Gupta, D. Agarwal, D. Tan, J. Kulesza, R. Pathak, S. Stefani, V. Srinivasan // SIGMOD '15 Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data – 2015 – P. 1917-1923.
31. Cuzzocrea A. Data Warehousing and OLAP over Big Data: Current Challenges and Future Research Directions / A. Cuzzocrea, L. Bellatreche, I.-Y. Song // DOLAP '13 Proceedings of the sixteenth international workshop on Data warehousing and OLAP – 2013 – ACM New York, NY, USA – P. 67-70.
32. Кучук Г.А. Метод синтезу логічної структури мережевої бази даних / Г. А. Кучук // Системи обробки інформації. – 2001. – № 2. – С. 32-37.

33. Приймак Д.В., Акимішин О.І. Шляхи побудови конфігурованих сховищ даних на основі платформи Hadoop. [Електронний ресурс] / Д.В. Приймак, О.І. Акимішин // Режим доступу: <http://eom.lp.edu.ua/seminar/spr/pryjmak.doc>.

34. Шаховська Н.Б. Сховища та простори даних. Монографія / Н. Б. Шаховська, В.В. Пасічник // Львів: Видавництво Львівської політехніки, 2009. – 244 с.

35. Horng J.T. A mechanism for view consistency in a data warehousing system / J.T. Horng, C.-W. Chen // Journal of Systems and Software – 2001 – Volume 56, Issue 1, 1 February 2001 – P. 23-37.

36. Inmon W.H. Corporate information factory / W. H. Inmon, C. Imhoff, R. Sousa // John Wiley & Sons, 2002 – 400 p.

37. Kimball R. The Data Warehouse Toolkit: Practical Techniques for Building Dimensional Data Warehouse / New York : John Wiley & Sons, 2000 – 374 p.

38. Яцишин А. Ю. Інформаційна технологія побудови розподілених сховищ даних гібридного типу: Дисс...канд. тех. наук 05.13.06 / А. Ю. Яцишин. – Київ: НТУУ «КПІ», 2016. – 192 с.

39. Українець О. Є., Удовенко С. Г., Чала Л. Е., Шергін В. Л. Нейромережева технологія розпізнавання зображень з використанням клітинних автоматів. Біоніка інтелекту. 2022. № 1 (98). С. 19–28. DOI: 10.30837/bi.2022.1(98).02.

40. Kyrychenko I., Nazarov A., Kozel N., Gruzdo I. Security in decentralized databases. Біоніка інтелекту. – Х.: ХНУРЕ. – №1 (92), 2019. – с. 59–64.

41. Kyrychenko I., Shyshlo O., Shanidze N., Minimizing Security Risks and Improving System Reliability in Blockchain Applications: a Testing Method Analysis. CEUR-WS, 2023, v. 3403, Volume III: Intelligent Systems Workshop, pp. 423–433 ISSN 16130073.

42. Kopp A., Orlovskiy D., Litvinova U.. Development and research of software solution for business process model correctness analysis using machine

learning. Bulletin of National Technical University "KhPI". Series: System Analysis, Control and Information Technologies. 2024. № 1(11). C. 35–46. DOI: 10.20998/2079-0023.2024.01.06.

43. Shaohua J., Na W., Jing W., et al. (2019). Combining BIM and Ontology to Facilitate Intelligent Green Building Evaluation, 33(1), 04018062. doi: 10.1061/(ASCE)CP.1943-5487.0000786.

44. Céline F., John B., et al. (2018). Choosing the best algorithm for event detection based on the intended application: A conceptual framework for syndromic surveillance. Journal of Biomedical Informatics, 86, 117-129. doi: 10.1016/j.jbi.2018.08.001.

45. Yojna A., et al. (2021). A Survey on Deep learning Models for Effective Content Based Image Retrieval. International Journal of Computer Science Trends and Technology (IJCST) – Volume 9 Issue 3.

46. Kyrychenko I., Nazarov O., Huliiev N., Avdieiev O.. Selection of Artificial Neural Networks for Disease Prediction. CEUR-WS, 2023, v. 3387, Volume I: Machine Learning Workshop, pp. 236-248. ISSN 16130073.

47. Cherednichenko O., Sharonova N., Pliekhova G., Babkova N. Intelligent Methods of Secure Routing in Software-Defined Networks. CEUR-WS, 2024. v. 3664, pp. 342-351. ISSN 16130073 CEUR-WS, 2024. v. 3664, pp. 342-351. ISSN 16130073.

48. Hongyan S., et al. (2021). Design of the online platform of intelligent library based on machine learning and image recognition. Microprocessors and Microsystems, 3. <https://doi.org/10.1016/j.micpro.2021.103851>.

49. Tong L., Jinzhen W., Qing L., et al. (2023). High-Ratio Lossy Compression: Exploring the Autoencoder to Compress Scientific Data. IEEE Transactions on Big Data, 6(2), 22-36. doi: 10.1109/TBDATA.2021.3066151.

50. Zheng Y., Xie X., Ma W., et al. (2019). Distributed Architecture for Large Scale Image-Based Search. IEEE Xplore. doi: 10.1109/ICME.2007.4284716.

51. Chen M., Yang J. F., et al. (2019). A remote-sensing image-retrieval model based on an ensemble neural network. *Big Earth Data*, 351-367. doi: 10.1080/20964471.2019.1570815.
52. Kopp A., Orlovskiy D., Olkhoviy O. Blockchain platform selection and software development for decentralized exchange of business process models. *Bulletin of National Technical University "KhPI". Series: System Analysis, Control and Information Technologies*. 2023 № 2 (10), С. 36–45. DOI: 10.20998/2079-0023.2023.02.06.
53. Dashenkov D., Smelyakov K., Sharonova N. Dataset for NLP-Enhanced Image Classification. *CEUR-WS*, 2023. v. 3664, pp. 342-351. ISSN 16130073
CEUR-WS, 2024. v. 3664, pp. 88-101. ISSN 16130073.
54. Fawzy D., Moussa S. M. and Badr N. L., "The Internet of Things and Architectures of Big Data Analytics: Challenges of Intersection at Different Domains," in *IEEE Access*, vol. 10, pp. 4969-4992, 2022, doi: 10.1109/ACCESS.2022.3140409.
55. Simran A., et al. (2021). Content Based Image Retrieval Using Deep Learning Convolutional Neural Network. *IOP Conf. Ser.: Mater. Sci. Eng.* 1084 012026. doi: 10.1088/1757-899X/1084/1/012026.
56. Gu C., Bu J., Zhou X. et al. (2022). Cross-modal image retrieval with deep mutual information maximization. *Neurocomputing* Volume 496, 28 July 2022, Pages 166-177. doi: 10.1016/j.neucom.2022.01.078.
57. Hussain A., Li H., Muqadar A. et al. (2022). An Efficient Supervised Deep Hashing Method for Image Retrieval. *Entropy*, Volume 24, Issue 10. doi: 10.3390/e24101425.
58. Мартюк М. Ю., Оксанич І. Г., Шевченко І. В.. Моделі ієрархічної багатоагентної системи для виконання бізнес-процесів. *Вісник КрНУ імені Михайла Остроградського*. 2021. Вип. 6/2021 (131). С. 73-78. DOI: 10.30929/1995-0519.2021.6.73-78.

59. Shevchenko I., Vasyliiev D., Prytchyn S., Samoiloiv A. / Business processes monitoring based on fuzzy cognitive maps // Radioelectronic and Computer Systems, 2022, no. 3(103) doi: 10.32620/reks.2022.3.08.

60. Filatov V. O., Yerokhin A. L., Zolotukhin O. V., Kudryavtseva M. S. Methods of intellectual analysis of processes in medical information systems. Відбір і обробка інформації. 2020. Вип. 48 (124). С. 92–98. DOI: 10.15407/vidbir2020.48.092.

61. Струбицький Р. П. Аналіз підходів до моделювання хмаркових сховищ даних / Р. П. Струбицький, Н. Б. Шаховська. // Актуальні проблеми економіки: Науковий економічний журнал. – 2013. – №11. – С. 263–269.

62. Yamburenko V., Liutenko I., Kopp A., Holovnia D. Algorithm and software of medical personnel selection system. Bulletin of National Technical University "KhPI". Series: System Analysis, Control and Information Technologies. 2023 № 2 (10), С. 46-52. DOI: 10.20998/2079-0023.2023.02.07

63. Струбицький Р. П. Методи та алгоритми побудови хмаркових сховищ даних на основі розподілених телекомунікаційних систем: Дисс.канд. тех. наук 05.12.02 / Р. П. Струбицький. – Львів : Нац. ун-т «Львівська політехніка». – 2017. – 176 с.

64. Ковтун Б. В. Порівняльна характеристика реляційних та NoSQL баз даних [Електронний ресурс] / Б. В. Ковтун, А. М. Манич, О. В. Романюк // Матеріали XLIX науково-технічної конференції підрозділів ВНТУ, Вінниця, 27-28 квітня 2020 р. – Електрон. текст. дані. – 2020. – Режим доступу: <https://conferences.vntu.edu.ua/index.php/all-fitki/all-fitki2020/paper/view/9907>.

65. Швець М. Ю. Порівняння SQL та NoSQL баз даних / М. Ю. Швець, Д. С. Заруба, Ю. В. Хохлов // Вчені зап. ТНУ ім. В. І. Вернадського. – Сер.: техн. науки. – Т. 29, Ч.2. – 2018. – С. 21 – 25.

66. Veen J. S. van der, Waaij B. van der, Meijer R. J. Sensor data storage performance: SQL or NoSQL, physical or virtual in Proceedings of the IEEE 5th International Conference on Cloud Computing (CLOUD '12), pp. 431 – 438, IEEE, June 2012.

67. Agrawal R., Ailamari A., Bernstein P.A., Brewer E. A. and Weikum G., The Claremont report database recache. 2008, 11 p.
68. Li Y., Monohara S. A performance comparison of SQL and NoSQL databases. Conference: Communications, Computers and Signal Processing (PACRIM), 2013 IEEE Pacific Rim Conference on, pp.15-19, IEEE, August 2013
69. Шаров С. В. Огляд нереляційних баз даних / С. В. Шаров, В. В. Петровський // Всеукраїнська Інтернет-конференція «Інтелектуальне суспільство: економічні і технічні аспекти становлення». – 2015. – С. 16-18.
70. Grimes S. Unstructured Data and the 80 Percent Rule, 2008, Clarabridge, Bridgepoints. <http://breakthroughanalysis.com/2008/08/01/unstructured-data-and-the-80-percent-rule/>
71. Vysotska V., Smelyakov K., Sharonova N., Vakulik E., Filipov O., Kotelnikov R.. Fast Color Images Clustering for Real-Time Computer Vision and AI System. CEUR-WS, 2024. v. 3664, pp. 161-177. ISSN 16130073.
72. Кветний Р. Н. Комп'ютерне моделювання систем та процесів. Методи обчислень / Р. Н. Кветний, І. В. Богач, О. Р. Бойко, О. Ю. Софіна, О. М. Шушура. – Частина 2: Навч. посіб. – Вінниця : ВНТУ, 2012. – 230 с.
73. Різуненко А. О. Теорія та практика цифрової обробки зображень: Монографія / А. О. Різуненко. – Полтава: РВВ ПУСКУ, 2009. – 195 с.
74. Глибинне навчання для комп'ютерного зору. Частина 1 / В. В. Дворжак, М. В. Талах – Чернівці: Технодрук, 2022 р. – 271 с.
75. Кумар Б. Р., Джозеф Д. К., Тігер Т. В. С. Сегментація клітин крові на основі енергії // 14-та міжнародна конференція з цифрової обробки сигналів. – ДСП, 2002. – 1–3 липня. – Санторіні, Греція. – Т. 2. – С. 619–622.
76. Бамфорд П. Емпіричне порівняння алгоритмів сегментації клітин з використанням анотованого набору даних // Proc. IEEE Міжнародна конференція з обробки зображень. – 2003. – Т. 2. – С. 1073–1077.
77. Beyer M. A., Laney D. The Importance of “Big Data”: A Definition [Electronic resource] / Mark A. Beyer, Douglas Laney // Gartner Inc. – Electronic

data. – [Stamford: Gartner, 2012]. – Mode of access: World WideWeb: <http://www.gartner.com/id=2057415>

78. Терещенко Г. Ю., Груздо І. В. Застосування симетричних алгоритмів в блокчейні. Біоніка інтелекту. – Харків : ХНУРЕ. – 2020. – № 1 (94). – С. 33–39.

79. Терещенко Г. Ю. Аналіз Методів Стиснення Зображень для Зберігання у Децентралізованих Блокчейн Сховищах / Г. Ю. Терещенко, І. В. Кириченко, К. С. Смеляков, А. Є. Олійник // Біоніка інтелекту. Інформація, мова, інтелект. – Х.: ХНУРЕ. – №1(100), 2024. - с. 23–35. DOI: 10.30837/bi.2024.1(100).04.

80. Koptyra K., Ogiela M., Imagechain – Application of Blockchain Technology for Images // Sensors 2021, 21(1), 82. doi: 10.3390/s21010082

81. Puthal D., Mohanty S., Nanda P., and Choppali U., “Building Security Perimeters to Protect Network Systems against Cyber Threats”, IEEE Consumer Electronics Magazine, Vol. 6, No. 4, pp. 24-27, 2017.

82. Puthal D., Malik N., Mohanty S. P., Kougianos E., and Das G., "Everything you Wanted to Know about the Blockchain", IEEE Consumer Electronics Magazine, Volume 7, Issue 4, July 2018, pp. 06– 14.

83. Grinberg R., “Bitcoin: An Innovative Alternative Digital Currency”, Hastings Science & Technology Law Journal, Vol. 4, pp. 159-208, 2012.

84. Barber S., Boyen X., Shi E., and Uzun E., “Bitter to better -- How to make bitcoin a better currency”, in Proceedings of the International Conference on Financial Cryptography and Data Security, pp. 399-414, 2012.

85. Nakamoto S., “Bitcoin: A Peer-to-Peer Electronic Cash System”, <https://bitcoin.org/bitcoin.pdf>, Last visited 11th November 2017.

86. Zimmerman P. Blockchain structure and cryptocurrency prices // Staff working Paper / Bank of England. – 2020. – No 855, February 2020. – P.1-75. URL: <https://www.bankofengland.co.uk/-/media/boe/files/working-paper/2020/blockchainstructure-and-cryptocurrency-prices.pdf> .

87. Biktimirov M. R. et al. Blockchain technology: Universal structure and requirements // *Automatic Documentation and Mathematical Linguistics*. – 2017. – Vol. 51 (6). – P. 235-238. URL: <https://link.springer.com/article/10.3103/S0005105517060036>.

88. Zheng Z. et al. Blockchain challenges and opportunities: A survey // *International journal of web and grid services*. – 2018. – Vol. 14, No 4. – P. 352-375. URL: <https://allquantor.at/blockchainbib/pdf/zheng2018blockchain.pdf>.

89. Belotti M. et al. A vademecum on blockchain technologies: When, which, and how // *IEEE Communications Surveys & Tutorials*. – 2019. – Vol. 21 (4). – P. 3796-3838. URL: <https://ieeexplore.ieee.org/document/8760539>

5. Nofer M. et al. Blockchain // *Business & Information Systems Engineering*. – 2017. – Vol. 59, No 3. – P. 183-187.

90. Kushch S., Prieto-Castrillo F. Blockchain for dynamic nodes in a smart city // *2019 IEEE 5th World Forum on Internet of Things (WF-IoT)*. – IEEE, 2019. – P. 29-34. URL: <https://wfiot.jkjmanagement.com/papers/1570523247.pdf>.

91. Шаронова Н. В. Проблеми і перспективи практичного застосування інформаційної технології blockchain в smart-контрактах / Н. В. Шаронова, І. В. Кириченко, Г. Ю. Терещенко // *Інтелектуальні системи та інформаційні технології (ISIT-2019)*. – Матеріали Міжн. Наук.-практ. Конф. – Одеса, 19 – 24 серпня 2019 р. – С. 214–219.

92. Reed J. *Smart Contracts: The Essential Guide to Using Blockchain Smart Contracts for Cryptocurrency Exchange* / Jeff Reed., 2016. – 54 p.

93. Li X. et al. A survey on the security of blockchain systems // *Future Generation Computer Systems*. – 2020. – Vol. 107. – P. 841-853. URL: <https://www.sciencedirect.com/science/article/abs/pii/S0167739X17318332>.

94. Bhowmik D., Feng T., The multimedia blockchain: a distributed and tamper-proof media transaction framework. // *IEEE 22nd International Conference. Digital Signal Processing (DSP)*, 2017, pp. 1-5. doi: 10.1109/icdsp.2017.8096051.

95. Ouni A., A machine learning approach for image retrieval tasks, 35th International Conference on Image and Vision Computing New Zealand (IVCNZ), Wellington, New Zealand, 2020, pp. 1-5, doi: 10.1109/IVCNZ51579.2020.9290617.
96. Gates M. Blockchain: Ultimate guide to understanding blockchain, bitcoin, cryptocurrencies, smart contracts and the future of money. / Mark Gates., 2017. – 125 p.
97. Tereshchenko G. Copyright protection using blockchain / N. Bilous, I. Kyrychenko. G.Tereshchenko // Біоніка інтелекту. Інформація, мова, інтелект. – X.: ХНУРЕ. – № 1 (92), 2019. – с. 52–58.
98. Rajaraman A., Ullman J. D. "Data Mining." In: Data Integration: The Potential and the Challenges. Proc. of ACM Symp. on Principles of Database Systems, pp. 44-58, June 2001.
99. Chang F., Dean J., et al. "Bigtable: A Distributed Storage System for Structured Data." Trans. on Computer Systems (TOCS), vol. 26, no. 2, pp. 1-26, June 2008.
100. Ghemawat S., Gobioff H., Leung S.-T. "The Google File System." ACM SIGOPS Operating Systems Review, vol. 37, no. 5, pp. 29-43, October 2003.
101. Borthakur D. "The Hadoop Distributed File System: Architecture and Design." Hadoop Project Website, vol. 11, no. 2007, pp. 21-44, September 2007.
102. Leskovec J., Rajaraman A., Ullman J. D. "Mining of Massive Datasets." Cambridge University Press, vol. 1, no. 3, pp. 77-134, November 2014.
103. Le Q. V., Ngiam J., Chen Z., Chia D., Koh P. W., Ng A. Y. "Tiled Convolutional Neural Networks." Advances in Neural Information Processing Systems, pp. 1279-1287, December 2010.
104. Simonyan K., Zisserman A. "Very Deep Convolutional Networks for Large-scale Image Recognition." Proc. of the International Conf. on Learning Representations, pp. 1-14, May 2015.
105. Smelyakov K., Chupryna A., Sandrkin D. and Kolisnyk M., "Search by Image Engine for Big Data Warehouse," 2020 IEEE Open Conference of Electrical,

Electronic and Information Sciences (eStream), Vilnius, Lithuania, 2020, pp. 1-4, doi: 10.1109/eStream50540.2020.9108782.

106. Dean J., Ghemawat S. "MapReduce: Simplified Data Processing on Large Clusters." *Communications of the ACM*, vol. 51, no. 1, pp. 107-113, January 2008.

107. Zaharia M., Chowdhury M., Das T., Dave A., Ma J., McCauley M., et al. "Resilient Distributed Datasets: A Fault-tolerant Abstraction for In-memory Cluster Computing." *Proc. of the 9th USENIX Conf. on Networked Systems Design and Implementation*, pp. 2-2, April 2012.

108. Sharonova N., Kyrychenko I., Tereshchenko G. Application of big data methods in E-learning systems. 5th International Conference on Computational Linguistics and Intelligent Systems (COLINS-2021), Kharkiv, Ukraine, April 22-23, 2021. – CEUR Workshop Proceedings 2870, Volume I, PP. 1302-1311, ISSN 6130073.

109. Kyrychenko I., Tereshchenko G., Proniuk G., Geseleva N. Predicate Clustering Method and its Application in the System of Artificial Intelligence. 2023 7th International Conference on Computational Linguistics and Intelligent Systems (COLINS-2023), 2023. – CEUR-WS, 2023, ISSN 16130073. - Volume 3396, PP. 395 – 406.

110. Greenberg A. G., Hamilton J. R., Maltz D. A., Patel P. The Cost of a Cloud: Research Problems in Data Center Networks. *ACM SIGCOMM Computer Communication Review*, vol. 39, no. 1, pp. 68-73, January 2009.

111. Krizhevsky A., Sutskever I., Hinton G. E. ImageNet Classification with Deep Convolutional Neural Networks. *Advances in Neural Information Processing Systems*, pp. 1097-1105, December 2012.

112. Mikolov T., Chen K., Corrado G., Dean J. Efficient Estimation of Word Representations in Vector Space. *arXiv preprint arXiv:1301.3781*, pp. 1-12, January 2013.

113. Taigman Y., Yang M., Ranzato M., Wolf L. "DeepFace: Closing the Gap to Human-level Performance in Face Verification." Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition, pp. 1701-1708, June 2014.
114. Bieliievstov S., Ruban I., Smelyakov K. and Sumtsov D., "Network technology for transmission of visual information", Selected Papers of the XVIII International Scientific and Practical Conference "Information Technologies and Security" (ITS 2018), Kyiv, Ukraine, November 27, 2018. In CEUR Workshop Proceedings, Vol-2318, 2018, pp. 160-175. <https://ceur-ws.org/Vol-2318/>.
115. Szeliski R., "Computer Vision: Algorithms and Applications." Springer Science & Business Media, pp. 101-150, September 2010.
116. Han L., Wenqing W., Pengfei J., et al. (2019). Content Based Image Retrieval via Sparse Representation and Feature Fusion. The 2019 IEEE 8th Data Driven Control and Learning Systems Conference. Doi: 10.1109/DDCLS.2019.8908926.
117. Peikun X., Enchen M., Zaihua X., et al. (2021). Cloud Computing Image Recognition System Assists the Construction of the Internet of Things Model of Administrative Management Event Parameters. Advances in Computational Intelligence Techniques for Next Generation Internet of Things. doi: 10.1155/2021/8630256.
118. Anna M. B., Osama E., Francesca M., et al. (2019). Image data reduction and big data analysis for targeted biopsy of prostate cancer. Journal of Medical Systems. doi: 10.1007/s00261-015-0353-8.
119. Noha A. S., Ali.I. ELdesouky, Hesham A., et al. (2018). An efficient fast-response content-based image retrieval framework for big data. Computers & Electrical Engineering. doi: 10.1016/j.compeleceng.2016.04.015.
120. Amirhessam T., Anahid E., Behshad M., Amir H G., Katja Pinker, et al. (2019). Big data analytics in medical imaging using deep learning. Big Data: Learning, Analytics, and Applications. doi: 10.1117/12.2516014.

121. Li J. S., Liu I. H., Tsai C. J., Su Z. Y., Li C. F. and Liu C. G., "Secure Content-Based Image Retrieval in the Cloud With Key Confidentiality," in *IEEE Access*, vol. 8, pp. 114940–114952, 2020, doi: 10.1109/ACCESS.2020.3003928.
122. Wan J., et al. (2018). Deep Learning for Content-Based Image Retrieval: A Comprehensive Study. *MM'14: Proceedings of the 22nd ACM International Conference on Multimedia*, November 2014, Pages 157–166. doi.org/10.1145/2647868.2654948.
123. Shaojuan L., Lizhi W., Jia L., et al. (2020). Image Classification Algorithm Based on Improved AlexNet. *Journal of Physics: Conference Series*. doi: 10.1088/1742-6596/1813/1/012051.
124. Ruqia B., Zahid M., Asmaa M., et al. (2022). Deep features optimization based on a transfer learning, genetic algorithm, and extreme learning machine for robust content-based image retrieval. doi: 10.1371/journal.pone.0274764.
125. Niu D., Zhao X., Lin X., et al. (2020). A novel image retrieval method based on multi-features fusion. *Signal Processing: Image Communication Volume 87*, September 2020, 115911. doi: 10.1016/j.image.2020.115911.
126. Bamidele A., Stentiford F. W. M., Morphett J. (2019). An Attention-Based Approach to Content-Based Image Retrieval *BT Technology Journal* volume 22, pages 151–160.
127. Smelyakov K., Chupryna A., Bohomolov O. and Ruban I. The Neural Network Technologies Effectiveness for Face Detection. 2020 *IEEE Third International Conference on Data Stream Mining & Processing (DSMP)*, 2020, pp. 201-205, doi: 10.1109/DSMP47368.2020.9204049.
128. Jiang D., Kim J. (2021). Image Retrieval Method Based on Image Feature Fusion and Discrete Cosine Transform. *Applied Computer Vision and Pattern Recognition*. doi: 10.3390/app11125701.
129. Smelyakov K., Chupryna A., Bohomolov O. and Hunko N. The Neural Network Models Effectiveness for Face Detection and Face Recognition. 2021 *IEEE*

Open Conference of Electrical, Electronic and Information Sciences (eStream), 2021, pp. 1-7, doi: 10.1109/eStream53087.2021.9431476.

130. Branch Richard, Tjeerdsma Heather, Wilson Cody, Hurley Richard, Mcconnell Sabine. (2014). Cloud Computing and Big Data: A Review of Current Service Models and Hardware Perspectives. *Journal of Software Engineering and Applications*. 7. 686-693. 10.4236/jsea.2014.78063.

131. Hussein E. S., El-Bastawissy A., Hazman M., et al. (2020). Lake Data Warehouse Architecture for Big Data Solutions. *International Journal of Advanced Computer Science and Applications*. doi: 10.14569/IJACSA.2020.0110854.

132. Liu, W., et al. Deep Semantic Image Indexing for Efficient Content-Based Retrieval. // *IEEE Transactions on Multimedia*. 2023, pp. 112787-112803. DOI:10.1109/ACCESS.2023.3324133.

133. Muniswamaiah M., Agerwala T., Tappert C. (2019). Big data in cloud computing review and opportunities. *International Journal of Computer Science & Information Technology (IJCSIT) Vol 11, No 4*.

134. Tchagna Kouanou, Aurelle & Tchiotsop, Daniel & Kengne, Romanic & Djoufack Tansaa, Zephirin & Adele, Ngo & Tchinda, René. (2018). An optimal big data workflow for biomedical image analysis. *Informatics in Medicine Unlocked*. 11. 10.1016/j.imu.2018.05.001.

135. Blockchain Technology Explained: Powering Bitcoin. URL: <https://www.ibm.com/topics/what-is-blockchain> (date of access: 27.11.2023).

136. Patel, A., Sai, S., Daiya, A., Akolekar, H., & Chamola, V. Blockchain enabled traceability in the jewel supply chain // *Scientific Reports*, 2025. 15(1), Article number: 3837, pp.1-14. <https://doi.org/10.1038/s41598-025-88245-4>.

137. Huckle S. Internet of Things, Blockchain and Shared Economy Applications / S. Huckle, R. Bhattacharya, M. White, N. Beloff // *Procedia Comput. Science*. – Oct. 2016. – Vol. 98. – P. 461–466.

138. Pennebaker W., Mitchell J., *JPEG: Still Image Data Compression Standard*. – Springer New York, NY, 1992. – 638p. – ISBN: 978-0-442-01272-4.

139. Bilous N., Tereshchenko G. Copyright protection using blockchain. *Біоніка інтелекту – Харків : ХНУРЕ.* – 2019. – № 1 (92). – С. 52-58.
140. Saji A. et al. Enhancing Image Security with Introduction to Blockchain. // *IJERA* , 2023, Volume 3, Issue 1, pp.394-399. DOI: 10.5281/zenodo.8013640.
141. GIF File Format Summary. URL: <https://www.fileformat.info/format/gif/egff.htm> (date of access: 07.05.2024).
142. LZW compression. URL: <https://www.techtarget.com/whatis/definition/LZW-compression> (date of access: 07.05.2024).
143. Білоконенко В. М., Ревенчук І. А. Алгоритми сегментації зображень на базі побудови матриць збігів. *Східно-Європейський журнал передових технологій.*- №2(62), том 2.-2013.-С.43-45.
144. Специфікація алгоритму стискання даних формату DEFLATE версія 1.3 (RFC1951). URL : <http://www.kytok.org.ua/post/cpetsyfikatsiya-alhorytmu-styskannya-danykh-formatu-deflate-versiya-1-3-rfc1951> (date of access: 07.04.2023).
145. Solem J. *Programming Computer Vision with Python: Tools and algorithms for analyzing images.* O'Reilly Media, Inc., 2012. 264 p.
146. JPEG YCbCr Support. URL: <https://learn.microsoft.com/en-us/windows/win32/wic/jpeg-ycbcr-support> (date of access: 09.04.2023).
147. Wang Z., Bovik A. C., Sheikh H. R., Simoncelli E. P. Image quality assessment: From error visibility to structural similarity // *IEEE Transactions on Image Processing.* – 2004. – Vol. 13, No. 4. – P. 600–612. DOI: 10.1109/TIP.2003.8198.
148. Коптыра, К., Ogiela, M. R. *Imagechain – Application of Blockchain Technology for Images* // *Sensors*, Basel, Switzerland, 2020. 21(1), 82. <https://doi.org/10.3390/s21010082>
149. Шаронова Н. В., Терещенко Г. Ю. Проблеми і перспективи практичного застосування інформаційної технології blockchain в smart-

контрактах. Інтелектуальні системи та інформаційні технології (ISIT-2019). – Матеріали Міжн. Наук.-практ. Конф. – Одеса, 19 – 24 серпня 2019 р. – С. 214–219.

150. Смеляков К.С., Кириченко І.В., Терещенко Г.Ю., Панасенко Д.П. Використання машинного навчання для оптимізації доступу до даних в гібридному сховищі зображень // Біоніка інтелекту. 2023. № 1 (99). с. 11–18. DOI 10.30837/bi.2023.1(99).02.

151. James, Blessing & Asagba, P. (2017). Hybrid Database System for Big Data Storage and Management. *International Journal of Computer Science, Engineering and Applications*. 7. 15-27. 10.5121/ijcsea.2017.7402.

152. Kyrychenko I., Nazarov O., Huliiev N., Avdieiev O. “Selection of Artificial Neural Networks for Disease Prediction”, 2023 7th International Conference on Computational Linguistics and Intelligent Systems (COLINS-2023), 2023. – CEUR-WS, 2023, ISSN 16130073. - Volume 3387, PP. 236-248.

153. Tabian I., Fu H., Khodaei Z.S. (2019). A Convolutional Neural Network for Impact Detection and Characterization of Complex Composite Structures. *Sensors*, 19(22), 4933. DOI: 10.3390/s19224933.

154. Berahmand, Kamal & Daneshfar, Fatemeh & Salehi, Elaheh & Li, Yuefeng & Xu, Yue. (2024). Autoencoders and their applications in machine learning: a survey. *Artificial Intelligence Review*. 57. 10.1007/s10462-023-10662-6.

155. Asokan, Anju & Jude, Anitha & Patrut, Bogdan & Danciulescu, Dana & D, Jude. (2020). Deep Feature Extraction and Feature Fusion for Bi-temporal Satellite Image Classification. *Computers, Materials & Continua*. 66. 373-388. 10.32604/cmc.2020.012364.

156. He K., Zhang X., Ren S., Sun J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770-778. doi:10.1109/CVPR.2016.90.

157. Meghana, Avuthu Sai. "Age and Gender prediction using Convolution, ResNet50 and Inception ResNetV2." *International Journal of Advanced Trends in*

Computer Science and Engineering 9, no. 2 (April 25, 2020): 1328–34.
<http://dx.doi.org/10.30534/ijatcse/2020/65922020>.

158. Yadav S. Understanding Cosine Similarity and Cosine Distance in Depth // AI Mind. – 18 September 2023. URL: <https://pub.aimind.so/understanding-cosine-similarity-and-cosine-distance-in-depth-cc91eac3ef2> (дата звернення: 29.04.2024).

159. Matallah, Houcine & Belalem, Ghalem & Bouamrane, K.. (2020). Evaluation of NoSQL Databases: MongoDB, Cassandra, HBase, Redis, Couchbase, OrientDB. International Journal of Software Science and Computational Intelligence. 12. 71-91. 10.4018/IJSSCI.2020100105.

160. Znakhur Serhii, Znakhur Liudmyla. (2022). Similar goods search based on FAISS. Bulletin of Kharkov National Automobile and Highway University. 40. 10.30977/BUL.2219-5548.2022.96.0.40.

161. Wong, Wai-Tak. (2019). Advanced Elasticsearch 7.0: A practical guide to designing, indexing, and querying advanced distributed search engines.

162. Keras Applications: Res Net. URL: <https://keras.io/api/applications/resnet/> (дата звернення: 20.04.2024).

163. Nelli, Fabio. (2018). The NumPy Library: With Pandas, NumPy, and Matplotlib. 10.1007/978-1-4842-3913-1_3.

164. Animal Image Dataset (90 Different Animals) on Kaggle. URL: <https://www.kaggle.com/datasets/iamsouravbanerjee/animal-image-dataset-90-different-animals> (дата звернення: 04.05.2024).

165. Fei-Fei L., Fergus R., Perona P. "One-shot Learning of Object Categories." IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 28, no. 4, pp. 594-611, April 2006.

166. Szegedy C., Liu W., Jia Y., Sermanet P., Reed S., Anguelov D., et al. "Going Deeper with Convolutions." Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition, pp. 1-9, June 2015.

167. Cevikalp H., Triggs B. "Face Recognition Based on Image Sets." Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition, pp. 2567-2573, June 2010.
168. Терещенко Г. Ю. Аналіз і обґрунтування використання наявних блокчейн-рішень для захисту цифрових активів / І. В. Кириченко, Г. Ю. Терещенко, // Сучасний стан наукових досліджень та технологій в промисловості. – Х.: ХНУРЭ. – №1 (27), 2024. – с. 164-178 DOI: 10.30837/ITSSI.2024.27.164.
169. Exploring the Darkverse: A Multi-Perspective Analysis of the Negative Societal Impacts of the Metaverse / Y. K. Dwivedi et al. Information Systems Frontiers. 2023. Vol. 25. No. 11. P. 2071–2114. DOI: <https://doi.org/10.1007/s10796-023-10400-x>.
170. Hossain M. S. What do we know about cryptocurrency? Past, present, future. China Finance Review International. Vol. 11. No. 4. 2021. P. 552–572. DOI: <https://doi.org/10.1108/cfri-03-2020-0026>.
171. Holochain: An Agent-Centric Distributed Hash Table Security in Smart IoT Applications / S. Gaba et al. IEEE Access. 2023. Vol. 11. P. 81205–81223. DOI: <https://doi.org/10.1109/access.2023.3300220>.
172. Digital transformation in tourism and hospitality industry: a literature review of blockchain, financial technology, and knowledge management / S. Ratna et al. EuroMed Journal of Business. 2023. DOI: <https://doi.org/10.1108/emjb-04-2023-0118>.
173. Review of Privacy Enhancement Methods for Federated Learning in Healthcare Systems / X. Gu et al. International Journal of Environmental Research and Public Health. 2023. Vol. 20. No. 15. Art. no. 6539. DOI: <https://doi.org/10.3390/ijerph20156539/>.
174. Blockchain technology-based sustainable management research: the status quo and a general framework for future application / W. Du et al. Environmental Science and Pollution Research. 2022. Vol. 29. No. 39. P. 58648–58663. DOI: <https://doi.org/10.1007/s11356-022-21761-2>.

175. Shamsuzzoha A., Marttila J., Helo P. Blockchain-enabled traceability system for the sustainable seafood industry. *Technology Analysis and Strategic Management*. 2023. DOI: <https://doi.org/10.1080/09537325.2023.2233632>.

176. Shubin I., Karataiev O. Reuse of information based on the interpretation of knowledge. *Innovative technologies and scientific solutions for industries*. Vol. 2 No. 24. 2023. P. 62–71. URL: <https://doi.org/10.30837/itssi.2023.24.062>.

177. Securing AI-based healthcare systems using blockchain technology: A state-of-the-art systematic literature review and future research directions / R. Shinde et al. *Transactions on Emerging Telecommunications Technologies*. 2024. Vol. 35. No. 1. DOI: <https://doi.org/10.1002/ett.4884>.

178. Rico-Peña J. J., Arguedas-Sanz R., López-Martin C. Models used to characterise blockchain features. A systematic literature review and bibliometric analysis. *Technovation*. Vol. 123. Art. no. 102711. 2023. DOI: <https://doi.org/10.1016/j.technovation.2023.102711>.

179. Zhang T., Jia F., Chen L. Blockchain adoption in supply chains: implications for sustainability. *Production Planning and Control*. 2024. DOI: <https://doi.org/10.1080/09537287.2023.2296669>.

180. Adoption and Applications of Blockchain Technology in Marketing: A Retrospective Overview and Bibliometric Analysis / M. Wasiq et al. *Sustainability (Switzerland)*. 2023. Vol. 15. No. 4. Art. no. 3279. DOI: <https://doi.org/10.3390/su15043279>.

181. Navigating the Future: Blockchain's Impact on Accounting and Auditing Practices / S. Sheela et al. *Sustainability*. 2023. Vol. 15. No. 24. Art. no. 16887. DOI: <https://doi.org/10.3390/su152416887>.

182. Emerging Trends in Blockchain Technology and Applications: A Review and Outlook / A. G. Gad et al. *Journal of King Saud University - Computer and Information Sciences*. 2022. Vol. 34. No. 9. P. 6719–6742. DOI: <https://doi.org/10.1016/j.jksuci.2022.03.007>.

183. Persistence in the cryptocurrency market: does size matter? / A. Plastun et al. *Investment Management and Financial Innovations*. 2023. Vol. 20. No. 4. P. 138–146. DOI: [https://doi.org/10.21511/imfi.20\(4\).2023.12](https://doi.org/10.21511/imfi.20(4).2023.12).
184. Use of cryptocurrencies bitcoin and ethereum in the field of e-commerce: case study of ukraine / V. Suslenko et al. *Financial and Credit Activity Problems of Theory and Practice*. 2022. Vol. 1 No. 42. P. 62–72. DOI: <https://doi.org/10.55643/fcaptp.1.42.2022.3603>.
185. Дамен Дж., Хектор Дж., Перрей Р., Ней Х. Автоматична класифікація еритроцитів за допомогою густин гауссового мікстури // *Proc. Bildverarbeitung für die Medizin*. – 2000. – С. 331–335.
186. Кострарідо Л. Методи аналізу медичних зображень: стратегії оцінки для аналізу медичних зображень. - Тейлор і Френсіс, Сполучені Штати Америки, 2005. – С. 433–471.
187. Кумар Б. Р., Джозеф Д. К., Тігер Т.В.С. Сегментація клітин крові на основі енергії // 14-та міжнародна конференція з цифрової обробки сигналів. – ДСП, 2002. – 1–3 липня. – Санторіні, Греція. – Т. 2. – С. 619–622.
188. Бамфорд П. Емпіричне порівняння алгоритмів сегментації клітин з використанням анотованого набору даних // *Proc. IEEE Міжнародна конференція з обробки зображень*. – 2003. – Т. 2. – С. 1073–1077.
189. Мукерджі Д. П., Рей Н., Ектон С. Т. Аналіз наборів рівнів для виявлення та відстеження лейкоцитів // *ІП*. – 2004. – Т. 13. – No 4. – С. 562–572.
190. Парк Дж., Келлер Дж.М. Змії на вододілі // *IEEE Transactions on Pattern Analysis and Machine Intelligence*. – ПАМІ, 2001. – Т. 23. – No 10. – С. 1201–1205.
191. Сінха Н., Рамакрішнан А. Г. Сегментація клітин крові за допомогою алгоритму EM // Третя індійська конференція з комп'ютерного зору. Обробка графічних зображень (ICVGIP), 2002. – Ахмадабад, Індія, 2002, 16–18 грудня. – С. 376–382.

192. Кумар Р. С., Верма А., Сінгх Дж. Сегментація кольорових зображень і багаторівневе порогове значення шляхом максимізації умовної ентропії // Міжнародний журнал обробки сигналів. – 2006. – Т. 3. – No 1. – С. 121–125.
193. Макінерні Т., Терзопулос Д. Деформівні моделі в аналізі медичних зображень: опитування // *Med Image Anal.* – 1996. – С. 91–108.
194. Кормен Т. Х., Лейзорсон К. Е., Рівест Р. Л. Вступ до алгоритмів. – Видавництво Массачусетського технологічного інституту, 1990. – С. 185–191.
195. Кенні Дж. Ф. Обчислювальний підхід до виявлення країв // *IEEE Trans. Pattern Analysis and Machine Intelligence.* – 1986. – С. 679–698.
196. Jia Y., Shelhamer E., Donahue J., Karayev S., Long J., Girshick R., et al. "Caffe: Convolutional Architecture for Fast Feature Embedding." *Proc. of the ACM International Conf. on Multimedia*, pp. 675-678, November 2014.
197. Bradski G. "The OpenCV Library." *Doctor Dobb's Journal*, vol. 25, no. 11, pp. 120-126, November 2000.
198. Schroff F., Kalenichenko D., Philbin J. FaceNet: A Unified Embedding for Face Recognition and Clustering // *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2015)*. – Boston, MA: IEEE, 2015. – P. 815–823. – <https://doi.org/10.1109/CVPR.2015.7298682>.

НАУКОВЕ ВИДАННЯ

Кириченко І.В., Терещенко Г.Ю., Шанідзе Н.О., Шаронова Н.В.

**ЗБЕРІГАННЯ ВЕЛИКИХ ДАНИХ В
ІНТЕЛЕКТУАЛЬНИХ СИСТЕМАХ**

[Монографія]

Рекомендовано Науково-технічною радою

Харківського національного університету радіоелектроніки

(протокол № 1 від 30. 01. 2026 року)

Рекомендовано Вченою радою Національного технічного університету

«Харківський політехнічний інститут»

(протокол № 3 від 27. 02. 2026 року)



КИРИЧЕНКО ІРИНА ВІТАЛІЇВНА

ДОЦЕНТКА КАФЕДРИ ПРОГРАМНОЇ ІНЖЕНЕРІЇ ХАРКІВСЬКОГО НАЦІОНАЛЬНОГО УНІВЕРСИТЕТУ РАДІОЕЛЕКТРОНІКИ, КАНДИДАТКА ТЕХНІЧНИХ НАУК, ДОЦЕНТКА.



ТЕРЕЩЕНКО ГЛІБ ЮРІЙОВИЧ

ДОЦЕНТ КАФЕДРИ ПРОГРАМНОЇ ІНЖЕНЕРІЇ ХАРКІВСЬКОГО НАЦІОНАЛЬНОГО УНІВЕРСИТЕТУ РАДІОЕЛЕКТРОНІКИ, ДОКТОР ФІЛОСОФІЇ (PHD).



ШАНІДЗЕ НАДІЯ ОЛЕКСАНДРІВНА

ДОЦЕНТКА КАФЕДРИ СОЦІОЛОГІЇ І ПУБЛІЧНОГО УПРАВЛІННЯ НАЦІОНАЛЬНОГО ТЕХНІЧНОГО УНІВЕРСИТЕТУ «ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ», КАНДИДАТКА СОЦІОЛОГІЧНИХ НАУК, ДОЦЕНТКА



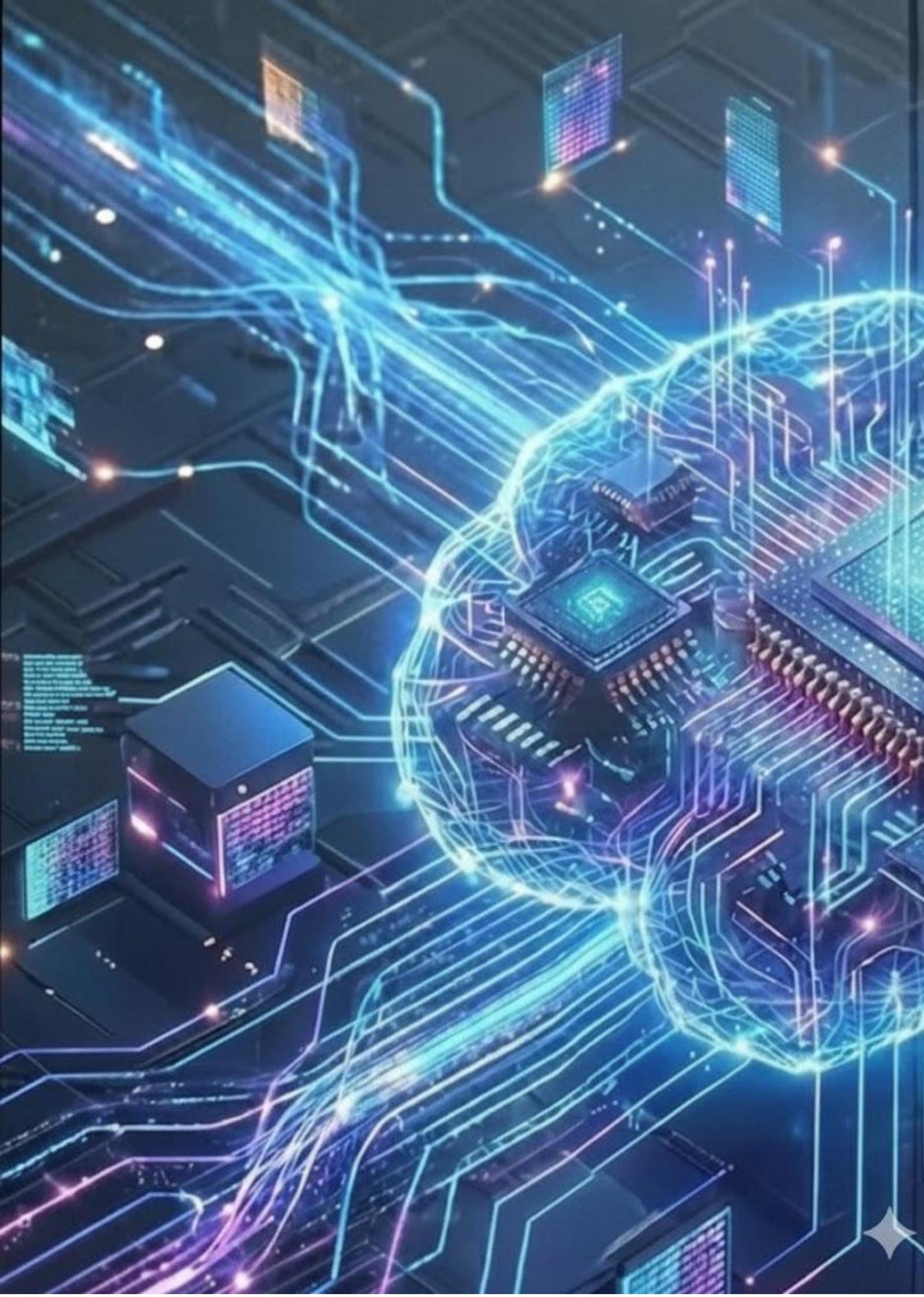
ШАРОНОВА НАТАЛІЯ ВАЛЕРІЇВНА

ПРОФЕСОРКА КАФЕДРИ ІНТЕЛЕКТУАЛЬНИХ КОМП'ЮТЕРНИХ СИСТЕМ НАЦІОНАЛЬНОГО ТЕХНІЧНОГО УНІВЕРСИТЕТУ «ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ», ДОКТОРКА ТЕХНІЧНИХ НАУК, ПРОФЕСОРКА.

Монографія містить опис сучасних уявлень про зберігання та обробку великих даних на основі вітчизняної та світової теорії та практики.

У монографії представлені результати досліджень, які виконують актуальне наукове завдання створення методів для формування гібридної моделі сховища зображень в умовах великих даних, що має істотне значення для розвитку комп'ютерних наук, зокрема у галузі штучного інтелекту.

Для інженерів, науковців, викладачів, аспірантів, діяльність яких пов'язана з роботою з різними типами даних, такими як зображення, звук, відео, текст тощо.



Digitally, we are
connected to a
world of
information
that is constantly
evolving. The
pace of change
is accelerating,
and the stakes
are high. We
must embrace
the future,
or risk being
left behind.

