

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ  
імені ІГОРЯ СІКОРСЬКОГО»**

**Навчально-науковий інститут прикладного системного аналізу  
Кафедра штучного інтелекту**

До захисту допущено:  
Завідувач кафедри  
\_\_\_\_\_ Ірина ДЖИГИРЕЙ  
«\_\_» \_\_\_\_\_ 2026 р.

**Дипломна робота**  
**на здобуття ступеня бакалавра**  
**за освітньо-професійною програмою «Системи і методи штучного інтелекту»**  
**спеціальності 122 «Комп'ютерні науки»**  
**на тему: «Інтелектуальна система виявлення заборонених предметів у**  
**багажі авіапасажирів»**

Виконала:

студентка IV курсу, групи КІ-22

Міхєєва Аліса Євгенівна \_\_\_\_\_

Керівник:

професорка кафедри ММСА, д.т.н., проф.

Кузнєцова Наталія Володимирівна \_\_\_\_\_

Консультант з нормоконтролю:

фахівець першої категорії кафедри ІІІ,

к.т.н., доц. Комариста Богдана Миколаївна \_\_\_\_\_

Рецензент:

завідувачка кафедри електронної інженерії,

к.т.н., доц. Іванько Катерина Олегівна \_\_\_\_\_

Засвідчую, що у цій дипломній роботі немає  
запозичень з праць інших авторів без  
відповідних посилань.

Студентка Міхєєва А.Є. \_\_\_\_\_

Київ – 2026 року

**Національний технічний університет України**  
**«Київський політехнічний інститут імені Ігоря Сікорського»**  
**Навчально-науковий інститут прикладного системного аналізу**  
**Кафедра штучного інтелекту**

Рівень вищої освіти – перший (бакалаврський)  
Спеціальність – 122 «Комп’ютерні науки»  
Освітня програма «Системи і методи штучного інтелекту»

ЗАТВЕРДЖУЮ  
Завідувач кафедри  
\_\_\_\_\_ Ірина ДЖИГИРЕЙ  
«\_\_» \_\_\_\_\_ 2026 р.

**ЗАВДАННЯ**  
**на дипломну роботу студентці**  
**Міхєєвій Алісі Євгенівні**

1. Тема роботи «Інтелектуальна система виявлення заборонених предметів у багажі авіапасажирів», керівник роботи Кузнецова Наталія Володимирівна, доктор технічних наук, професор, затверджені наказом по університету від 27.05.2026 р. № 1944-с.

2. Термін подання студентом роботи «05» червня 2026 р.

3. Вихідні дані до роботи: науково-методична та спеціалізована література за темою дослідження; нормативно-правові документи у сфері авіаційної безпеки; набір рентгенівських зображень багажу SIXray; програмні засоби Python, PyTorch, Ultralytics YOLO, torchvision, OpenCV, Albumentations та Gradio; результати навчання й тестування моделей YOLOv8n, Faster R-CNN та Deformable DETR.

4. Зміст роботи: аналіз предметної області та сучасних підходів до виявлення заборонених предметів; проектування інтелектуальної системи виявлення заборонених предметів у багажі авіапасажирів; розробка та експериментальне дослідження інтелектуальної системи.

5. Перелік ілюстративного матеріалу (із зазначенням плакатів, презентацій тощо): презентація, архітектура інтелектуальної системи; структура набору даних SIXray; результати навчання моделей YOLOv8n, Faster R-CNN та Deformable DETR; матриці помилок досліджуваних моделей; приклади виявлення заборонених предметів на рентгенівських зображеннях; порівняльний аналіз ефективності моделей; інтерфейс розробленої системи; презентація.

6. Дата видачі завдання \_\_\_\_\_

### Календарний план

| № з/п | Назва етапів виконання дипломної роботи                                                      | Термін виконання етапів роботи | Примітка |
|-------|----------------------------------------------------------------------------------------------|--------------------------------|----------|
| 1     | Затвердження теми бакалаврської дипломної роботи (БДР), опанування ДСТУ 3008:2015            | 20.04.2026-21.04.2026          | виконано |
| 2     | Огляд властивостей графів                                                                    | 22.04.2026-26.04.2026          | виконано |
| 3     | Розбір основних теоретичних конструкцій клітинних автоматів                                  | 27.04.2026-03.05.2026          | виконано |
| 4     | Проектування алгоритмів і розробка програмного забезпечення                                  | 04.05.2026-10.05.2026          | виконано |
| 5     | Оформлення пояснювальної записки (ПЗ), перевірка правильності структури та оформлення        | 11.05.2026-18.05.2026          | виконано |
| 6     | Оновлення програмного забезпечення та відповідне оформлення розділу програмного забезпечення | 19.05.2026-25.05.2026          | виконано |
| 7     | Остаточне оформлення ПЗ, підготовка презентації                                              | 25.05.2026-31.05.2026          | виконано |
| 8     | Передзахист БДР                                                                              |                                | виконано |

Студентка \_\_\_\_\_

Аліса МІХЄЄВА

Керівник \_\_\_\_\_

Наталія КУЗНЄЦОВА

## РЕФЕРАТ

Дипломна робота: 121 с., 31 рис., 18 табл., 56 джерел, додаток.

ІНТЕЛЕКТУАЛЬНА СИСТЕМА, ВИЯВЛЕННЯ ЗАБОРОНЕНИХ ПРЕДМЕТІВ, РЕНТГЕНІВСЬКІ ЗОБРАЖЕННЯ, КОМП'ЮТЕРНИЙ ЗІР, OBJECT DETECTION, YOLOv8n, FASTER R-CNN, DEFORMABLE DETR, АВІАЦІЙНА БЕЗПЕКА.

Об'єктом дослідження є процес автоматизованого виявлення заборонених предметів на рентгенівських зображеннях багажу авіапасажирів.

Предметом дослідження є методи та моделі комп'ютерного зору для детекції заборонених предметів.

Метою роботи є розробка інтелектуальної системи виявлення заборонених предметів у багажі авіапасажирів та дослідження ефективності сучасних моделей детекції об'єктів.

У роботі проаналізовано сучасні методи комп'ютерного зору та набори даних для аналізу рентгенівських зображень. Розроблено архітектуру системи та реалізовано модулі детекції на основі моделей YOLOv8n, Faster R-CNN і Deformable DETR.

Експериментальні дослідження проведено на наборі даних SIXray. За результатами порівняльного аналізу встановлено, що модель YOLOv8n забезпечує найкращий баланс між точністю та повнотою виявлення об'єктів.

Практичним результатом роботи є програмний прототип системи, який виконує автоматичне виявлення заборонених предметів на рентгенівських зображеннях та може використовуватися як система підтримки прийняття рішень під час контролю багажу.

## ABSTRACT

Bachelor's thesis: 121 pages, 31 figures, 18 tables, 56 references, appendix.

INTELLIGENT SYSTEM, PROHIBITED ITEM DETECTION, X-RAY IMAGES, COMPUTER VISION, OBJECT DETECTION, YOLOv8n, FASTER R-CNN, DEFORMABLE DETR, AVIATION SECURITY.

The object of the study is the process of automated detection of prohibited items in X-ray images of passengers' baggage.

The subject of the study is computer vision methods and models for prohibited item detection.

The aim of the work is to develop an intelligent system for detecting prohibited items in passengers' baggage and to evaluate the effectiveness of modern object detection models.

The paper analyzes modern computer vision approaches and datasets used for X-ray image analysis. An intelligent system architecture was designed, and object detection modules based on YOLOv8n, Faster R-CNN, and Deformable DETR models were implemented.

Experimental studies were conducted using the SIXray dataset. Based on the comparative analysis, YOLOv8n demonstrated the best balance between detection precision and recall.

The practical outcome of the work is a software prototype capable of automatically detecting prohibited items in X-ray images and supporting decision-making during baggage security screening.

## ЗМІСТ

|                                                                                                                |    |
|----------------------------------------------------------------------------------------------------------------|----|
| ВСТУП.....                                                                                                     | 9  |
| РОЗДІЛ 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА СУЧАСНИХ МЕТОДІВ<br>ДЕТЕКЦІЇ ОБ'ЄКТІВ НА РЕНТГЕНІВСЬКИХ ЗОБРАЖЕННЯХ..... | 13 |
| 1.1 Нормативно-правове регулювання перевезення небезпечних предметів в<br>авіації .....                        | 13 |
| 1.2 Принцип роботи рентгенівських систем контролю багажу.....                                                  | 16 |
| 1.2.1 Будова рентгенівського сканера .....                                                                     | 16 |
| 1.2.2 Принцип формування X-ray image .....                                                                     | 19 |
| 1.2.3 Двоенергетичне рентгенівське сканування.....                                                             | 21 |
| 1.2.4 Проблеми аналізу рентгенівських знімків .....                                                            | 23 |
| 1.3 Методи комп'ютерного зору для аналізу рентгенівських зображень .....                                       | 23 |
| 1.3.1 Класичні методи комп'ютерного зору .....                                                                 | 23 |
| 1.3.2 Методи глибокого навчання .....                                                                          | 25 |
| 1.3.3 Методи object detection .....                                                                            | 26 |
| 1.4 Особливості виявлення заборонених предметів у багажі .....                                                 | 28 |
| 1.5 Аналіз сучасних досліджень у сфері автоматизованого контролю багажу                                        | 30 |
| 1.6 Аналіз наборів даних для дослідження рентгенівських зображень .....                                        | 32 |
| 1.7 Аналіз сучасних моделей виявлення заборонених предметів .....                                              | 35 |
| 1.8 Порівняння результатів сучасних досліджень.....                                                            | 38 |
| 1.9 Основні проблеми та перспективи розвитку систем автоматизованого<br>виявлення заборонених предметів .....  | 40 |
| Висновки до розділу 1 .....                                                                                    | 44 |
| РОЗДІЛ 2 ПРОЄКТВАННЯ ІНТЕЛЕКТУАЛЬНОЇ СИСТЕМИ ВИЯВЛЕННЯ<br>ЗАБОРОНЕНИХ ПРЕДМЕТІВ .....                          | 45 |
| 2.1 Постановка задачі .....                                                                                    | 45 |
| 2.2 Архітектура інтелектуальної системи виявлення заборонених предметів<br>у багажі авіапасажирів .....        | 46 |

|                                                             |                                                                |           |
|-------------------------------------------------------------|----------------------------------------------------------------|-----------|
| 2.3                                                         | Опис датасету.....                                             | 49        |
| 2.4                                                         | Попередня обробка рентгенівських зображень.....                | 50        |
| 2.5                                                         | Проектування модулів виявлення заборонених предметів .....     | 53        |
| 2.6                                                         | Логіка ручної перевірки та взаємодія з інспектором .....       | 55        |
|                                                             | Висновки до розділу 2 .....                                    | 57        |
| <b>РОЗДІЛ 3 РОЗРОБЛЕННЯ ТА ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ</b> |                                                                |           |
|                                                             | <b>ІНТЕЛЕКТУАЛЬНОЇ СИСТЕМИ ВИЯВЛЕННЯ ЗАБОРОНЕНИХ ПРЕДМЕТІВ</b> | <b>59</b> |
| 3.1                                                         | Середовище розробки та програмна реалізація системи .....      | 59        |
| 3.2                                                         | Формування експериментального набору даних .....               | 62        |
| 3.2.1                                                       | Формат анотацій та організація даних .....                     | 63        |
| 3.2.2                                                       | Попередня обробка та аугментація.....                          | 65        |
| 3.2.3                                                       | Підготовка даних для навчання моделей .....                    | 67        |
| 3.3                                                         | Результати навчання моделі YOLOv8n .....                       | 67        |
| 3.3.1                                                       | Аналіз процесу навчання .....                                  | 68        |
| 3.3.2                                                       | Оцінювання якості моделі.....                                  | 69        |
| 3.3.3                                                       | Аналіз результатів за окремими класами .....                   | 70        |
| 3.3.4                                                       | Приклади роботи моделі .....                                   | 72        |
|                                                             | Висновки до підрозділу 3.3 .....                               | 72        |
| 3.4                                                         | Результати навчання моделі Faster R-CNN .....                  | 73        |
| 3.4.1                                                       | Налаштування та параметри навчання моделі FASTER R-CNN .....   | 73        |
| 3.4.2                                                       | Аналіз процесу навчання .....                                  | 74        |
| 3.4.3                                                       | Оцінювання якості моделі.....                                  | 75        |
| 3.4.5                                                       | Аналіз результатів детекції.....                               | 78        |
|                                                             | Висновки до підрозділу 3.4 .....                               | 79        |
| 3.5                                                         | Результати навчання моделі DETR .....                          | 79        |
| 3.5.1                                                       | Налаштування та параметри навчання моделі DETR.....            | 79        |
| 3.5.2                                                       | Аналіз процесу навчання .....                                  | 82        |
| 3.5.3                                                       | Оцінювання якості моделі.....                                  | 83        |
| 3.5.4                                                       | Аналіз матриці помилок моделі DETR.....                        | 84        |
| 3.5.5                                                       | Аналіз результатів детекції.....                               | 85        |

|                                                                    |     |
|--------------------------------------------------------------------|-----|
| Висновки до підрозділу 3.5 .....                                   | 87  |
| 3.6 Порівняльний аналіз моделей YOLOv8n, Faster R-CNN та DETR..... | 88  |
| 3.7 Реалізація користувацького інтерфейсу системи .....            | 94  |
| 3.7.1 Архітектура програмного інтерфейсу .....                     | 94  |
| 3.7.2 Реалізація веб-інтерфейсу засобами Gradio.....               | 95  |
| 3.7.3 Аналіз роботи розробленої системи .....                      | 98  |
| Висновки до розділу 3 .....                                        | 101 |
| ВИСНОВКИ .....                                                     | 103 |
| ПЕРЕЛІК ПОСИЛАНЬ .....                                             | 105 |
| ДОДАТОК .....                                                      | 111 |

## ВСТУП

Сучасна цивільна авіація є однією з ключових складових глобальної транспортної інфраструктури. Зростання кількості авіаперевезень та пасажиропотоку призводить до збільшення навантаження на системи авіаційної безпеки, зокрема на служби контролю багажу авіапасажирів. За даними Міжнародної організації цивільної авіації (ICAO), кількість авіапасажирів у світі зросла з 3.57 млрд осіб у 2015 році до 4.67 млрд осіб у 2024 році. Після різкого спаду авіаперевезень у період пандемії COVID-19 світова авіаційна галузь демонструє стабільне відновлення та подальше зростання. У зв'язку з цим значно підвищуються вимоги до ефективності та швидкості роботи систем контролю безпеки в аеропортах. Динаміка зростання кількості авіапасажирів у світі протягом 2015–2024 років наведена на рисунку 1.1.

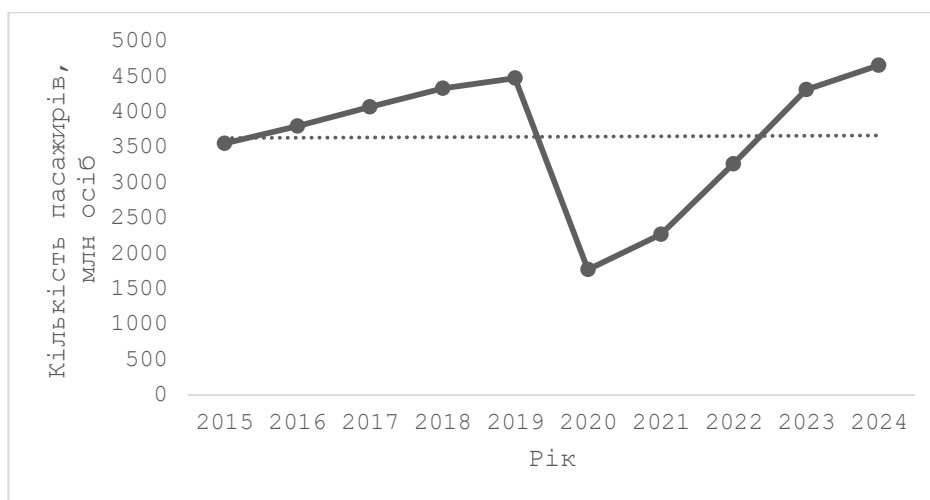


**Рисунок 1.1** – Динаміка кількості авіапасажирів у світі у 2015–2024 роках

Джерело: побудовано автором на основі даних ICAO [1]

Одним із найважливіших етапів забезпечення авіаційної безпеки є перевірка багажу пасажирів із використанням рентгенівських скануючих систем. Основною метою такого контролю є виявлення заборонених предметів,

зокрема зброї, вибухових речовин, боєприпасів, ножів та інших потенційно небезпечних об'єктів. Проте аналіз рентгенівських зображень є складною задачею через низьку контрастність зображень, накладання предметів один на одного, різноманітність форм та матеріалів об'єктів, а також значний обсяг багажу, що проходить перевірку.

Традиційний підхід до аналізу рентгенівських зображень передбачає участь операторів служби безпеки, які здійснюють візуальну перевірку сканів багажу. Однак людський фактор суттєво впливає на якість контролю, оскільки тривала робота з великим потоком однотипних зображень призводить до втоми операторів та зниження точності виявлення небезпечних предметів. Крім того, збільшення інтенсивності пасажиропотоку потребує автоматизації процесів аналізу багажу та підвищення швидкості прийняття рішень.

У сучасних умовах перспективним напрямком вирішення цієї проблеми є використання методів комп'ютерного зору та глибокого навчання. Застосування згорткових нейронних мереж (CNN), моделей object detection та сучасних архітектур глибокого навчання дозволяє автоматизувати процес аналізу рентгенівських зображень і підвищити ефективність виявлення заборонених предметів. Останні дослідження у сфері авіаційної безпеки демонструють високу ефективність таких підходів при обробці складних рентгенівських сцен із великою кількістю об'єктів та значним рівнем накладання предметів.

Особливої актуальності питання автоматизації систем контролю багажу набуває в контексті майбутнього відновлення та модернізації авіаційної інфраструктури України. Використання сучасних інтелектуальних систем аналізу рентгенівських зображень може стати важливою складовою підвищення рівня авіаційної безпеки, оптимізації роботи служб контролю та зменшення впливу людського фактору.

Таким чином, задача розробки методів класифікації заборонених предметів у багажі авіапасажирів є актуальною та має важливе практичне значення.

Об'єктом дослідження є процес автоматизованого аналізу рентгенівських 2D зображень багажу авіапасажирів.

Предметом дослідження є методи класифікації заборонених предметів на рентгенівських зображеннях із використанням технологій комп'ютерного зору та глибокого навчання.

Метою дипломної роботи є дослідження методів класифікації заборонених предметів у багажі авіапасажирів на основі методів комп'ютерного зору та нейронних мереж.

Для досягнення поставленої мети необхідно виконати такі завдання:

- проаналізувати сучасні методи контролю багажу авіапасажирів;
- дослідити особливості рентгенівських зображень багажу;
- виконати аналіз сучасних методів комп'ютерного зору та глибокого навчання для задач класифікації зображень;
- дослідити існуючі підходи до виявлення заборонених предметів у багажі;
- розробити метод класифікації заборонених предметів;
- провести експериментальне дослідження ефективності запропонованого підходу;
- виконати оцінювання отриманих результатів.

У роботі використано методи комп'ютерного зору, глибокого навчання, аналізу зображень, обробки даних та методи оцінювання якості класифікації.

Практичне значення отриманих результатів полягає у можливості використання розроблених методів у системах автоматизованого контролю

багажу для підвищення ефективності роботи служб авіаційної безпеки та зменшення часу аналізу рентгенівських зображень.<sup>1</sup>

---

<sup>1</sup>Тут і нижче використано такий інструмент штучного інтелекту як чат-бот з генеративним штучним інтелектом ChatGPT виключно для корегування та редагування тексту, створеного автором цієї дипломної роботи, на основі автоматизованої перевірки граматики, структури та стилю, що відповідає Політиці використання штучного інтелекту для академічної діяльності в КПІ ім. Ігоря Сікорського (протокол №11 Вченої ради КПІ ім. Ігоря Сікорського від 11 грудня 2023 р.).



## **РОЗДІЛ 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА СУЧАСНИХ МЕТОДІВ ДЕТЕКЦІЇ ОБ'ЄКТІВ НА РЕНТГЕНІВСЬКИХ ЗОБРАЖЕННЯХ**

### **1.1 Нормативно-правове регулювання перевезення небезпечних предметів в авіації**

Забезпечення авіаційної безпеки є одним із найважливіших завдань сучасної цивільної авіації. Щороку кількість авіапасажирів зростає, розширюється мережа міжнародних перевезень, а разом із цим підвищуються вимоги до контролю безпеки в аеропортах. Одним із ключових елементів такого контролю є перевірка багажу та ручної поклажі, метою якої є виявлення предметів, що можуть становити загрозу для пасажирів, екіпажу або повітряного судна.

Питання перевезення небезпечних предметів регулюються як міжнародними, так і національними нормативними документами. Визначальну роль у формуванні вимог та стандартів авіаційної безпеки відіграють Міжнародна організація цивільної авіації (ICAO), Міжнародна асоціація повітряного транспорту (IATA) та Агентство авіаційної безпеки Європейського Союзу (EASA). Ці організації розробляють правила, рекомендації та методичні документи, які визначають порядок перевезення небезпечних речовин і проведення процедур безпеки в аеропортах. Їхні вимоги використовуються як основа для організації контролю багажу та забезпечення належного рівня безпеки авіаперевезень.

Одним із основних нормативних документів Європейського Союзу у сфері авіаційної безпеки є регламент Європейської Комісії «Air security: Commission draws up EU-wide list of prohibited articles on passenger aircraft», у якому визначено перелік предметів, заборонених до перевезення у ручній поклажі та на борту повітряного судна. Основною метою даного регламенту є забезпечення єдиних правил авіаційної безпеки для всіх аеропортів

Європейського Союзу та надання пасажиром чіткої інформації щодо предметів, які можуть бути конфісковані під час проходження контролю безпеки.

Відповідно до зазначеного документа, до категорії заборонених предметів належать вогнепальна зброя та пристрої, здатні здійснювати постріл або завдавати тілесних ушкоджень. До них відносяться:

- пістолети;
- револьвери;
- рушниці;
- сигнальні пістолети;
- пневматична зброя;
- арбалети;
- гарпуни;
- електрошокові пристрої;
- імітації вогнепальної зброї.

Окрему категорію становлять колючі та ріжучі предмети, зокрема:

- ножі;
- сокири;
- мачете;
- леза;
- шаблі;
- скальпелі;
- ножиці з довжиною леза понад 6 см;
- пилки;
- молотки;
- викрутки;
- свердла;
- інші будівельні інструменти, які можуть бути використані як зброя.

До тупих предметів, здатних завдати фізичних ушкоджень, належать:

- бейсбольні біти;

- кийки;
- хокейні ключки;
- більярдні кийи;
- риболовні вудки;
- весла;
- спорядження для бойових мистецтв.

Особливу небезпеку становлять вибухові та легкозаймисті речовини, серед яких:

- боєприпаси;
- детонатори;
- гранати;
- вибухові речовини;
- феєрверки;
- газові балони;
- бензин;
- дизельне паливо;
- розчинники;
- спиртовмісні рідини високої концентрації.

Також забороняється перевезення токсичних та хімічно небезпечних речовин, а саме:

- кислот;
- лугів;
- отруйних речовин;
- інфекційних матеріалів;
- радіоактивних елементів;
- газових балончиків із подразнювальними речовинами;
- інших небезпечних хімічних сполук.

Окремі обмеження встановлені для перевезення речовин у зареєстрованому багажі. До категорії заборонених належать вибухові речовини,

легкозаймисті гази, токсичні матеріали, окисники, а також радіоактивні та корозійні речовини. Їх транспортування у вантажному відсіку літака не допускається через ризик виникнення пожеж, вибухів або пошкодження повітряного судна під час польоту .[2]

Контроль багажу ускладнюється великою кількістю заборонених предметів та їхньою різноманітністю. Деякі з них мають схожі контури або внутрішню структуру з дозволеними речами, а в окремих випадках можуть бути частково приховані серед інших об'єктів у валізі. Через це аналіз рентгенівських зображень лише силами оператора не завжди забезпечує необхідну ефективність, особливо в аеропортах із великим пасажиропотоком та високим навантаженням на служби контролю.

З огляду на це дедалі більшого поширення набувають системи автоматизованого аналізу рентгенівських зображень, засновані на методах комп'ютерного зору та штучного інтелекту. Використання нейронних мереж і алгоритмів глибокого навчання дає змогу точніше виявляти потенційно небезпечні предмети, зменшувати вплив людського фактору та прискорювати процедуру перевірки багажу.

## **1.2 Принцип роботи рентгенівських систем контролю багажу**

### ***1.2.1 Будова рентгенівського сканера***

Рентгенівський апарат є пристроєм для генерування рентгенівського випромінювання, яке належить до електромагнітного спектра та здатне проникати крізь різні матеріали. Саме ця особливість зробила рентгенівські технології незамінними як у медицині, так і в системах безпеки. В аеропортах вони використовуються для перевірки багажу та виявлення предметів, перевезення яких заборонене правилами авіаційної безпеки.

Робота рентгенівського апарата ґрунтується на генерації випромінювання у вакуумній трубці, що містить катод і анод. Під час нагрівання катод випускає електрони, які під дією високої напруги прискорюються та спрямовуються до анода. У момент їх зіткнення утворюються рентгенівські фотони. Характеристики отриманого випромінювання, зокрема його інтенсивність та енергія, залежать від величини прикладеної напруги.

Далі наведені основні типи взаємодії рентгенівського випромінювання з матеріалами.

1. Фотоелектричний ефект, коли фотон повністю поглинається електроном внутрішньої оболонки атома, що призводить до його вивільнення. Цей ефект переважає при нижчих енергіях та в речовинах з високим атомним номером.
2. Комптонівське розсіювання, коли фотон взаємодіє з вільним або слабко зв'язаним електроном, змінюючи напрямок і частково втрачаючи енергію. Цей ефект відіграє ключову роль у формуванні зображення під час сканування.

Будова сучасного рентгенівського апарату зображена на рисунку 2.1. До його конструкції входять.

- Рентгенівська трубка, що містить катод і анод.
- Високовольтний генератор, який створює необхідну напругу.
- Система охолодження, що запобігає перегріву анода.
- Пульт керування, за допомогою якого регулюють параметри роботи.
- Алюмінієвий фільтр, який відсікає небажане випромінювання.
- Коліматор, який формує спрямований пучок променів.
- Рентгенівська плівка або цифровий детектор для фіксації зображення.

Сучасні рентгенівські сканери, що використовуються для контролю багажу, здатні просвічувати сталеві перешкоди товщиною приблизно від 32 до 43 мм залежно від моделі та технічних характеристик обладнання. Така проникаюча здатність дає змогу аналізувати вміст валіз, сумок і поштових

відправлень, виявляючи як органічні, так і неорганічні матеріали, приховані всередині.

Приклади глибини проникнення рентгенівських сканерів.

- ZKTeco ZKX6040: проникає через сталь товщиною до 35 мм.
- MatriXcore MTC-XR53PRO: здатен проникати через сталь товщиною до 39 мм.
- ARSENAL-XR6550: має проникнення через сталь товщиною до 43 мм.
- Labtron LXBS-D10: проникає через сталь товщиною до 38 мм.
- SA6550: здатен проникати через сталь товщиною до 34 мм.

Ефективність рентгенівського сканування залежить від властивостей матеріалів, через які проходять промені. Найбільші труднощі створюють щільні метали, зокрема свинець, який може майже повністю поглинати рентгенівське випромінювання. У таких випадках предмети, розташовані за металевим екраном, стають частково або повністю невидимими на зображенні.

На якість сканування також впливає товщина об'єкта. Чим товстіший предмет, тим складніше променям пройти крізь нього та сформувати чітке зображення. Водночас органічні матеріали, такі як папір, тканина чи пластик, зазвичай пропускають випромінювання краще, ніж метали та інші щільні речовини.

Додаткові труднощі виникають через розташування предметів усередині багажу. Якщо об'єкти перекривають один одного або знаходяться під незручними кутами, їхні контури можуть накладатися, що ускладнює аналіз зображення та підвищує ймовірність пропуску небезпечних предметів.

Саме на різниці у поглинанні рентгенівських променів різними матеріалами ґрунтується робота багажних сканерів. Отримане зображення дозволяє оператору оцінити вміст поклажі та виявити предмети, які можуть становити потенційну загрозу.

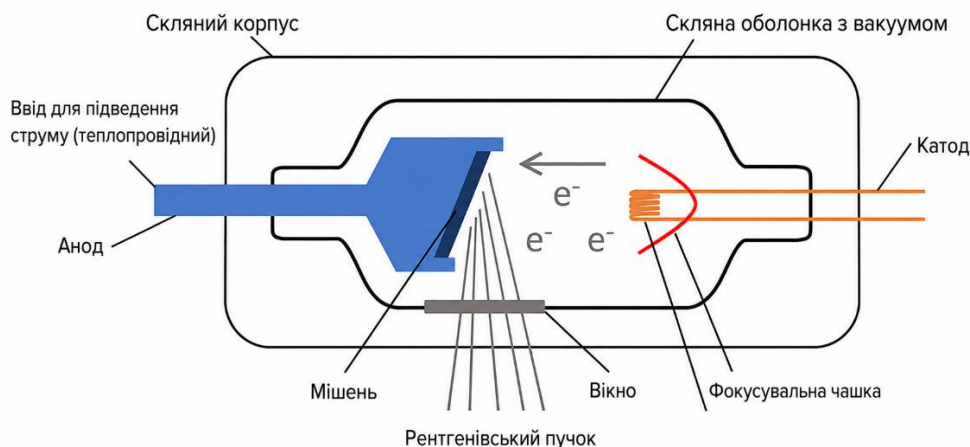


**Рисунок 2.1** – Структура рентгенівського апарату [3]

### 1.2.2 Принцип формування X-ray image

Рентгенівські системи контролю багажу є одним із основних технічних засобів забезпечення авіаційної безпеки в аеропортах. Їх основне призначення полягає у виявленні заборонених та потенційно небезпечних предметів у ручній поклажі та зареєстрованому багажі пасажирів без необхідності фізичного відкриття сумок або валіз [4].

Основними компонентами рентгенівської системи контролю багажу є:

- рентгенівська трубка;
- конвеєрна стрічка;
- система детекторів;
- модуль обробки зображень;
- робоча станція оператора.

Під час сканування багаж переміщується через тунель сканера за допомогою конвеєрної стрічки. Рентгенівська трубка генерує рентгенівське випромінювання, яке проходить крізь об'єкти всередині багажу. Після проходження через предмети рентгенівські промені потрапляють на систему

детекторів, яка реєструє рівень ослаблення випромінювання та формує цифрове зображення [4].

Принцип роботи рентгенівських систем базується на явищі ослаблення рентгенівського випромінювання. Різні матеріали по-різному поглинають рентгенівські промені залежно від:

- щільності матеріалу;
- товщини об'єкта;
- атомного номера речовини;
- енергії рентгенівського випромінювання.

Чим вища щільність матеріалу, тим більше рентгенівського випромінювання він поглинає. Саме тому металеві предмети добре помітні на рентгенівських зображеннях і відрізняються від менш щільних матеріалів. Металеві предмети, такі як зброя або ножі, зазвичай добре помітні через високу щільність матеріалу. Органічні матеріали поглинають випромінювання слабше та відображаються інакше. Наприклад, на рисунку 2.2 видно, як металеві предмети такі як: ножі та пістолет підсвічені на рентгенівському скані контрастніше через свою щільність матеріалу [5].

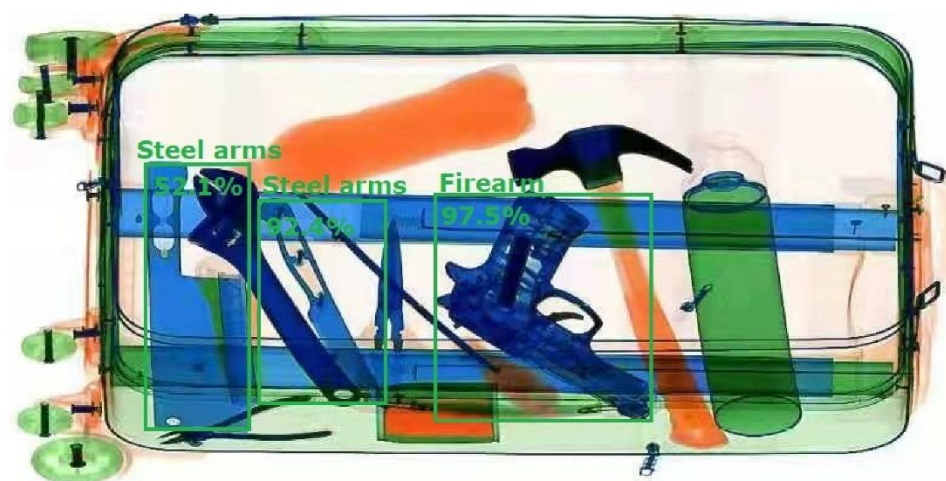


**Рисунок 2.2** – Рентгенівський знімок багажу з вмістом об'єктів з щільного матеріалу [9]

Навіть сучасні рентгенівські системи не здатні повністю усунути складнощі під час аналізу багажу. Найчастіше проблеми виникають через перекриття предметів, коли один об'єкт частково або повністю закриває інший на зображенні. Крім цього, на точність аналізу впливають:

- низька текстурованість рентгенівських зображень;
- велика кількість об'єктів у багажі;
- складний фон;
- шум та артефакти зображення;
- схожість деяких небезпечних предметів із дозволеними об'єктами.

Найскладніше виявляти дрібні предмети та об'єкти зі схожими характеристиками поглинання випромінювання. Через це навіть досвідчені оператори не завжди можуть однозначно визначити вміст багажу за рентгенівським зображенням.

Саме тому в сучасних системах контролю все ширше використовуються методи комп'ютерного зору та штучного інтелекту. Нейронні мережі допомагають автоматично знаходити потенційно небезпечні предмети, підвищуючи точність перевірки та зменшуючи залежність від людського фактору.

### *1.2.3 Двоенергетичне рентгенівське сканування*

У сучасних аеропортах широко використовуються системи двоенергетичного рентгенівського сканування (dual-energy), які застосовують два різні рівні енергії рентгенівського випромінювання. Такий підхід дозволяє не лише оцінювати щільність об'єкта, але й визначати його приблизний матеріальний склад на основі ефективного атомного номера речовини [4, 6].

Однією з переваг технології dual-energy є можливість автоматично розрізняти різні типи матеріалів. Щоб спростити аналіз зображень, система

позначає їх різними кольорами. Зазвичай у рентгенівських сканерах використовується така схема кольорового відображення [7]:

- помаранчевий колір — органічні матеріали;
- зелений колір — змішані або проміжні матеріали;
- синій або темно-синій колір — метали та матеріали з високим атомним номером.

Як показано на рис. 2.3, металеві об'єкти мають темно-синій колір через високу щільність матеріалу та значний рівень поглинання рентгенівського випромінювання. Органічні матеріали відображаються помаранчевим кольором, тоді як змішані матеріали мають зелений відтінок. Подібне кольорове кодування дозволяє оператору швидше ідентифікувати потенційно небезпечні об'єкти та аналізувати складні сцени у багажі.

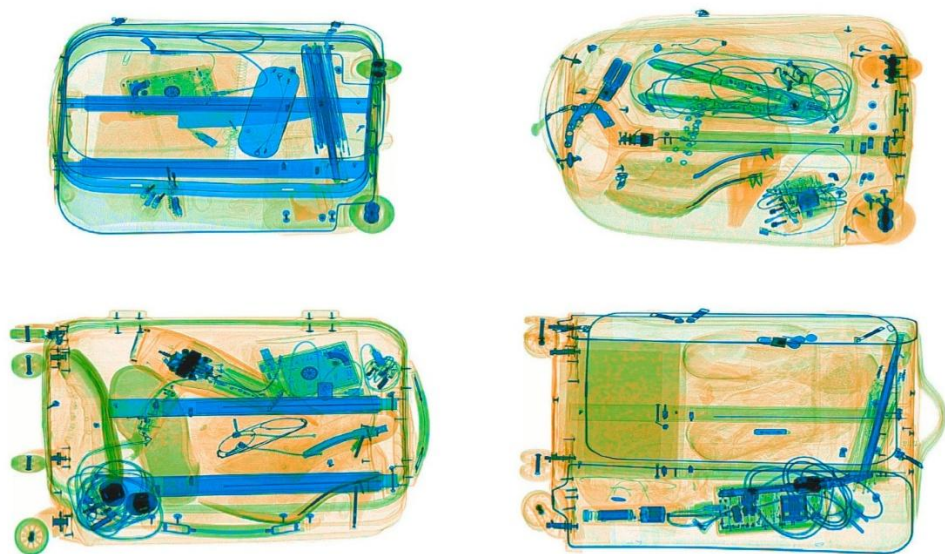


**Рисунок 2.3** - Приклад кольорового відображення матеріалів у dual-energy рентгенівській системі контролю багажу [8]

### *1.2.4 Проблеми аналізу рентгенівських знімків*

Навіть сучасні рентгенівські системи не завжди забезпечують простий аналіз багажу. Основною проблемою залишається перекриття предметів, коли один об'єкт закриває інший на зображенні. Додаткові складнощі пов'язані з:

- низька текстурованість рентгенівських зображень;
- велика кількість об'єктів у багажі;
- складний фон;
- шум та артефакти зображення [10];
- схожість деяких небезпечних предметів із дозволеними об'єктами.

Особливо складним є виявлення невеликих предметів або об'єктів, що мають схожі характеристики поглинання рентгенівського випромінювання. У деяких випадках навіть оператору складно однозначно визначити тип предмета за рентгенівським зображенням [11].

Для аналізу рентгенівських зображень усе частіше використовуються методи комп'ютерного зору та глибокого навчання. Нейронні мережі допомагають автоматично знаходити небезпечні предмети, підвищуючи точність перевірки багажу та зменшуючи навантаження на операторів.

## **1.3 Методи комп'ютерного зору для аналізу рентгенівських зображень**

### *1.3.1 Класичні методи комп'ютерного зору*

До широкого впровадження глибокого навчання для аналізу рентгенівських зображень використовувалися переважно класичні методи комп'ютерного зору. Вони ґрунтувалися на виділенні характерних ознак

зображення, таких як контури, текстури та геометричні особливості об'єктів, після чого виконувалася їх класифікація.

Одним із найпоширеніших підходів було виявлення контурів за допомогою алгоритмів Sobel, Canny та Laplacian. Такі методи дозволяли визначати межі об'єктів і аналізувати їхню форму. Проте для рентгенівських зображень їхня ефективність обмежена через низький контраст і часте перекриття предметів у багажі.

Іншим поширеним методом була сегментація зображень. Сегментація дозволяє розділити зображення на окремі області або об'єкти для подальшого аналізу. Для цього використовувалися методи порогової обробки, region growing та watershed segmentation. У системах аналізу багажу сегментація застосовувалася для виділення потенційно небезпечних об'єктів із загального фону рентгенівського зображення. Однак через складну структуру рентгенівських зображень та наявність накладання об'єктів результати сегментації часто були неточними.

Для виділення характерних ознак зображення використовувалися дескриптори HOG (Histogram of Oriented Gradients) та SIFT (Scale-Invariant Feature Transform). Зокрема, HOG описує форму об'єктів на основі напрямків градієнтів і добре працює для виявлення предметів із чіткими контурами, таких як зброя або ножі.

Даний підхід активно застосовувався для задач виявлення зброї, ножів та інших предметів із чіткими контурами.

Метод SIFT використовується для пошуку ключових точок та локальних ознак, які залишаються стійкими до масштабування, повороту та часткових змін зображення. Це дозволяло виконувати порівняння об'єктів навіть за умов часткової деформації або зміни положення предмета. (researchgate.net)

Незважаючи на поширеність класичних методів комп'ютерного зору, їхня ефективність для аналізу рентгенівських зображень залишається обмеженою. До основних недоліків належать:

- низька ефективність при складних сценах;

- чутливість до шуму та зміни контрасту;
- складність роботи з накладанням об'єктів;
- необхідність ручного налаштування ознак;
- обмежена здатність до узагальнення;
- низька точність при виявленні прихованих предметів.

Класичні алгоритми потребують ручного налаштування ознак, що суттєво обмежує їхню ефективність у складних рентгенівських сценах. Натомість сучасні нейронні мережі навчаються знаходити потрібні ознаки автоматично, тому значно краще справляються із завданням виявлення заборонених предметів.

### *1.3.2 Методи глибокого навчання*

У сучасних системах автоматизованого аналізу рентгенівських зображень дедалі ширше використовуються методи глибокого навчання (deep learning). Найбільш поширеними серед них є згорткові нейронні мережі (Convolutional Neural Networks, CNN), які демонструють високу ефективність у задачах класифікації та виявлення об'єктів на зображеннях [12].

Основною перевагою CNN є здатність автоматично виконувати виділення ознак (feature extraction) без необхідності ручного проєктування дескрипторів зображення. На відміну від класичних методів комп'ютерного зору, нейронні мережі самостійно навчаються виявляти характерні ознаки об'єктів під час процесу навчання моделі [13].

Базовим елементом згорткових нейронних мереж є операція згортки (convolution), яка дозволяє аналізувати локальні області зображення та формувати карти ознак (feature maps). У процесі проходження через декілька шарів мережі модель поступово навчається розпізнавати складні структури та форми об'єктів [14].

Однією з основних проблем під час навчання моделей для аналізу X-ray зображень є обмежена кількість розмічених даних. Для вирішення цієї проблеми широко використовується підхід transfer learning, який передбачає використання попередньо натренованих моделей, навчання яких виконувалося на великих наборах зображень, з подальшою адаптацією до задачі аналізу рентгенівських сканів багажу [12].

У задачах аналізу рентгенівських зображень найчастіше використовуються архітектури ResNet, VGG, DenseNet та EfficientNet. Дані моделі демонструють високу ефективність при виявленні заборонених предметів у багажі пасажирів навіть за умов складних сцен, накладання об'єктів та низької якості зображення [15].

У сучасних системах аналізу рентгенівських зображень широко застосовуються моделі детекції об'єктів, зокрема Faster R-CNN, YOLO та RetinaNet. На відміну від звичайної класифікації, вони не лише визначають тип об'єкта, а й знаходять його розташування на зображенні. Це робить такі моделі особливо корисними для автоматизованого контролю багажу.

### *1.3.3 Методи object detection*

Одним із ключових напрямків сучасного комп'ютерного зору є методи виявлення об'єктів (object detection), які дозволяють не лише визначати тип об'єкта на зображенні, але й локалізувати його положення. На відміну від задачі класифікації, де модель визначає лише наявність певного об'єкта на зображенні, object detection додатково виконує пошук координат об'єкта за допомогою обмежувальних рамок (bounding boxes). Подібний підхід є особливо важливим для аналізу рентгенівських зображень багажу, де необхідно не тільки виявити заборонений предмет, але й визначити його місце розташування [14].

Одними з перших ефективних моделей object detection стали архітектури сімейства R-CNN (Region-based Convolutional Neural Networks). Дані методи виконують пошук потенційних областей з об'єктами, після чого застосовують згорткову нейронну мережу для класифікації знайдених регіонів. Подальшим розвитком цієї архітектури стала модель Faster R-CNN, у якій використовується спеціальна Region Proposal Network (RPN) для автоматичного генерування областей пошуку, що дозволило значно підвищити швидкість роботи системи [16].

Для задач реального часу широко застосовуються моделі сімейства YOLO (You Only Look Once). На відміну від R-CNN, алгоритми YOLO виконують локалізацію та класифікацію об'єктів за один прохід нейронної мережі, що забезпечує високу швидкість обробки зображень. Завдяки цьому моделі YOLO активно використовуються в системах автоматизованого контролю багажу та відеоспостереження [17].

Ще одним популярним методом є SSD (Single Shot MultiBox Detector), який також належить до однопрохідних алгоритмів object detection. SSD поєднує високу швидкість роботи та достатню точність виявлення об'єктів, що робить його ефективним для задач аналізу рентгенівських зображень [18].

Поряд із CNN-моделями останніми роками активно розвиваються підходи на основі архітектури Transformer. Одним із найвідоміших представників цього напрямку є DETR (Detection Transformer), який поєднує згорткові нейронні мережі та механізм attention для виявлення об'єктів. Такий підхід показав хороші результати на складних зображеннях із великою кількістю об'єктів [19].

У задачах аналізу рентгенівських зображень багажу методи object detection використовуються для автоматичного виявлення зброї, ножів, вибухових речовин та інших небезпечних предметів. Основними перевагами таких підходів є можливість автоматизації процесу контролю, зменшення навантаження на операторів служби безпеки та підвищення точності аналізу складних X-ray зображень [11].

## 1.4 Особливості виявлення заборонених предметів у багажі

Виявлення заборонених предметів на рентгенівських зображеннях багажу є складною задачею комп'ютерного зору через специфічні особливості X-ray imagery. На відміну від звичайних фотографічних зображень, рентгенівські скани містять велику кількість накладених об'єктів, обмежену текстурну інформацію та складні фонові структури. Це значно ускладнює процес автоматичного аналізу та класифікації небезпечних предметів [11].

Складність аналізу рентгенівських зображень значною мірою пов'язана з перекриттям предметів (overlapping objects). Під час сканування об'єкти накладаються один на одного на двовимірному зображенні, що ускладнює їх виявлення та класифікацію.

Додатковою складністю є наявність прихованих або частково закритих предметів (hidden objects та occlusion). У деяких випадках заборонений предмет може бути розташований між іншими речами або закритий щільними матеріалами, через що на рентгенівському зображенні видно лише окремі його фрагменти [10].

Важливою проблемою також є виявлення малих об'єктів (small object detection). Невеликі предмети, такі як леза, детонатори або компактні вибухові пристрої, можуть займати незначну область на зображенні та втрачатися серед інших предметів у багажі. Через це системи автоматизованого аналізу повинні мати високу точність локалізації навіть для малорозмірних об'єктів [20].

Ще однією особливістю задачі є дисбаланс класів (class imbalance). У реальних умовах переважна більшість рентгенівських зображень не містить небезпечних предметів, тоді як кількість сканів із забороненими об'єктами є значно меншою. Це створює труднощі під час навчання моделей глибокого навчання та може призводити до зниження точності виявлення небезпечних предметів [11].

Для систем контролю багажу також важливими є вимоги реального часу (real-time constraints). В аеропортах щоденно обробляється велика кількість багажу, тому системи автоматизованого аналізу повинні швидко виконувати обробку рентгенівських зображень без суттєвого зниження точності. У зв'язку з цим сучасні моделі object detection мають поєднувати високу швидкість роботи та ефективність виявлення об'єктів [17].

На якість перевірки також впливає втома операторів (operator fatigue). Постійний аналіз великої кількості схожих рентгенівських зображень з часом знижує уважність і збільшує ймовірність пропуску небезпечних предметів. Саме тому автоматизовані системи дедалі частіше використовуються як допоміжний інструмент під час контролю багажу.

У задачах автоматичного аналізу рентгенівських зображень особливо критичними є помилки типу false positive та false negative. Помилка false positive виникає у випадку, коли система помилково визначає безпечний предмет як небезпечний, що може призводити до затримок перевірки багажу. Натомість false negative є більш небезпечним випадком, коли система не виявляє реальний заборонений предмет, що створює потенційну загрозу безпеці польоту [10].

Особливості рентгенівських зображень, наявність прихованих об'єктів і вимоги до швидкості роботи роблять задачу виявлення заборонених предметів досить складною. Ефективно розв'язувати її сьогодні дозволяють методи комп'ютерного зору та глибокого навчання, які стали основою сучасних систем контролю багажу.

## 1.5 Аналіз сучасних досліджень у сфері автоматизованого контролю багажу

Останніми роками інтерес до автоматизованого аналізу рентгенівських зображень суттєво зріс. Це пов'язано насамперед зі збільшенням кількості авіапасажирів та необхідністю прискорення процедур контролю безпеки в аеропортах. У таких умовах оператори змушені аналізувати значну кількість рентгенівських знімків, що підвищує ризик помилок і пропуску небезпечних предметів. Саме тому значна кількість сучасних наукових робіт присвячена створенню систем, здатних автоматично знаходити підозрілі об'єкти на рентгенівських зображеннях за допомогою методів комп'ютерного зору та глибокого навчання [25, 26].

На початкових етапах розвитку таких систем переважно використовувалися класичні алгоритми аналізу зображень. Вони базувалися на пошуку контурів, аналізі текстур та ручному виділенні ознак об'єктів. Проте для рентгенівських зображень подібні підходи виявилися недостатньо ефективними. Причиною стали накладання предметів один на одного, низька контрастність окремих ділянок та відсутність чітко виражених текстур. Подальший розвиток глибокого навчання дозволив перейти до моделей, які самостійно навчаються знаходити необхідні ознаки без ручного налаштування [27].

Однією з помітних робіт у цьому напрямі стало дослідження *Toward Automatic Threat Recognition for Airport X-ray Baggage Screening with Deep Convolutional Object Detection*, виконане за участю TSA та Rapiscan Systems. У межах роботи було сформовано масштабний набір рентгенівських зображень багажу та досліджено можливість використання згорткових нейронних мереж для автоматичного виявлення зброї, ножів, тупих предметів і рідин. Отримані результати показали, що моделі глибокого навчання можуть суттєво підвищити ефективність контролю безпеки порівняно з традиційними підходами [28].

Серед моделей, які сьогодні найчастіше застосовуються для пошуку небезпечних предметів, можна виділити Faster R-CNN, YOLO, SSD, RetinaNet та архітектури на основі Transformer. Особливої популярності набули моделі сімейства YOLO завдяки поєднанню достатньо високої точності та швидкої обробки зображень. Так, у дослідженні Deep Learning for Detecting Dangerous Objects in X-rays of Luggage модифікована версія YOLOv5 забезпечила значення mean average recall на рівні 87 % для п'яти категорій заборонених предметів [29].

Окремий напрям сучасних досліджень пов'язаний із використанням трансформерних архітектур. На відміну від традиційних згорткових мереж, вони здатні враховувати взаємозв'язки між різними ділянками зображення та ефективніше працювати зі складними сценами. У роботі Illicit Object Detection in X-ray Imaging Using Deep Learning Techniques: A Comparative Evaluation було виконано порівняння декількох сучасних моделей на наборах даних OPIXray, SIXray, HiXray та PIDray. Результати показали, що Transformer-підходи демонструють хорошу якість детекції у випадках значного перекриття предметів та складної структури багажу [30].

Для практичного використання в аеропортах важливою є не лише точність виявлення, але й швидкість роботи алгоритмів. Тому значна частина сучасних досліджень спрямована на створення моделей, які можуть виконувати аналіз рентгенівських зображень у режимі реального часу без суттєвого зниження якості детекції.

Також значна увага приділяється проблемам виявлення перекритих об'єктів, прихованих предметів та дисбалансу класів у наборах даних. Дослідники відзначають, що рентгенівські зображення значно складніші для автоматичного аналізу, ніж звичайні фотографії, оскільки містять велику кількість накладених об'єктів, мають низьку контрастність та обмежену текстурну інформацію [32]. Саме ці особливості залишаються одними з головних викликів під час розроблення сучасних систем автоматизованого контролю багажу.

Узагальнення основних напрямків сучасних досліджень наведено в таблиці 1.1.

**Таблиця 1.1** – Сучасні методи аналізу X-ray зображень багажу

| <i>Напрямок дослідження</i> | <i>Основні методи</i>            | <i>Особливості</i>                            |
|-----------------------------|----------------------------------|-----------------------------------------------|
| Класичний computer vision   | Edge detection, texture analysis | низька ефективність для складних X-ray scenes |
| CNN-based methods           | ResNet, VGG, DenseNET            | автоматичне виділення ознак                   |
| Object detection            | YOLO, Faster R-CNN, SSD          | локалізація та класифікація об'єктів          |
| Transformer-based models    | DETR, hybrid CNN-transformers    | ефективна робота зі складними сценами         |
| Real-time detection         | Lightweight YOLO models          | висока швидкість аналізу                      |

Розвиток глибокого навчання суттєво змінив підходи до аналізу рентгенівських зображень. Сучасні моделі, зокрема YOLO, Faster R-CNN та DETR, забезпечують значно кращі результати порівняно з класичними методами комп'ютерного зору. Проте складна структура багажу, перекриття предметів і різноманітність заборонених об'єктів досі створюють труднощі для автоматизованих систем.

## **1.6 Аналіз наборів даних для дослідження рентгенівських зображень**

Якість роботи моделей комп'ютерного зору значною мірою залежить від набору даних, на якому вони навчаються. Для задач аналізу рентгенівських зображень це особливо важливо, оскільки такі зображення суттєво відрізняються від звичайних фотографій. Предмети в багажі часто перекривають один одного, можуть бути частково прихованими, а деякі

категорії об'єктів трапляються значно рідше за інші. Через це для навчання та тестування моделей використовуються спеціалізовані датасети, створені саме для задач контролю безпеки [33].

Найчастіше у наукових роботах використовується набір даних SIXray, представлений у дослідженні SIXray: A Large-scale Security Inspection X-ray Benchmark for Prohibited Item Discovery in Overlapping Images. До його складу входить понад мільйон рентгенівських зображень, отриманих у реальних умовах роботи пунктів контролю безпеки. У наборі представлені шість категорій заборонених предметів: зброя, ножі, гайкові ключі, плоскогубці, ножиці та молотки. Особливістю SIXray є значна кількість перекритих об'єктів і виражений дисбаланс між класами, що робить його одним із найскладніших наборів даних для задач автоматичного аналізу багажу. Для частини зображень також доступна розмітка у вигляді обмежувальних рамок, необхідна для навчання моделей детекції об'єктів (рис.2.4) [33].

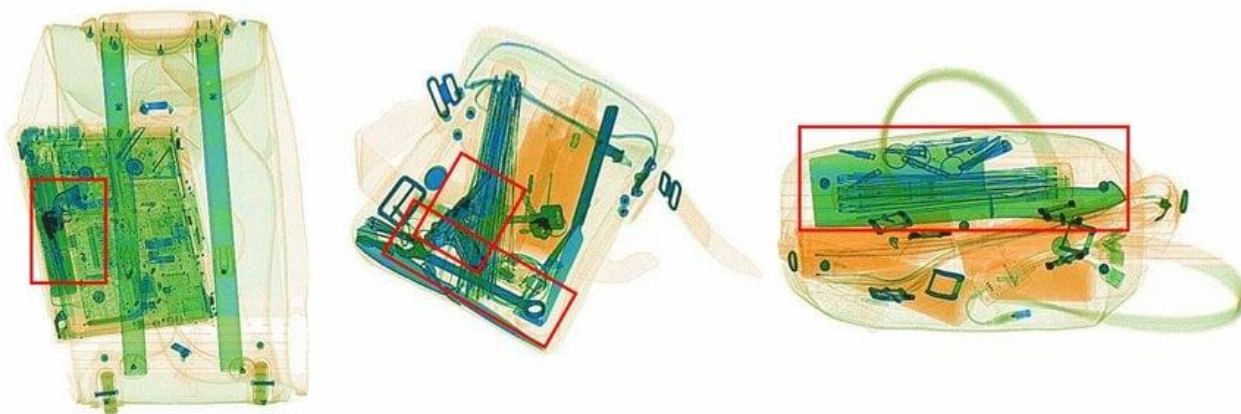


**Рисунок 2.4** – Приклад X-ray знімку з датасету SIXray [33]

Для дослідження більш складних сценаріїв використовується датасет PIDray, описаний у роботі PIDray: A Large-Scale X-ray Benchmark for Real-World Prohibited Item Detection. Він містить понад 120 тисяч рентгенівських зображень та охоплює дванадцять категорій заборонених предметів. Особливістю цього набору є наявність прикладів, де небезпечні об'єкти

навмисно приховані серед інших речей. Крім обмежувальних рамок, PIDray містить сегментаційну розмітку та багатоміткові анотації, а також поділ на рівні складності Easy, Hard і Hidden [34].

Ще одним джерелом даних для досліджень є HiXray. Набір включає понад сто тисяч рентгенівських зображень для восьми категорій заборонених предметів. Його формування здійснювалося на основі реальних даних аеропортів, а розмітка виконувалася за участю фахівців служб безпеки. Завдяки цьому HiXray часто використовується для оцінювання моделей у наближених до реальних умовах експлуатації [35].

Для задач, пов'язаних із сильним перекриттям об'єктів, розроблено датасет OPIXray. Основна увага під час його створення приділялася саме випадкам, коли небезпечний предмет частково прихований іншими речами. Такі сцени є одними з найскладніших для автоматичного аналізу, тому OPIXray часто використовується для перевірки стійкості моделей до оклюзій [36].

Серед більш ранніх відкритих наборів даних варто відзначити GDXray. Хоча його обсяг значно менший порівняно із сучасними датасетами, він і досі використовується під час тестування алгоритмів комп'ютерного зору та задач перенесення навчання. Саме GDXray став одним із перших відкритих джерел рентгенівських зображень для дослідників у цій галузі [37].

У таблиці 1.2 наведено порівняння основних характеристик розглянутих датасетів.

Порівняння розглянутих наборів даних показує, що більшість із них стикається з подібними труднощами. До них належать нерівномірний розподіл класів, велика кількість перекритих об'єктів, складність виявлення малих предметів і обмежений обсяг детально анотованих даних. Саме тому останніми роками значна увага приділяється створенню нових наборів даних та використанню синтетичних рентгенівських зображень для розширення навчальних вибірок і покращення якості моделей глибокого навчання [38].

**Таблиця 1.2** – Основні набори даних для аналізу рентгенівських зображень багажу

| <i>Dataset</i> | <i>Кількість зображень</i> | <i>Кількість класів</i> | <i>Особливості</i>                      |
|----------------|----------------------------|-------------------------|-----------------------------------------|
| SIXray         | 1 059 231                  | 6                       | overlapping objects, class imbalance    |
| PIDray         | 124 486                    | 12                      | hidden objects, multi-task annotations  |
| HiXray         | 102 928                    | 8                       | high-quality airport imagery            |
| OPIXray        | 8 885                      | 5                       | heavily occluded prohibited items       |
| GDXray         | 19 407                     | Різні категорії         | один із перших відкритих X-ray datasets |

### 1.7 Аналіз сучасних моделей виявлення заборонених предметів

Більшість сучасних систем автоматичного контролю багажу використовує моделі детекції об'єктів. На відміну від звичайної класифікації, такі моделі не лише визначають тип знайденого предмета, а й указують його розташування на зображенні. Для рентгенівських сканів це має особливе значення, оскільки в одному багажі може міститися багато різних предметів, частина з яких перекриває один одного або частково прихована [39].

Сучасні алгоритми детекції умовно поділяють на три групи: двоетапні моделі (two-stage detectors), одноетапні моделі (one-stage detectors) та архітектури на основі Transformer. Вони відрізняються насамперед підходом до пошуку об'єктів і співвідношенням між швидкістю роботи та точністю результатів [40].

Серед двоетапних моделей найбільш відомою є Faster R-CNN. Її робота складається з двох послідовних етапів: спочатку визначаються ділянки, де потенційно можуть знаходитися об'єкти, а потім виконується їх класифікація. Завдяки такому підходу модель забезпечує високу якість локалізації навіть у складних сценах, проте потребує більше часу на обробку зображень порівняно з сучасними швидкісними алгоритмами [41].

У задачах аналізу рентгенівських зображень Faster R-CNN демонструє хороші результати під час виявлення зброї, ножів та інших небезпечних предметів. У роботі *Evaluation of a Dual Convolutional Neural Network Architecture for Object-wise Anomaly Detection in Cluttered X-ray Security Imagery* показано, що Faster R-CNN і RetinaNet успішно працюють навіть зі складними рентгенівськими сценами, які містять велику кількість об'єктів [42].

Найбільш поширеними одноетапними моделями залишаються представники сімейства YOLO. На відміну від Faster R-CNN, вони виконують пошук і класифікацію об'єктів одночасно, без проміжного етапу генерації областей пошуку. Саме завдяки цьому досягається висока швидкість роботи, що дозволяє використовувати такі моделі в системах реального часу [43].

Практика застосування моделей YOLO показує, що їх головною перевагою є поєднання швидкої обробки зображень із достатньо високою точністю детекції. Водночас під час роботи з дуже малими або сильно перекритими об'єктами якість розпізнавання може знижуватися порівняно з двоетапними підходами [44].

Останніми роками для аналізу рентгенівських зображень активно розробляються спеціалізовані модифікації YOLO. Одним із прикладів є модель Xray-YOLO-Mamba, створена для роботи зі складними рентгенівськими сценами. Автори відзначають, що запропонована архітектура поєднує високу швидкість роботи та здатність знаходити приховані або частково перекриті предмети [45].

Ще одним представником одноетапних детекторів є SSD (Single Shot MultiBox Detector). Для пошуку об'єктів різного масштабу модель використовує

ознаки з декількох рівнів мережі. За швидкістю роботи SSD перевершує Faster R-CNN, однак у задачах локалізації малих об'єктів зазвичай демонструє дещо нижчу точність [46].

Для задач із вираженим дисбалансом класів часто застосовується RetinaNet. Особливістю цієї архітектури є використання функції втрат Focal Loss, яка дозволяє приділяти більше уваги складним прикладам під час навчання. Це особливо актуально для рентгенівських зображень, де переважна більшість сканів не містить небезпечних предметів [47].

Окрім згорткових мереж, останніми роками активно розвиваються підходи на основі Transformer. Одним із найвідоміших представників цього напрямку є DETR (Detection Transformer). Модель використовує механізм уваги для аналізу взаємозв'язків між різними ділянками зображення, що дозволяє ефективно працювати зі складними сценами та випадками перекриття об'єктів [48].

Попри хороші результати, Transformer-моделі зазвичай потребують більших обчислювальних ресурсів і довшого навчання. З цієї причини останнім часом дедалі більше дослідників звертають увагу на гібридні CNN-Transformer архітектури, які поєднують сильні сторони обох підходів [49].

У таблиці 1.3 наведено порівняння сучасних моделей детекції об'єктів за основними показниками ефективності, зокрема mean Average Precision (mAP), precision, recall та швидкістю роботи.

**Таблиця 1.3** – Порівняння сучасних методів object detection

| <i>Метод</i> | <i>Тип моделі</i> | <i>mAP</i> | <i>Precision</i> | <i>Recall</i> | <i>FPS</i> |
|--------------|-------------------|------------|------------------|---------------|------------|
| Faster R-CNN | Two-stage         | 0.79       | 0.82             | 0.77          | 7          |
| YOLOv5       | One-stage         | 0.87       | 0.89             | 0.85          | 45         |
| SSD          | One-stage         | 0.74       | 0.76             | 0.72          | 32         |
| RetinaNet    | One-stage         | 0.81       | 0.84             | 0.79          | 18         |
| DETR         | Transformer-based | 0.83       | 0.86             | 0.8           | 12         |

Порівняння сучасних моделей показує, що кожна з них має власні переваги та обмеження. Faster R-CNN забезпечує високу точність локалізації, YOLO вирізняється швидкістю, а Transformer-підходи краще працюють зі складними сценами. Саме тому сучасні дослідження переважно спрямовані на пошук оптимального балансу між швидкістю роботи та якістю детекції.

## 1.8 Порівняння результатів сучасних досліджень

Для оцінювання якості моделей виявлення об'єктів зазвичай використовують показники Precision, Recall, F1-score та mean Average Precision (mAP). Серед них найбільш поширеною метрикою є mAP, оскільки вона враховує як правильність класифікації об'єктів, так і точність їх локалізації на зображенні [50].

На результати роботи моделей впливає низка факторів. Серед них архітектура нейронної мережі, особливості набору даних, кількість перекритих об'єктів, співвідношення між класами та якість анотацій. Через це одна й та сама модель може демонструвати різні результати залежно від умов експерименту та характеристик датасету [33].

Одним із найбільш складних наборів даних для оцінювання алгоритмів детекції вважається SIXray. Значна кількість перекритих предметів і виражений дисбаланс між класами суттєво ускладнюють процес виявлення заборонених об'єктів. У роботі *\*SIXray: A Large-scale Security Inspection X-ray Benchmark for Prohibited Item Discovery in Overlapping Images\** було проведено порівняння кількох моделей, серед яких Faster R-CNN та архітектури на основі ResNet. Отримані результати показали, що ефективність традиційних згорткових мереж помітно знижується під час роботи зі складними сценами, де предмети частково перекривають один одного [33].

Серед моделей, які сьогодні найчастіше використовуються для аналізу рентгенівських зображень, важливе місце займає сімейство YOLO. У дослідженні *\*Deep Learning for Detecting Dangerous Objects in X-rays of Luggage\** для пошуку небезпечних предметів застосовувалася модель YOLOv5. За інформацією авторів, значення mean average recall для п'яти категорій заборонених предметів досягло 87 %. Крім достатньо високої точності, модель забезпечила швидку обробку зображень, що дозволяє використовувати її в системах реального часу [29].

Високі результати також демонструють Faster R-CNN і RetinaNet. Дослідження *\*Evaluation of a Dual Convolutional Neural Network Architecture for Object-wise Anomaly Detection in Cluttered X-ray Security Imagery\** показало, що Faster R-CNN забезпечує якісну локалізацію об'єктів навіть у складних рентгенівських сценах. Водночас RetinaNet краще адаптується до задач із дисбалансом класів завдяки використанню функції втрат Focal Loss [42].

Останніми роками все більшої популярності набувають архітектури на основі Transformer. У роботах, присвячених порівнянню моделей виявлення заборонених предметів, зазначається, що DETR та гібридні CNN-Transformer рішення ефективніше працюють із частково прихованими або перекритими об'єктами. Це робить такі підходи перспективними для аналізу складних рентгенівських зображень багажу [30].

У таблиці 1.4 наведено результати сучасних досліджень у сфері автоматизованого виявлення заборонених предметів на рентгенівських зображеннях багажу.

**Таблиця 1.4** – Результати сучасних досліджень з виявлення заборонених предметів

| <i>Дослідження</i>                                                 | <i>Модель</i>    | <i>Dataset</i>          | <i>mAP</i> | <i>Recall</i> | <i>Особливості</i>                      |
|--------------------------------------------------------------------|------------------|-------------------------|------------|---------------|-----------------------------------------|
| SIXray (2019)                                                      | ResNet-based CNN | SIXray                  | 0.76       | 0.73          | проблеми overlap objects                |
| Deep Learning for Detecting Dangerous Objects in X-rays of Luggage | YOLOv5           | baggage X-ray dataset   | 0.87       | 0.85          | real-time detection                     |
| Evaluation of Dual CNN Architecture                                | Faster R-CNN     | cluttered X-ray imagery | 0.81       | 0.79          | висока точність localization            |
| Comparative Evaluation of Illicit Object Detection                 | DETR             | OPIXray, HiXray         | 0.83       | 0.8           | ефективний analysis of occluded objects |

### **1.9 Основні проблеми та перспективи розвитку систем автоматизованого виявлення заборонених предметів**

Незважаючи на помітний прогрес у розвитку комп'ютерного зору та глибокого навчання, автоматичний аналіз рентгенівських зображень багажу залишається складною задачею. Це пов'язано як з особливостями самих зображень, так і з обмеженнями сучасних алгоритмів детекції об'єктів [51].

Однією з найбільших складностей є перекриття предметів усередині багажу. На рентгенівських знімках різні речі часто накладаються одна на одну, через що контури небезпечних об'єктів можуть бути частково прихованими. У

таких умовах навіть сучасні моделі не завжди здатні точно визначити межі об'єкта або правильно його класифікувати [33].

Складною залишається і задача пошуку невеликих предметів. Леза, компактні ножі або окремі елементи вибухових пристроїв займають незначну частину зображення та можуть губитися серед великої кількості інших речей. Через це точність виявлення малих об'єктів зазвичай є нижчою порівняно з більшими предметами [35].

Додаткові труднощі виникають у випадках навмисного приховування небезпечних предметів. У реальних умовах контролю безпеки такі об'єкти можуть бути замасковані або розміщені серед великої кількості інших речей. Це ускладнює їх пошук навіть для моделей, які демонструють хороші результати на стандартних тестових вибірках [34].

На якість детекції впливає і специфіка рентгенівських зображень. Частина матеріалів має схожі характеристики поглинання випромінювання, тому межі між окремими предметами можуть бути малопомітними. Найчастіше подібні труднощі виникають під час аналізу складних сцен із металевими предметами або об'єктами зі змішаних матеріалів [52].

Окремої уваги заслуговує проблема інтерпретації результатів роботи нейронних мереж. Більшість сучасних моделей функціонує як «чорна скринька», тому оператор не завжди може зрозуміти причину конкретного рішення системи. Для сфер, пов'язаних із безпекою, така особливість є суттєвим обмеженням [53].

Важливим фактором також залишається доступність якісних навчальних даних. Незважаючи на появу великих наборів даних, таких як SIXray, PIDray та HiXray, проблема дисбалансу класів і нестачі прикладів для окремих категорій предметів усе ще не втратила актуальності. Особливо це стосується рідкісних або сильно перекритих об'єктів [33].

Крім точності виявлення, для практичного використання велике значення має швидкість роботи алгоритмів. В умовах реального пасажиропотоку система

повинна аналізувати зображення практично без затримок, тому оптимізація моделей залишається одним із важливих напрямів досліджень [45].

Серед перспективних підходів значний інтерес викликають архітектури на основі Transformer та гібридні CNN-Transformer моделі. Завдяки механізму уваги вони краще враховують взаємозв'язки між різними ділянками зображення та ефективніше працюють зі складними сценами.

Ще одним напрямом розвитку є мультимодальні системи, які поєднують декілька джерел інформації: рентгенівські зображення, тривимірні дані, результати dual-energy сканування та додаткові службові відомості. Поєднання різних типів даних дає змогу підвищити надійність виявлення небезпечних предметів [51].

Для розширення навчальних вибірок активно використовуються методи генерації синтетичних рентгенівських зображень. Такий підхід дозволяє збільшити кількість тренувальних прикладів і підвищити здатність моделей працювати з новими даними [38].

Отже, хоча сучасні системи контролю багажу вже демонструють високі результати, низка проблем досі залишається невирішеною. Подальший розвиток Transformer-підходів, мультимодальних систем та методів генерації даних розглядається як один із найбільш перспективних шляхів підвищення ефективності автоматизованого контролю безпеки [49].

#### Метрики оцінювання ефективності моделей

Для оцінювання ефективності моделей комп'ютерного зору та глибокого навчання, що використовуються для виявлення заборонених предметів у багажі, застосовуються спеціальні метрики якості. Дані показники дозволяють оцінити точність роботи моделей, ефективність виявлення небезпечних об'єктів та кількість помилок під час класифікації й локалізації предметів [21].

Однією з базових метрик є асигасу — загальна точність класифікації. Вона визначає частку правильно класифікованих зображень відносно загальної кількості перевірених прикладів. Проте для задач аналізу рентгенівських зображень дана метрика не завжди є достатньо інформативною через дисбаланс

класів, коли кількість зображень без небезпечних предметів значно перевищує кількість сканів із забороненими об'єктами [11].

Для більш точного оцінювання використовуються метрики precision та recall. Метрика precision показує, яка частина об'єктів, визначених системою як небезпечні, дійсно є забороненими предметами. Високе значення precision дозволяє зменшити кількість помилкових спрацювань (false positive). Метрика recall характеризує здатність системи знаходити всі наявні небезпечні об'єкти на зображенні. Низьке значення recall може призводити до пропуску реальних загроз (false negative), що є критичним для систем авіаційної безпеки [21].

Для комплексного оцінювання якості класифікації використовується F1-score — гармонійне середнє між precision та recall. Дана метрика є особливо корисною у випадках дисбалансу класів та дозволяє оцінювати загальну ефективність моделі при роботі зі складними наборами даних [21].

У задачах object detection важливою метрикою є Intersection over Union (IoU). Вона використовується для оцінювання точності локалізації об'єкта та визначає ступінь перекриття між реальною областю об'єкта та прогнозованою обмежувальною рамкою (bounding box). Чим вищим є значення IoU, тим точніше модель визначає місце розташування об'єкта на зображенні [22].

Однією з основних метрик для оцінювання моделей object detection є mean Average Precision (mAP). Дана метрика широко використовується у сучасних дослідженнях комп'ютерного зору та дозволяє оцінювати якість виявлення об'єктів з урахуванням точності локалізації та класифікації. Значення mAP обчислюється як середнє значення Average Precision для всіх класів об'єктів [23].

Для аналізу результатів класифікації також використовується confusion matrix (матриця помилок). Вона дозволяє визначити кількість правильних та помилкових класифікацій для кожного класу об'єктів, а також оцінити типи помилок, які найчастіше допускає модель [24].

Поєднання кількох метрик дозволяє всебічно оцінити роботу моделі та визначити, наскільки вона підходить для задач автоматизованого контролю багажу.

## **Висновки до розділу 1**

У першому розділі було розглянуто особливості контролю багажу в цивільній авіації та основні вимоги до перевезення потенційно небезпечних предметів. Описано принцип роботи рентгенівських систем, особливості формування X-ray зображень і можливості двоенергетичного сканування для розпізнавання різних типів матеріалів.

Проведений аналіз показав, що автоматичне виявлення заборонених предметів у багажі ускладнюється перекриттям об'єктів, низькою інформативністю рентгенівських зображень і нерівномірним розподілом даних між класами. Водночас сучасні методи комп'ютерного зору та глибокого навчання дозволяють ефективно розв'язувати ці задачі. Особливий інтерес становлять моделі детекції об'єктів, зокрема сімейство YOLO, які поєднують високу точність із швидкою обробкою зображень.

Результати проведеного аналізу стали основою для вибору підходів і моделей, що використовуються в подальших розділах роботи.

## РОЗДІЛ 2 ПРОЄКТУВАННЯ ІНТЕЛЕКТУАЛЬНОЇ СИСТЕМИ ВИЯВЛЕННЯ ЗАБОРОНЕНИХ ПРЕДМЕТІВ

### 2.1 Постановка задачі

Щодня через пункти контролю в аеропортах проходить велика кількість багажу, тому швидкість і точність його перевірки мають важливе значення для забезпечення авіаційної безпеки. Основним інструментом такого контролю є рентгенівські сканери, за допомогою яких інспектори аналізують вміст валіз і ручної поклажі.

Попри широке використання рентгенівських систем, процес перевірки значною мірою залежить від людини. Інспектору доводиться працювати зі складними зображеннями, на яких предмети можуть перекривати один одного, частково приховуватися або мати схожий вигляд. За високого навантаження це підвищує ймовірність помилок і пропуску небезпечних об'єктів.

Одним зі способів підвищення ефективності контролю є використання методів комп'ютерного зору та глибокого навчання. У цій роботі розглядається система, яка автоматично аналізує рентгенівські зображення багажу, визначає місце розташування потенційно небезпечних предметів, класифікує їх та надає інспектору додаткову інформацію для прийняття рішення.

Система орієнтована на виявлення таких категорій об'єктів: knife, gun, scissors, wrench та pliers. Для кожного знайденого предмета визначаються координати обмежувальної рамки, клас об'єкта та рівень впевненості моделі.

Робота системи складається з кількох послідовних етапів. Спочатку рентгенівське зображення проходить попередню обробку, після чого передається до моделі детекції об'єктів. Далі аналізуються результати розпізнавання, оцінюється рівень впевненості прогнозу та формується рекомендація щодо подальших дій. У випадках, коли існує ймовірність загрози або модель не впевнена у своєму рішенні, результати передаються на додаткову перевірку інспектору.

Для оцінювання якості роботи використовуються метрики Precision, Recall та mAP, які дають змогу оцінити точність локалізації об'єктів і повноту їх виявлення. Окремою вимогою до системи є можливість роботи в режимі реального часу, що є важливим для практичного використання в умовах аеропорту.

Метою роботи є розробка інтелектуальної системи виявлення заборонених предметів на рентгенівських зображеннях багажу авіапасажирів із використанням методів глибокого навчання та детекції об'єктів.

## **2.2 Архітектура інтелектуальної системи виявлення заборонених предметів у багажі авіапасажирів**

Розроблена система призначена для автоматизованого аналізу рентгенівських зображень багажу та підтримки роботи інспектора авіаційної безпеки. Вона допомагає виявляти потенційно небезпечні предмети, проте остаточне рішення щодо подальших дій залишається за оператором.

Архітектура системи складається з кількох взаємопов'язаних компонентів: модуля введення даних, блоку попередньої обробки зображень, моделі детекції об'єктів YOLOv8, модуля оцінювання рівня загрози, системи рекомендацій та інтерфейсу інспектора.

Робота системи починається з отримання рентгенівського зображення від сканера. Перед аналізом зображення проходить попередню обробку: масштабується до розміру 640×640 пікселів, нормалізується та, за потреби, додатково покращується за допомогою методу CLAHE. Це дозволяє зробити менш помітні об'єкти більш виразними та полегшити їх подальше розпізнавання.

Основним елементом системи є модель YOLOv8, яка використовується для виявлення заборонених предметів на рентгенівських зображеннях. У межах

роботи модель навчається розпізнавати п'ять категорій об'єктів: knife, gun, scissors, wrench і pliers. Для кожного виявлення вона повертає координати обмежувальної рамки, назву класу та значення confidence score.

Після завершення детекції результати додатково обробляються. Алгоритм Non-Maximum Suppression усуває дубльовані рамки, а confidence threshold відсіює передбачення з низькою достовірністю. Значення порогу підбирається експериментально, щоб зберегти баланс між кількістю помилкових спрацьовувань і ризиком пропуску небезпечних предметів.

Оскільки на рентгенівських зображеннях предмети часто перекривають один одного, система враховує не лише клас об'єкта, а й рівень впевненості моделі. На основі цих даних формується оцінка загрози, яка може відповідати низькому, середньому або високому рівню небезпеки.

Далі результати передаються до системи рекомендацій. Інспектор отримує інформацію про виявлений об'єкт, рівень загрози та необхідність додаткової перевірки. Для зручності аналізу на зображенні відображаються обмежувальні рамки та текстові позначки, а в разі виявлення потенційно небезпечних предметів формуються додаткові аудіовізуальні сповіщення.

Запропонована архітектура реалізує підхід human-in-the-loop, за якого автоматизований аналіз поєднується з контролем з боку людини. Це дає змогу використовувати переваги нейронних мереж без втрати контролю над критично важливими рішеннями.

Для наочного представлення запропонованої системи на рисунку 2.1 наведено UML-схему архітектури інтелектуальної системи виявлення заборонених предметів у багажі авіапасажирів.

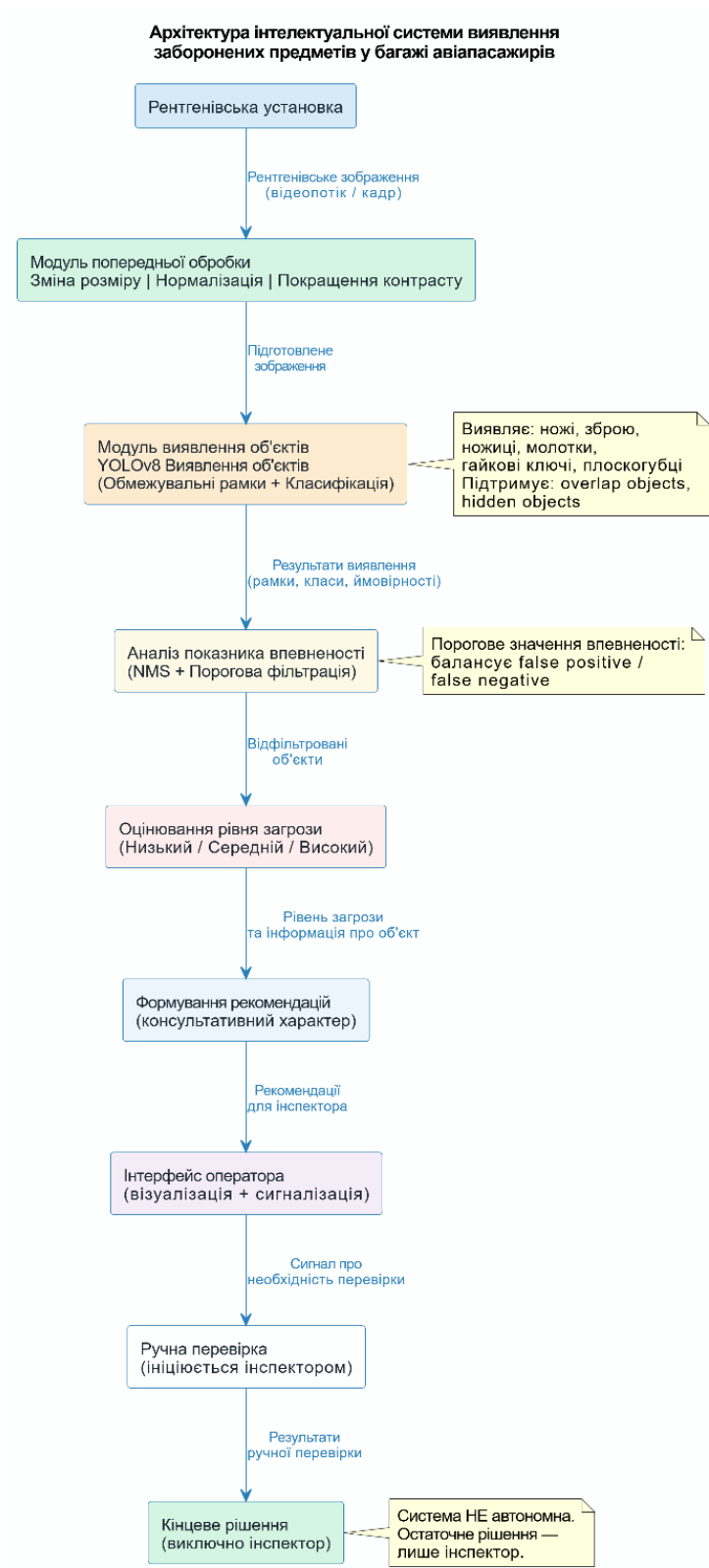


**Рисунок 2.1 – UML-схема архітектури інтелектуальної системи виявлення заборонених предметів у багажі авіапасажирів**

## 2.3 Опис датасету

У роботі використано датасет SIXray, який містить рентгенівські зображення багажу з різними типами заборонених предметів. Завдяки великому обсягу даних і поширеності в наукових дослідженнях він став одним із основних наборів даних для оцінювання моделей виявлення небезпечних об'єктів.

SIXray містить рентгенівські зображення багажу, отримані в реальних умовах роботи систем контролю безпеки. Характерною особливістю набору даних є наявність складних сцен із перекриттям предметів, частково прихованими об'єктами та значною різноманітністю їх розташування. Такі особливості наближають умови дослідження до реальних сценаріїв контролю багажу в аеропортах.

У межах даної роботи використовується підмножина датасету SIXray, що містить п'ять категорій заборонених предметів: вогнепальна зброя (gun), ножі (knife), ножиці (scissors), гайкові ключі (wrench) та плоскогубці (pliers). Для кожного об'єкта наявні анотації у вигляді координат обмежувальних рамок та відповідних класів.

На рисунку 2.2 зображено приклади рентгенівських зображень з датасету.

Важливою особливістю датасету є значний дисбаланс між кількістю зображень із забороненими предметами та зображень без них. Така структура даних відповідає реальним умовам функціонування систем контролю багажу, де більшість перевірених сумок не містять небезпечних об'єктів. Саме тому використання SIXray дозволяє проводити оцінювання моделей у наближених до практичного застосування умовах.

З огляду на поширеність датасету в наукових дослідженнях, наявність складних рентгенівських сцен та достатню кількість анотованих зображень, набір даних SIXray було обрано як основу для навчання та порівняння моделей YOLOv8n, Faster R-CNN та Deformable DETR.

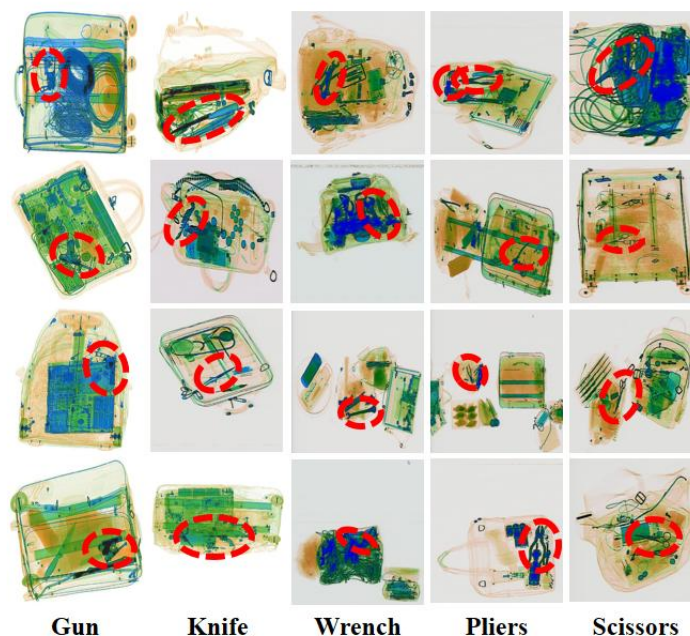


**Рисунок 2.2** – Приклад рентгенівських знімків та класифікація об'єктів в обраному датасеті [56]

## 2.4 Попередня обробка рентгенівських зображень

Рентгенівські зображення багажу суттєво відрізняються від звичайних фотографій, що створює додаткові труднощі для автоматичного аналізу. Найчастіше проблеми пов'язані з низькою контрастністю, перекриттям предметів, наявністю прихованих об'єктів і шумом на зображенні. Крім того, якість даних може відрізнятися залежно від типу рентгенівського обладнання. Через це перед навчанням і тестуванням моделі виконується попередня обробка зображень.

Модуль попередньої обробки (preprocessing pipeline) використовується для стандартизації вхідних даних, покращення якості рентгенівських зображень

та підвищення стійкості моделі до складних сценаріїв виявлення. У даній роботі preprocessing реалізовано з використанням бібліотеки OpenCV, а для аугментації даних застосовується бібліотека Albumentations.

Першим етапом preprocessing pipeline є масштабування зображення (resize). Оскільки модель YOLOv8 працює з фіксованим розміром вхідного зображення, усі рентгенівські знімки приводяться до формату 640×640 пікселів. Для уникнення деформації об'єктів використовується метод letterboxing, при якому зображення масштабується зі збереженням пропорцій, а вільні області заповнюються нейтральними пікселями. Це дозволяє зберегти геометрію предметів та уникнути помилок класифікації.

Другим етапом є нормалізація значень пікселів (normalization). Значення пікселів переводяться з діапазону [0;255] у діапазон [0;1] шляхом ділення на 255. Нормалізація забезпечує стабільну роботу нейромережевої моделі та покращує процес навчання.

Після цього виконується покращення контрастності зображення за допомогою методу CLAHE (Contrast Limited Adaptive Histogram Equalization). Завдяки підвищенню локального контрасту окремі деталі стають більш виразними, а межі предметів — чіткішими. Це полегшує виявлення об'єктів, які частково перекриваються або слабо виділяються на рентгенівському зображенні.

CLAHE застосовується лише у випадках, коли зображення має недостатній рівень контрастності. Такий підхід дозволяє уникнути надмірної обробки якісних знімків та зменшує ризик підсилення шуму.

Окремим компонентом preprocessing pipeline є аугментація навчальних даних (data augmentation). Аугментація використовується лише для training set та не застосовується до validation і test вибірок, оскільки тестування моделі повинно виконуватися на реалістичних незмінених зображеннях.

У даній роботі використовуються такі методи аугментації:

- horizontal flip;
- scaling;

- random rotation;
- brightness/contrast adjustment.

Під час підготовки даних застосовувалися методи аугментації, зокрема відображення, масштабування, обертання та зміна яскравості зображень. Це дозволило відтворити різні варіанти розташування предметів у багажі та зробити модель менш чутливою до відмінностей у якості рентгенівських зображень.

У результаті модель краще розпізнає об'єкти в умовах перекриття предметів і складних конфігурацій багажу, що позитивно впливає на її здатність до узагальнення.

Для validation set та test set застосовуються лише базові етапи preprocessing: resize, normalization та, за необхідності, CLAHE. Такий підхід забезпечує коректне оцінювання метрик precision, recall та mAP в умовах, максимально наближених до реальної роботи системи.

На рисунку 2.3 наведено узагальнений конвеєр попередньої обробки рентгенівських зображень.

Якість попередньої обробки безпосередньо впливає на результати роботи моделі. Використання масштабування, нормалізації, покращення контрастності та аугментації дозволяє отримати більш однорідні дані для навчання і тестування. У результаті модель YOLOv8 краще справляється з аналізом рентгенівських зображень та точніше виявляє потенційно небезпечні предмети.

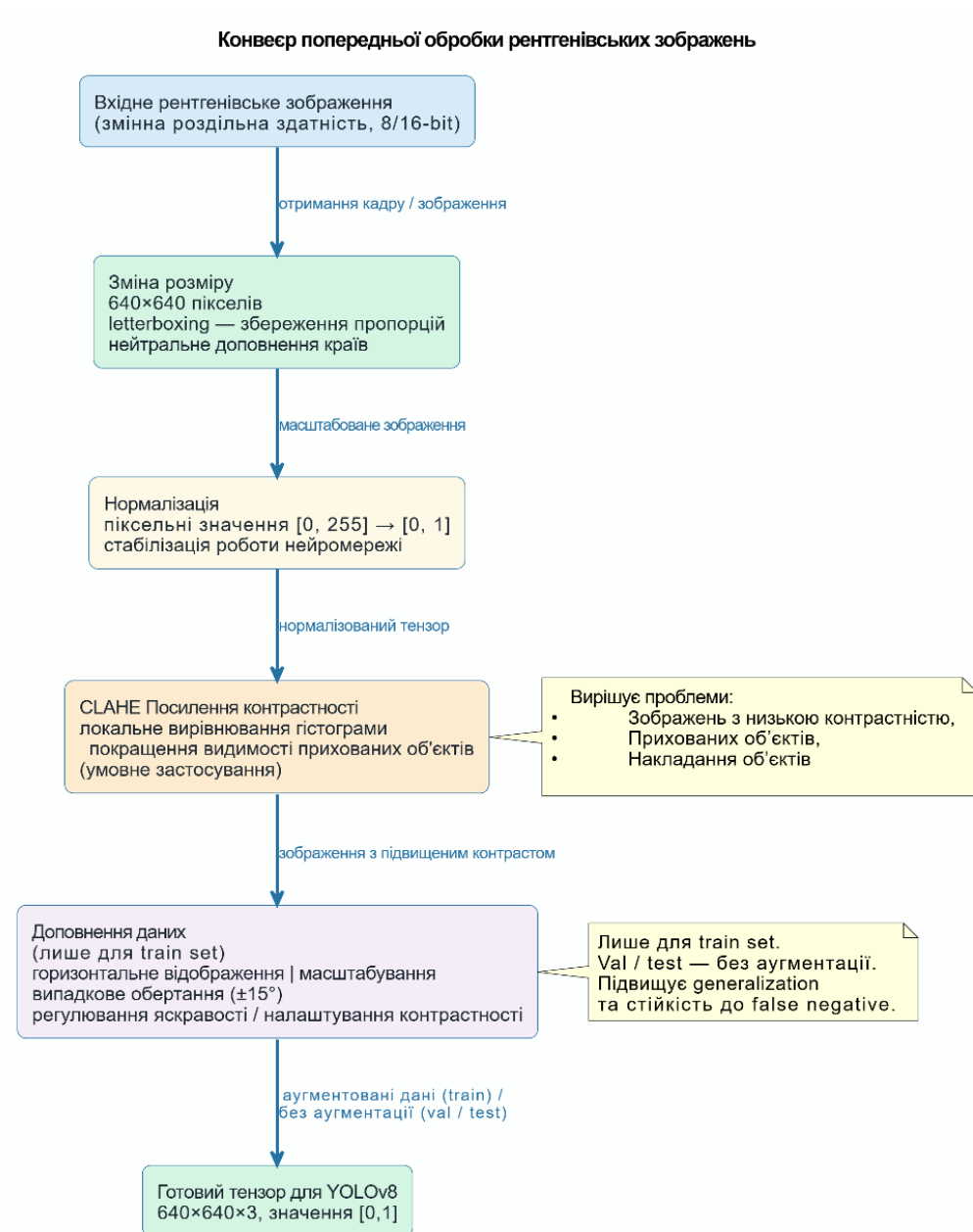


**Рисунок 2.3 – Конвертер попередньої обробки рентгенівських зображень**

## 2.5 Проектування модулів виявлення заборонених предметів

Ключовим компонентом розроблюваної інтелектуальної системи є модуль автоматичного виявлення заборонених предметів на рентгенівських зображеннях багажу. Саме від якості роботи даного модуля залежить

ефективність виявлення потенційно небезпечних об'єктів та можливість своєчасного інформування інспектора про наявність загрози.

У роботі досліджуються три моделі детекції об'єктів: YOLOv8n, Faster R-CNN та Deformable DETR. Кожна з них використовує власний підхід до пошуку та локалізації об'єктів, тому їх порівняння дає можливість визначити, яка архітектура найкраще підходить для аналізу рентгенівських зображень багажу.

YOLOv8n належить до сімейства одноетапних моделей детекції об'єктів. Особливістю даного підходу є одночасне виконання локалізації та класифікації об'єктів під час одного проходу нейронної мережі. Завдяки цьому забезпечується висока швидкість роботи та можливість використання моделі в системах, що працюють у режимі реального часу.

Як основну модель для автоматичного аналізу рентгенівських зображень обрано YOLOv8n. Вона поєднує високу швидкість з відносно невеликим розміром моделі та добре працює з об'єктами різних масштабів, що важливо для задач контролю багажу.

Faster R-CNN є представником двоетапних моделей детекції об'єктів. На першому етапі виконується генерація кандидатних областей, які потенційно можуть містити об'єкти, а на другому — їх класифікація та уточнення координат обмежувальних рамок.

Основною перевагою Faster R-CNN є висока точність локалізації об'єктів. Саме тому дана архітектура широко використовується в задачах, де якість виявлення є важливішою за швидкість обробки даних. У межах даної роботи Faster R-CNN використовується як представник класичних двоетапних методів детекції.

Використання моделей різних типів дає можливість порівняти їхню роботу в однакових умовах і визначити сильні та слабкі сторони кожного підходу. На основі отриманих результатів буде обрано модель, яка найкраще підходить для аналізу рентгенівських зображень у межах розроблюваної системи.

Особливістю Deformable DETR є використання деформованої уваги, що дозволяє зосереджувати обчислення лише на найбільш інформативних ділянках зображення. Завдяки цьому модель забезпечує ефективнішу роботу порівняно з базовою архітектурою DETR та потребує менше часу для навчання.

Використання трьох моделей різних архітектурних типів дозволяє комплексно дослідити можливості сучасних методів детекції об'єктів у задачі виявлення заборонених предметів на рентгенівських зображеннях багажу. Подальше експериментальне оцінювання дасть змогу визначити найбільш ефективну модель для використання у складі розроблюваної інтелектуальної системи.

## **2.6 Логіка ручної перевірки та взаємодія з інспектором**

Попри високі результати сучасних моделей комп'ютерного зору, у системах авіаційної безпеки остаточне рішення не може повністю покладатися на алгоритм. Тому запропонована система використовується як інструмент підтримки інспектора, а не як автономний засіб контролю.

Основою логіки прийняття рішень є показник впевненості моделі (confidence score), який відображає достовірність кожного виявлення. У роботі використовуються два порогові значення: 0,30 та 0,65. Передбачення з confidence score нижче 0,30 відкидаються як ненадійні. Якщо значення знаходиться в межах від 0,30 до 0,65, система рекомендує додаткову перевірку. За значень вище 0,65 виявлення вважається достатньо достовірним і потребує обов'язкового огляду багажу.

Окрему складність становлять випадки, коли предмети перекривають один одного або частково приховують небезпечний об'єкт. У таких ситуаціях навіть правильне виявлення може супроводжуватися нижчим рівнем

впевненості, тому результати із середніми значеннями confidence score також передаються на ручну перевірку.

Для оцінювання потенційної небезпеки використовується трирівнева шкала загроз: низький, середній та високий рівень. Низький рівень відповідає відсутності достовірних виявлень, середній формується за невпевнених прогнозів або складних умов аналізу, а високий — у разі виявлення об'єктів підвищеного ризику чи отримання високого значення confidence score.

Після аналізу система відображає межі виявлених об'єктів, їх клас, рівень впевненості та оцінку загрози. Для привернення уваги інспектора використовуються кольорові позначки та повідомлення про потенційно небезпечні об'єкти.

Якщо система визначає середній або високий рівень загрози, результати передаються оператору для додаткової перевірки. Інспектор отримує рентгенівське зображення з виділеними підозрілими ділянками та приймає рішення щодо подальшого огляду багажу.

Такий підхід поєднує швидкість автоматичного аналізу з контролем з боку людини, що особливо важливо для задач авіаційної безпеки.

## **2.7 Система рекомендацій для інспектора служби авіаційної безпеки**

Після аналізу рентгенівського зображення система формує рекомендації для інспектора на основі результатів роботи моделі YOLOv8. Її завдання полягає у швидкому відображенні важливої інформації та приверненні уваги до потенційно небезпечних об'єктів.

Під час формування рекомендацій враховуються клас виявленого об'єкта, рівень впевненості моделі та оцінка загрози. На основі цих даних система може працювати в одному з чотирьох режимів: «Без загрози», «Рекомендація уваги», «Рекомендація ручної перевірки» або «Висока загроза».

Якщо небезпечних предметів не виявлено, додаткові повідомлення не відображаються. Для об'єктів із середнім рівнем достовірності система рекомендує оператору звернути увагу на відповідну ділянку зображення. У разі підвищеного ризику формується рекомендація щодо ручної перевірки багажу. Найвищий рівень загрози використовується для об'єктів підвищеної небезпеки або виявлень із високим рівнем впевненості.

Взаємодія з оператором здійснюється через графічний інтерфейс. На рентгенівському зображенні відображаються обмежувальні рамки, назва виявленого об'єкта та значення confidence score. Для швидкого сприйняття інформації використовується кольорове маркування: зелений колір відповідає низькому рівню загрози, жовтий — середньому, помаранчевий — підвищеному, а червоний — високому.

Додатково система формує короткі текстові повідомлення з інформацією про тип виявленого предмета, рівень загрози та рекомендовані дії. Для привернення уваги оператора можуть використовуватися кольорові попередження та звукові сигнали.

У разі необхідності додаткової перевірки інспектор отримує рентгенівське зображення з виділеними підозрілими ділянками. Це допомагає швидше локалізувати потенційно небезпечний об'єкт і скоротити час перевірки багажу.

Таким чином, система рекомендацій спрощує роботу оператора, допомагає швидше орієнтуватися в результатах аналізу та зосереджувати увагу на найбільш важливих об'єктах, залишаючи остаточне рішення за інспектором.

## **Висновки до розділу 2**

У другому розділі описано структуру інтелектуальної системи для автоматичного аналізу рентгенівських зображень багажу. Основою системи є

модель YOLOv8, яка використовується для виявлення та локалізації потенційно небезпечних предметів.

Окрему увагу приділено підготовці даних. Перед подачею до моделі зображення проходять масштабування, нормалізацію та покращення контрастності, а для навчальної вибірки додатково застосовується аугментація. Це дозволяє підвищити стійкість моделі до складних рентгенівських сцен, де об'єкти можуть перекриватися або бути частково прихованими.

Також розглянуто механізм оцінювання рівня загрози та взаємодії системи з інспектором. Запропонований підхід поєднує автоматичний аналіз зображень із контролем з боку людини, що є особливо важливим для задач авіаційної безпеки.

Описані в розділі рішення стали основою для подальших експериментів і порівняння моделей детекції об'єктів, результати яких наведено в наступному розділі.

## **РОЗДІЛ 3 РОЗРОБЛЕННЯ ТА ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ ІНТЕЛЕКТУАЛЬНОЇ СИСТЕМИ ВИЯВЛЕННЯ ЗАБОРОНЕНИХ ПРЕДМЕТІВ**

### **3.1 Середовище розробки та програмна реалізація системи**

Реалізація інтелектуальної системи виявлення заборонених предметів у багажі авіапасажирів виконувалась у хмарному середовищі Google Colaboratory (Colab) із використанням графічного прискорювача NVIDIA Tesla T4 з обсягом відеопам'яті 16 ГБ. Вибір Google Colab як основного середовища розробки зумовлений можливістю використання GPU для навчання нейромережових моделей, вбудованою підтримкою Python-екосистеми без необхідності локального налаштування середовища, а також інтеграцією з Google Drive для збереження датасетів, контрольних точок та результатів експериментів.

Основною мовою програмування системи є Python 3.10, що широко використовується у задачах машинного навчання та комп'ютерного зору завдяки наявності великої кількості спеціалізованих бібліотек. У таблиці 3.1 наведено перелік основних програмних засобів, використаних у межах реалізації проєкту.

Центральним компонентом програмної реалізації є фреймворк PyTorch, на базі якого були реалізовані та досліджені три різні архітектурні підходи до задачі виявлення об'єктів. Для реалізації моделі YOLOv8n використовувалась бібліотека Ultralytics, що надає зручний програмний інтерфейс для завантаження попередньо навчених ваг, налаштування параметрів навчання, виконання валідації та отримання результатів inference. Модель Faster R-CNN реалізовувалась за допомогою бібліотеки torchvision, яка містить готові реалізації сучасних алгоритмів комп'ютерного зору. Для реалізації DETR (Detection Transformer) використовувалась бібліотека Transformers, що забезпечує інструменти для роботи з трансформерними архітектурами та дозволяє адаптувати їх до задач object detection.

**Таблиця 3.1** — Програмні засоби та їх призначення

| <i>Засіб</i>   | <i>Версія</i> | <i>Призначення у проєкті</i>                                    |
|----------------|---------------|-----------------------------------------------------------------|
| Python         | 3.10          | Основна мова програмування системи                              |
| PyTorch        | 2.0           | Фреймворк глибокого навчання, GPU-прискорення                   |
| Ultralytics    | 8.0           | API навчання та inference моделі YOLOv8                         |
| torchvision    | 0.15          | Утиліти для роботи з зображеннями та трансформаціями            |
| Transformers   | 4.44.2        | Реалізація моделі DETR                                          |
| OpenCV         | 4.8           | Попередня обробка зображень, CLAHE, візуалізація bounding boxes |
| Albumentations | 1.3           | Аугментація навчальних даних із підтримкою bounding boxes       |
| Gradio         | 5.50.0        | Веб-інтерфейс для демонстрації inference системи                |
| matplotlib     | 3.7           | Побудова графіків втрат, метрик та кривих Precision-Recall      |
| pandas         | 2.0           | Аналіз та обробка табличних результатів навчання                |
| Google Colab   | -             | Хмарне середовище розробки з доступом до GPU Tesla T4           |
| Google Drive   | -             | Зберігання датасету, вагів моделей та результатів               |

Таким чином, у межах експериментального дослідження було реалізовано та порівняно три принципово різні підходи до виявлення об'єктів: одностадійний детектор YOLOv8n, двостадійний детектор Faster R-CNN та трансформерну модель DETR. Такий підхід дозволив оцінити ефективність різних архітектурних рішень для задачі автоматизованого аналізу рентгенівських зображень багажу авіапасажирів. На рисунку 3.1 зображено блок-схему обраних архітектурних підходів.

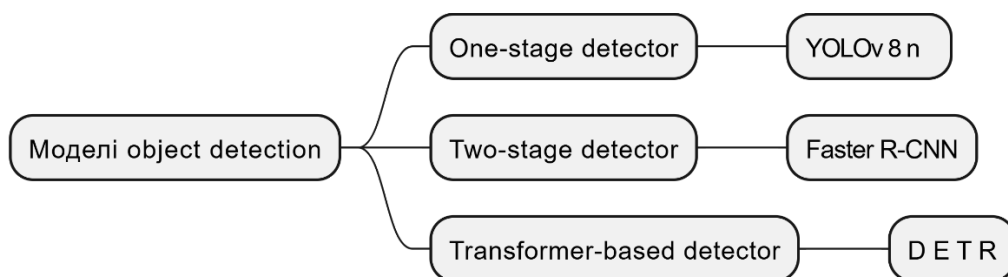


**Рисунок 3.1** – Архітектурні підходи, використані в експериментальному дослідженні

PyTorch забезпечував апаратне прискорення обчислень через CUDA та підтримку автоматичного диференціювання під час процесу навчання моделей. Навчання виконувалось із використанням GPU Tesla T4, що дозволило суттєво скоротити час тренування моделей порівняно з використанням центрального процесора.

Попередня обробка рентгенівських зображень реалізована засобами OpenCV і включала масштабування вхідних зображень до розміру  $640 \times 640$  пікселів, нормалізацію значень пікселів та покращення контрастності за допомогою алгоритму CLAHE. Також OpenCV використовувався для візуалізації результатів детекції шляхом накладання bounding boxes та підписів класів на зображення.

Для підвищення стійкості моделей до варіацій вхідних даних використовувалась бібліотека Albumentations, що забезпечує коректну аугментацію зображень разом із відповідними анотаціями bounding boxes. У процесі навчання застосовувались горизонтальне віддзеркалення, зміна масштабу, випадкові повороти та модифікація яскравості й контрастності зображень.

Аналіз результатів навчання та візуалізація метрик виконувались за допомогою бібліотек matplotlib і pandas. З їх використанням формувались

графіки функції втрат, таблиці порівняння моделей та діаграми співвідношення швидкодії й точності.

Демонстраційний інтерфейс системи реалізовано за допомогою бібліотеки Gradio, яка дозволяє створити веб-інтерфейс без необхідності окремого backend-розгортання. Через інтерфейс користувач може завантажити рентгенівське зображення та отримати результат inference у вигляді зображення з позначеними bounding boxes, назвами виявлених об'єктів та значеннями confidence score.

Практичний pipeline системи організовано як послідовність етапів: підготовка датасету, попередня обробка зображень, навчання моделей, валідація, inference та візуалізація результатів. Для навчання використовувались готові train, validation та test підмножини датасету SIXray. Процес тренування моделей запускався безпосередньо з Colab-ноутбуків із автоматичним збереженням ваг найкращих моделей та логів навчання до Google Drive.

Усі конфігураційні файли, ноутбуки, ваги моделей та результати експериментів зберігались у структурованому вигляді на Google Drive. Такий підхід забезпечує відтворюваність експериментів та спрощує подальшу модифікацію й масштабування системи.

### **3.2 Формування експериментального набору даних**

Для проведення експериментального дослідження використовувався відкритий набір даних SIXray, призначений для задач автоматичного виявлення заборонених предметів на рентгенівських зображеннях багажу авіапасажирів. Даний набір даних містить анотовані рентгенівські зображення, на яких представлені різні категорії потенційно небезпечних предметів, що можуть бути присутніми в багажі.

У межах дослідження використовувалась готова структура набору даних, яка вже містила розподіл на тренувальну, валідаційну та тестову підмножини. Кількість зображень, що містила кожна з підмножин наведено у таблиці 3.2. Такий підхід забезпечує коректність проведення експериментів та дозволяє об'єктивно оцінювати результати навчання моделей на даних, які не використовувалися під час тренування.

**Таблиця 3.2** – Розподіл зображень у наборі даних SIXray

| <i>Підмножина</i>         | <i>Кількість зображень</i> |
|---------------------------|----------------------------|
| Зображення для тренування | 11638                      |
| Зображення для валідації  | 1662                       |
| Зображення для тестування | 831                        |
| Всього                    | 14131                      |

Підготовлений набір даних зберігався у структурі каталогів, сумісній із фреймворком Ultralytics YOLO та іншими реалізаціями моделей object detection. Для кожного зображення існував відповідний файл анотацій, який містив інформацію про координати обмежувальних рамок та належність об'єктів до певного класу.

### 3.2.1 Формат анотацій та організація даних

Анотації в наборі даних зберігались у форматі YOLO, де кожен текстовий файл відповідає окремому зображенню та містить інформацію про всі об'єкти, присутні на ньому. Кожен рядок анотації описує один об'єкт і складається з ідентифікатора класу та нормалізованих координат обмежувальної рамки.

Class\_id визначає клас об'єкта, а інші параметри задають координати центру рамки та її розміри відносно ширини і висоти зображення.

У дослідженні використовувались п'ять категорій заборонених предметів: пістолети (Gun), ножі (Knife), плоскогубці (Pliers), ножиці (Scissors) та гайкові ключі (Wrench) (табл.3.3).

Під час попереднього аналізу набору даних було визначено кількість об'єктів кожного класу, що дозволило оцінити баланс даних між категоріями.

**Таблиця 3.3** – Розподіл об'єктів за класами

| <i>Клас</i>   | <i>Кількість об'єктів</i> |
|---------------|---------------------------|
| Пістолети     | 7340                      |
| Ножі          | 3513                      |
| Плоскогубці   | 8389                      |
| Ножиці        | 2059                      |
| Гайкові ключі | 4656                      |

Аналіз розподілу класів показує наявність певної нерівномірності між категоріями. Найбільш представленим класом є Pliers, тоді як найменшу кількість прикладів містить клас Scissors. Подібна диспропорція є характерною для реальних наборів даних та може впливати на точність детекції окремих категорій об'єктів.

На рисунку 3.2 наведено приклад рентгенівського зображення з накладання анотації на нього

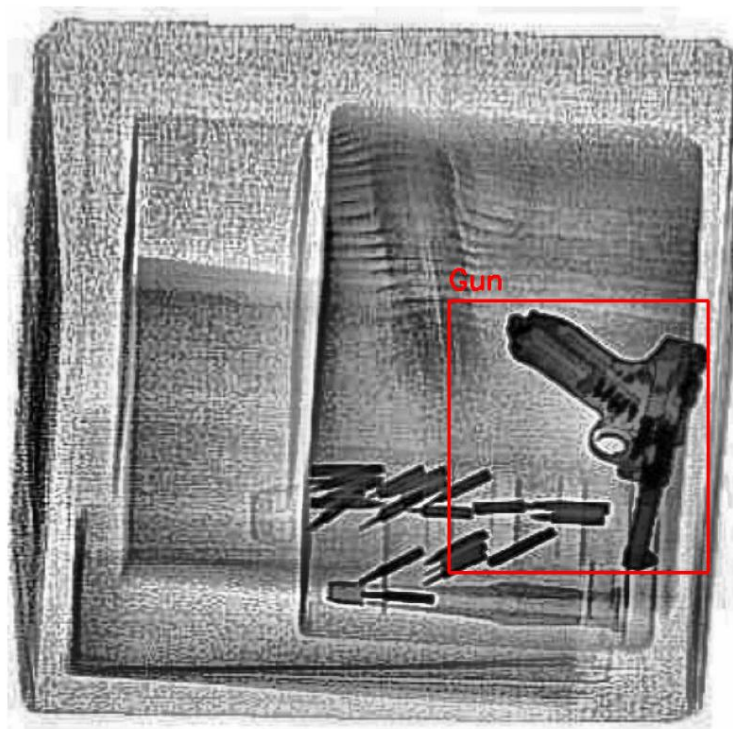


**Рисунок 3.2** – Приклад рентгенівського зображення з анотаціями

### *3.2.2 Попередня обробка та аугментация*

Перед початком навчання всі зображення проходили етап попередньої обробки. Основною метою preprocessing було приведення вхідних даних до єдиного формату та покращення якості ознак, що використовуються моделями для виявлення об'єктів.

Попередня обробка включала масштабування зображень до розміру  $640 \times 640$  пікселів, нормалізацію значень пікселів та покращення локального контрасту за допомогою алгоритму CLAHE (Contrast Limited Adaptive Histogram Equalization). Застосування CLAHE дозволяє підсилити слабкоконтрастні ділянки рентгенівських зображень і зробити контури об'єктів більш помітними.

Для підвищення узагальнюючої здатності моделей використовувалась аугментація навчальних даних. Реалізація аугментації виконувалась за допомогою бібліотеки Albumentations із автоматичним коригуванням координат bounding boxes після кожного перетворення.

**Таблиця 3.4** – Використані операції попередньої обробки та аугментації

| <i>Операція</i>     | <i>Призначення</i>                   |
|---------------------|--------------------------------------|
| Resize (640×640)    | Уніфікація розміру вхідних зображень |
| Normalization       | Стабілізація процесу навчання        |
| CLAHE               | Підвищення локального контрасту      |
| Horizontal Flip     | Збільшення різноманітності даних     |
| Random Rotation     | Підвищення стійкості до поворотів    |
| Scaling             | Стійкість до зміни масштабу          |
| Brightness/Contrast | Імітація різних умов сканування      |

На рисунку 3.3 наведено приклад виконаної аугментації з рентгенівським зображення і порівняно результат з оригінальним фото.

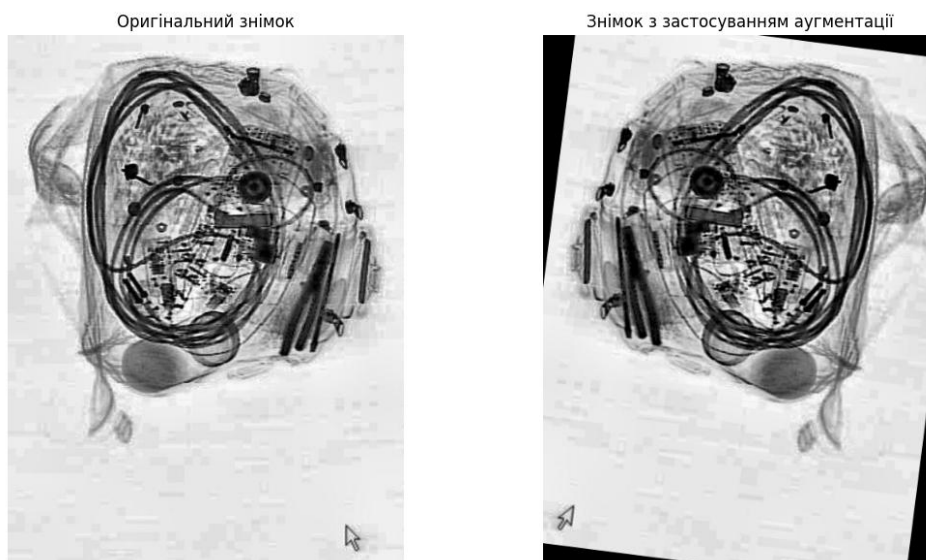


**Рисунок 3.3** – Результат аугментації даних

### 3.2.3 Підготовка даних для навчання моделей

Попри використання одного й того самого датасету, процес підготовки даних для кожної моделі відрізнявся. Причиною є різні вимоги архітектур до формату зображень і анотацій.

Для YOLOv8n було достатньо стандартного формату анотацій YOLO. Faster R-CNN працює з тензорами PyTorch, тому анотації додатково перетворювалися у відповідний формат із координатами рамок та ідентифікаторами класів. Для Deformable DETR використано AutoImageProcessor із бібліотеки Transformers, який автоматично виконує необхідну підготовку даних перед подачею їх до моделі (табл. 3.5).

**Таблиця 3.5** – Особливості підготовки даних моделей

| <i>Модель</i> | <i>Особливості підготовки</i>                  |
|---------------|------------------------------------------------|
| YOLOv8n       | Використання анотацій у форматі YOLO           |
| Faster R-CNN  | Перетворення анотацій у формат PyTorch tensors |
| DETR          | Підготовка через AutoImageProcessor            |

Для всіх моделей використовувалися однакові тренувальна, валідаційна та тестова вибірки. Завдяки цьому відмінності в результатах пов'язані насамперед з особливостями самих архітектур, а не з набором даних.

## 3.2 Результати навчання моделі YOLOv8n

Для навчання YOLOv8n використовувався датасет SIXray та обчислювальні ресурси Google Colab з графічним прискорювачем NVIDIA

A100. Модель ініціалізувалася попередньо навченими вагами, що дало змогу швидше досягти стабільних результатів і покращити якість виявлення об'єктів.

Під час навчання тренувальна вибірка використовувалася для оновлення параметрів моделі, а валідаційна — для оцінки її якості. Після кожної епохи обчислювалися основні метрики, що дозволяло контролювати процес навчання та своєчасно виявляти можливі ознаки перенавчання.

У таблиці 3.6 наведено, які були обрані параметри для навчання моделі YOLOv8n.

**Таблиця 3.6** – Параметри навчання моделі YOLOv8n

| <i>Параметри</i>        | <i>Значення</i>    |
|-------------------------|--------------------|
| Архітектура             | YOLOv8n            |
| Розмір зображення       | 640×640            |
| Кількість епох          | 40                 |
| Batch size              | 16                 |
| Попередньо навчені ваги | YOLOv8n pretrained |
| Пристрій навчання       | NVIDIA A100        |
| Набір даних             | SIXray             |

### 3.3.1 Аналіз процесу навчання

Для оцінювання процесу навчання аналізувалися значення функції втрат і основних метрик якості. Зміна цих показників дозволяє простежити, наскільки стабільно навчається модель та чи здатна вона узагальнювати отримані знання на нових даних.

Як видно з рисунка 3.4, протягом навчання значення функцій втрат поступово зменшувалися як для тренувальної, так і для валідаційної вибірок.

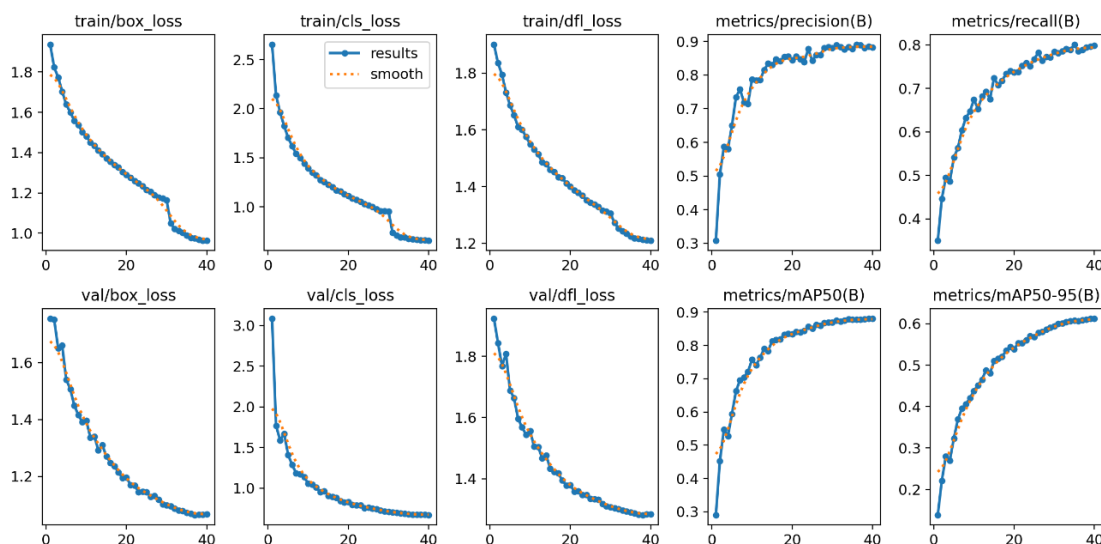


**Рисунок 3.4** – Динаміка зміни функції втрат та основних метрик моделі YOLO8n під час навчання

Водночас спостерігалось зростання основних метрик якості. Наприкінці навчання Precision досягло 0,89, Recall — 0,80, а mAP50 — близько 0,88. Стабілізація показників на останніх епохах свідчить про досягнення моделлю збіжності.

### 3.3.2 Оцінювання якості моделі

Для оцінювання ефективності моделі використовувались метрики Precision, Recall, mAP50 та mAP50–95, які є стандартними показниками якості для задач object detection (таб.3.7).

**Таблиця 3.7** – Загальні результати моделі YOLOv8n

| <i>Метрики</i> | <i>Значення</i> |
|----------------|-----------------|
| Precision      | 0.8817          |
| Recall         | 0.6231          |
| mAP50          | 0.8104          |
| Map50-95       | 0.5796          |

Значення отриманих метрик вказують на високу якість роботи моделі. Найкращий результат спостерігається за показником Precision (0,8817), що свідчить про невелику кількість хибних спрацьовувань. Значення Recall дорівнює 0,6231, а mAP50 становить 0,8104. Це означає, що модель достатньо точно визначає розташування об'єктів та правильно класифікує більшість заборонених предметів на рентгенівських зображеннях.

### 3.3.3 Аналіз результатів за окремими класами

Для більш детального аналізу було проведено оцінювання ефективності моделі окремо для кожного класу об'єктів.

У таблиці 3.8 наведено отримані метрики в результаті тренування моделі YOLOv8n за кожним з класів окремо.

**Таблиця 3.8** – Метрики моделі YOLOv8n за класами

| <i>Класи</i>  | <i>Precision</i> | <i>Recall</i> | <i>mAP50</i> |
|---------------|------------------|---------------|--------------|
| Пістолети     | 0.946            | 0.94          | 0.973        |
| Ножі          | 0.885            | 0.766         | 0.854        |
| Плоскогубці   | 0.927            | 0.763         | 0.9          |
| Ножиці        | 0.901            | 0.754         | 0.857        |
| Гайкові ключі | 0.846            | 0.683         | 0.807        |

Найкращі результати модель показала для класу «Пістолети», де значення mAP50 досягло 0,973. Для інших категорій показники також залишаються високими, хоча дещо нижчі значення спостерігаються для класу «Гайкові ключі». Це може бути пов'язано зі складнішою формою об'єктів, їх частковим перекриттям або меншою кількістю прикладів у навчальній вибірці (рис. 3.5).



**Рисунок 3.5** – Матриця помилок моделі YOLOv8n

Матриця помилок підтверджує, що більшість об'єктів класифікується правильно. Основна частина помилок пов'язана з невиявленням окремих предметів, тоді як випадки плутанини між різними класами трапляються рідко.

### 3.3.4 Приклади роботи моделі

Для якісної оцінки роботи моделі було проаналізовано результати детекції на тестових зображеннях набору даних SIXray (рис. 3.6).

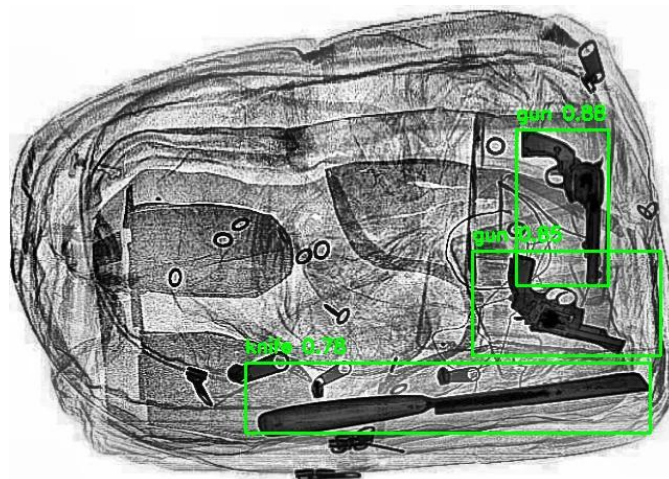


**Рисунок 3.6** – Приклад успішного виявлення заборонених предметів за допомогою моделі YOLOv8n

На наведеному прикладі видно, що модель успішно локалізує заборонені предмети та коректно визначає їх належність до відповідних класів. Крім координат об'єктів система також формує оцінку впевненості (confidence score), що надалі використовується підсистемою оцінювання рівня загрози.

### Висновки до підрозділу 3.3

Під час дослідження модель YOLOv8n показала хороші результати на рентгенівських зображеннях набору SIXray. Вона успішно знаходила більшість заборонених предметів та забезпечила високі значення основних метрик якості.

Аналіз тестових прикладів також показав, що модель коректно визначає розташування об'єктів і рідко допускає помилки класифікації. З огляду на отримані результати YOLOv8n була обрана для подальшого порівняння з Faster R-CNN та Deformable DETR.

### **3.3 Результати навчання моделі Faster R-CNN**

#### *3.4.1 Налаштування та параметри навчання моделі FASTER R-CNN*

Для порівняння також було обрано модель Faster R-CNN — один із найвідоміших двоетапних детекторів об'єктів. Ця архітектура часто використовується як орієнтир під час оцінювання нових моделей детекції.

Навчання проводилося на датасеті SIXray із використанням тих самих тренувальної, валідаційної та тестової вибірок, що й для інших моделей. Такий підхід дає змогу оцінювати відмінності між архітектурами без впливу додаткових факторів.

У роботі використано реалізацію Faster R-CNN з бібліотеки PyTorch torchvision. Модель була ініціалізована вагами, попередньо навченими на наборі даних COCO, а як backbone обрано ResNet-50 із Feature Pyramid Network (FPN), що добре підходить для виявлення об'єктів різного масштабу.

У таблиці 3.9 наведено обрані параметри для навчання моделі Faster R-CNN.

**Таблиця 3.9** – Параметри навчання моделі Faster R-CNN

| <i>Параметр</i>    | <i>Значення</i> |
|--------------------|-----------------|
| Архітектура        | Faster R-CNN    |
| Backbone           | ResNet-50 FPN   |
| Попереднє навчання | COCO            |
| Кількість епох     | 10              |
| Batch size         | 4               |
| Optimizer          | SGD             |
| Пристрій навчання  | NVIDIA A100     |
| Набір даних        | SIXray          |

### 3.4.2 Аналіз процесу навчання

Під час навчання відстежувалося значення функції втрат, яка відображає якість локалізації об'єктів та їх класифікації. Динаміку зміни цього показника наведено на рисунку 3.7.

З графіка видно, що значення loss поступово зменшується протягом усього навчання. Така тенденція свідчить про коректну роботу процесу навчання та поступове покращення якості прогнозів моделі.

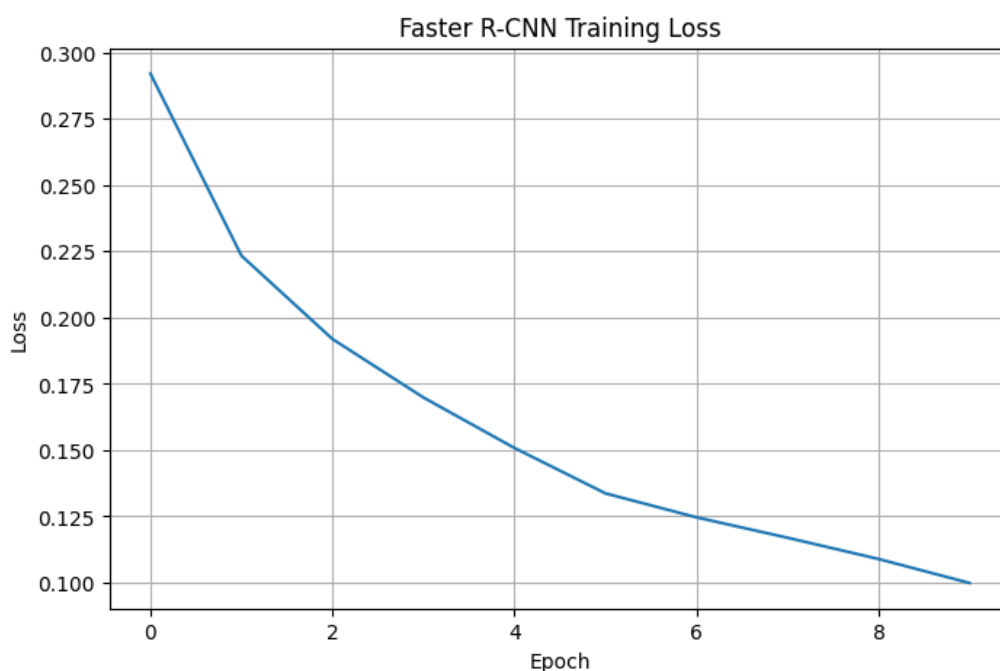


**Рисунок 3.7** – Крива зміни функції втрат моделі Faster R-CNN

### 3.4.3 Оцінювання якості моделі

Після завершення навчання було виконано оцінювання моделі на валідаційній вибірці. Для оцінки використовувалися стандартні метрики object detection: Precision, Recall, mAP50 та mAP50–95 (таб.3.10).

**Таблиця 3.10** – Результати оцінювання моделі Faster R-CNN

| <i>Метрика</i> | <i>Значення</i> |
|----------------|-----------------|
| Precision      | 0.635           |
| Recall         | 0.5792          |
| mAP50          | 0.8173          |
| mAP50-95       | 0.5247          |

Результати оцінювання свідчать про достатньо високу ефективність моделі Faster R-CNN. Найкраще значення отримано для метрики mAP50, яка

становила 0,8173. При використанні суворіших порогів IoU значення mAP50–95 зменшилося до 0,5247, що є очікуваним для задач детекції об'єктів.

Значення Precision дорівнює 0,635, а Recall — 0,5792. Це означає, що модель досить точно визначає знайдені об'єкти, проте частина цільових об'єктів залишається невиявленою. Час обробки одного зображення становить 12,19 мс, що дозволяє використовувати модель для практичних задач аналізу рентгенівських зображень.

Загалом Faster R-CNN демонструє хорошу якість детекції, хоча показник Recall свідчить про можливість подальшого покращення повноти виявлення об'єктів.

#### *3.4.4 Аналіз результатів за окремими класами*

Окрім загальних метрик, було проаналізовано якість розпізнавання для кожної категорії об'єктів окремо. Для цього використано нормалізовану матрицю помилок, яка показує, наскільки часто модель правильно визначає клас об'єкта та між якими категоріями виникає найбільше помил

На рисунку 3.8 наведено нормалізовану матрицю помило для моделі Faster R-CNN

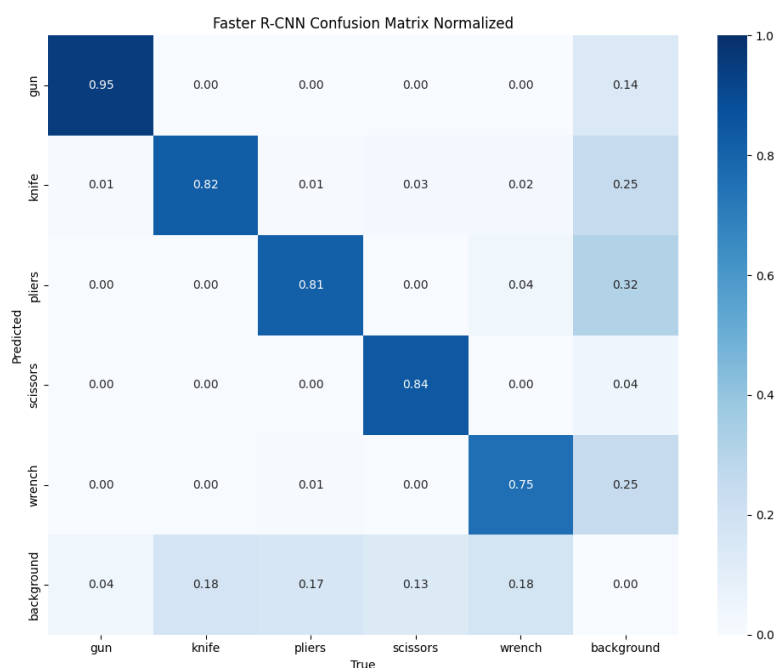


**Рисунок 3.8** – Нормалізована матриця помилок моделі Faster R-CNN

Матриця помилок свідчить про високу якість класифікації для всіх досліджуваних категорій. Найкращий результат отримано для класу gun — частка правильних класифікацій становить 0,95. Високі показники також продемонстрували класи scissors (0,84), knife (0,82) та pliers (0,81).

Найнижче значення зафіксовано для класу wrench (0,75). Попри це, якість класифікації залишається достатньо високою. Нижчий результат може бути пов'язаний зі схожістю гайкових ключів з іншими металевими предметами на рентгенівських зображеннях.

Кількість міжкласових помилок є невеликою. Найчастіше модель плутає окремі об'єкти класу knife з scissors або wrench, а для класу pliers спостерігаються поодинокі помилки класифікації як wrench. Частка таких випадків залишається низькою.

Аналіз категорії background показує, що основні труднощі пов'язані з пропуском частини об'єктів. Найбільша частка невиявлених предметів спостерігається для класів knife, pliers та wrench (близько 17–18%). Для класу

гип цей показник становить лише 4%, що свідчить про надійне виявлення найбільш критичної категорії об'єктів.

Загалом Faster R-CNN демонструє високу якість класифікації, а більшість помилок пов'язана саме з пропуском окремих об'єктів, а не з неправильним визначенням їх класу. Це узгоджується з нижчим значенням Recall порівняно з моделлю YOLOv8n.

### 3.4.5 Аналіз результатів детекції

Для якісного аналізу роботи моделі було досліджено результати детекції на окремих тестових зображеннях (рис. 3.9).

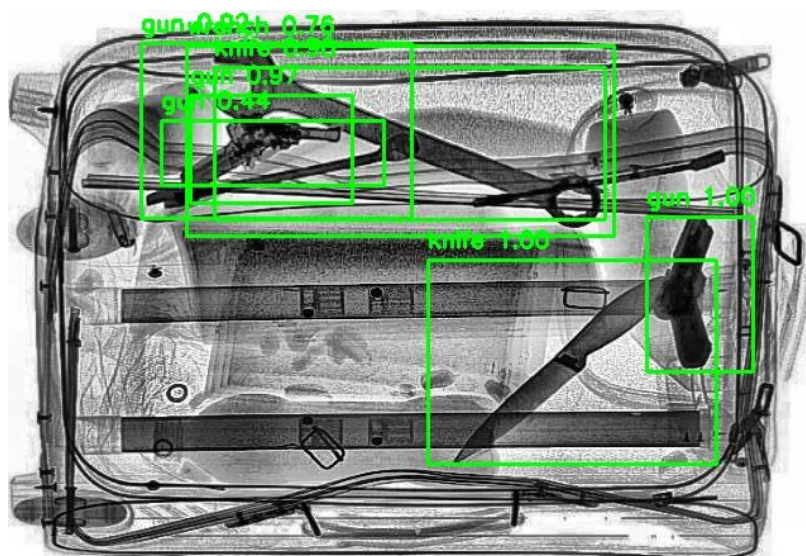


**Рисунок 3.9** – Приклад виявлення заборонених об'єктів моделлю Faster R-CNN

Тестові приклади показали, що Faster R-CNN може коректно виявляти кілька небезпечних предметів на одному рентгенівському зображенні.

Найбільше труднощів для моделі створюють перекриття об'єктів, низька контрастність та схожість окремих предметів між собою.

Хоча за сукупністю показників Faster R-CNN поступається сучасним моделям, її включення до дослідження є виправданим. Вона представляє клас двоетапних детекторів і слугує хорошою базою для порівняння з YOLOv8n та Deformable DETR.

### *Висновки до підрозділу 3.4*

Дослідження показало, що Faster R-CNN здатна розпізнавати заборонені предмети на рентгенівських зображеннях із достатньо високою точністю. Під час навчання значення функції втрат поступово зменшувалося, що вказує на стабільну роботу моделі. За більшістю показників Faster R-CNN продемонструвала хороші результати, однак за повнотою виявлення об'єктів поступилася YOLOv8n. Крім того, для її роботи потрібно більше обчислювальних ресурсів, що також важливо враховувати під час вибору моделі для практичного використання.

## **3.5 Результати навчання моделі DETR**

### *3.5.1 Налаштування та параметри навчання моделі DETR*

Для дослідження можливостей трансформерних архітектур у задачі виявлення заборонених предметів на рентгенівських зображеннях було використано модель DETR (Detection Transformer). На відміну від класичних детекторів об'єктів, DETR виконує виявлення об'єктів як задачу прямого

прогнозування набору об'єктів на зображенні без використання механізмів генерації регіонів-кандидатів та процедур придушення немаксимумів.

На рисунку 3.10 зображена архітектура моделі DETR.

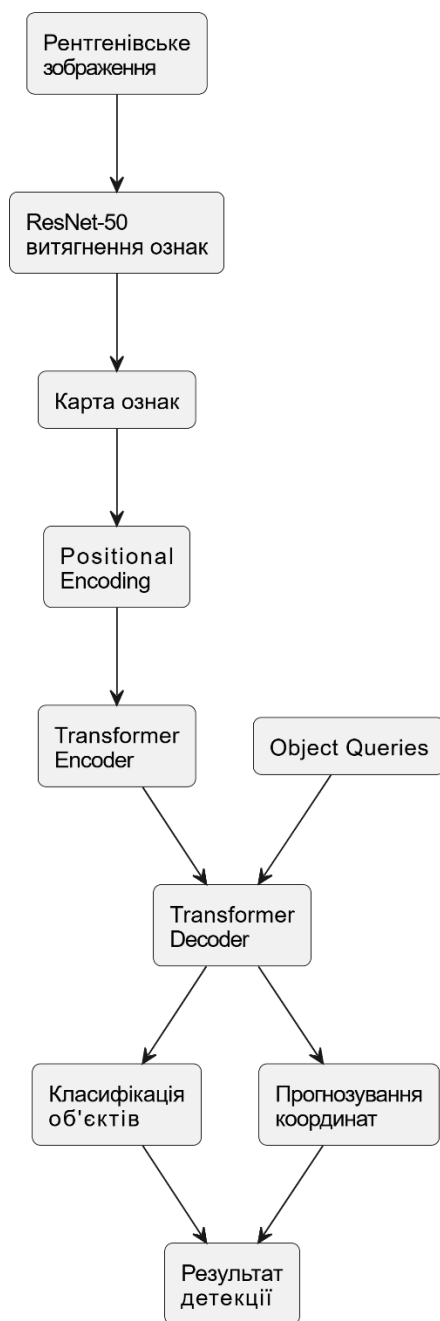


**Рисунок 3.10** – Архітектура моделі DETR

Архітектура DETR поєднує згорткову нейронну мережу для вилучення ознак із трансформерним блоком, який аналізує глобальні взаємозв'язки між усіма областями зображення. Такий підхід дозволяє враховувати контекст сцени та покращує якість локалізації об'єктів.

Навчання моделі виконувалося у середовищі Google Colaboratory із використанням графічного прискорювача NVIDIA A100. Для забезпечення коректного порівняння результатів усі експерименти проводилися на тому самому наборі даних SIXray, який використовувався для моделей YOLOv8n та Faster R-CNN. Всі параметри навчання наведені у таблиці 3.11.

**Таблиця 3.11** – Параметри навчання моделі DETR

| <i>Параметр</i>    | <i>Значення</i>    |
|--------------------|--------------------|
| Архітектура        | Deformable DETR    |
| Backbone           | ResNet-50          |
| Попереднє навчання | COCO               |
| Розмір зображення  | 640×640 пікселів   |
| Кількість епох     | 15                 |
| Batch size         | 4                  |
| Optimizer          | AdamW              |
| Learning rate      | $2 \times 10^{-5}$ |
| Weight decay       | $1 \times 10^{-4}$ |
| Пристрій навчання  | NVIDIA A100        |
| Обсяг відеопам'яті | 16 ГБ              |
| Набір даних        | SIXray             |
| Кількість класів   | 5                  |

### 3.5.2 Аналіз процесу навчання

Під час навчання здійснювався контроль значення функції втрат, яка характеризує якість локалізації об'єктів та правильність їх класифікації. Аналіз кривої навчання дозволяє оцінити стабільність процесу оптимізації та ступінь збіжності моделі.

Як видно з рисунка 3.11, у процесі навчання спостерігається поступове зменшення значення функції втрат.

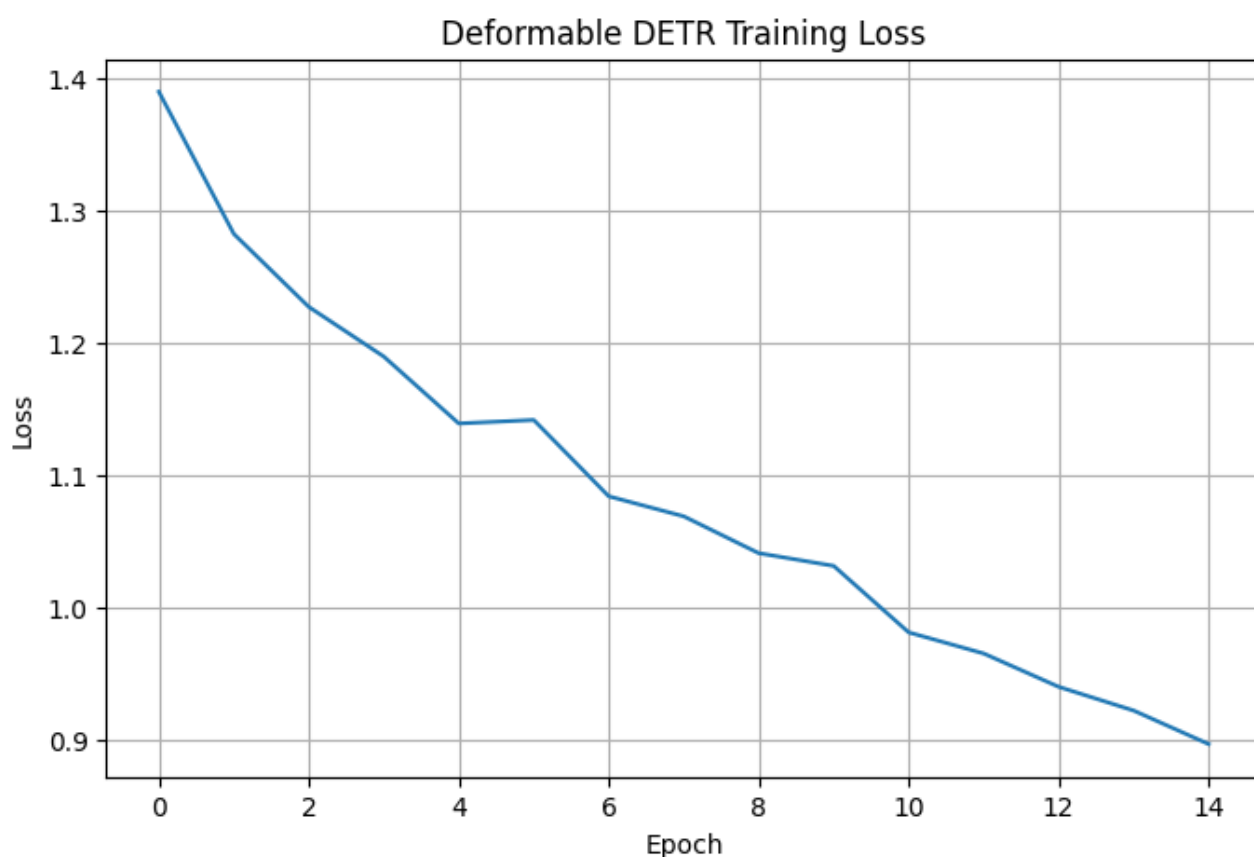


**Рисунок 3.11** – Крива зміни функції втрат моделі DETR

Відсутність різких коливань свідчить про стабільний процес оптимізації параметрів моделі. Наприкінці навчання крива наближається до плато, що

вказує на досягнення збіжності та завершення процесу адаптації моделі до навчального набору даних.

Отримані результати свідчать про успішне навчання моделі та можливість її подальшого використання для задачі автоматизованого виявлення заборонених предметів на рентгенівських зображеннях.

### 3.5.3 Оцінювання якості моделі

Після завершення навчання було проведено оцінювання моделі на валідаційній вибірці. Для оцінювання використовувалися стандартні метрики задачі виявлення об'єктів: Precision, Recall, mAP50 та mAP50–95 (табл. 3.12).

**Таблиця 3.12** – Результати оцінювання моделі DETR

| <i>Метрика</i> | <i>Значення</i> |
|----------------|-----------------|
| Precision      | 0.635           |
| Recall         | 0.3624          |
| mAP50          | 0.4893          |
| mAP50-95       | 0.2589          |

Результати оцінювання моделі DETR наведено в таблиці 3.12. Отримане значення Precision становить 0,635, що свідчить про достатньо високу точність класифікації виявлених об'єктів. Це означає, що більшість об'єктів, які модель визначає як заборонені предмети, дійсно належать до відповідних класів.

Значення Recall дорівнює 0,362, що є помітно нижчим за Precision. Такий результат вказує на наявність значної кількості пропущених об'єктів, тобто модель не виявляє частину заборонених предметів, присутніх на рентгенівських зображеннях. Це може бути пов'язано зі складністю сцени, перекриттям предметів або недостатньою кількістю епох навчання.

Показник mAP50 становить 0,489, що характеризує загальну якість локалізації та класифікації об'єктів при порозі IoU 0,5. Водночас значення mAP50–95 дорівнює 0,259, що свідчить про зниження точності моделі при використанні більш суворих критеріїв оцінювання локалізації.

Загалом модель DETR продемонструвала задовільні результати детекції заборонених предметів, однак отримані метрики свідчать про необхідність подальшого налаштування гіперпараметрів та збільшення тривалості навчання для покращення повноти виявлення об'єктів.

#### *3.5.4 Аналіз матриці помилок моделі DETR*

Для детальнішого дослідження роботи моделі було побудовано нормалізовану матрицю помилок, яка дозволяє оцінити якість класифікації кожного класу окремо та визначити найбільш проблемні категорії об'єктів (рис. 3.12)

Аналіз матриці помилок показує, що найкращі результати модель демонструє для класу «gun», для якого частка правильних класифікацій становить 0,86. Високі показники також отримано для класів «pliers» (0,63) та «knife» (0,59). Найскладнішим для моделі виявився клас «wrench», для якого частка правильних класифікацій становить лише 0,33.

Помітна кількість помилок пов'язана з пропуском об'єктів, що відображається у взаємодії класів із категорією background. Найбільша частка невиявлених об'єктів спостерігається для класів «scissors» та «wrench», що свідчить про складність їх виявлення на рентгенівських зображеннях.

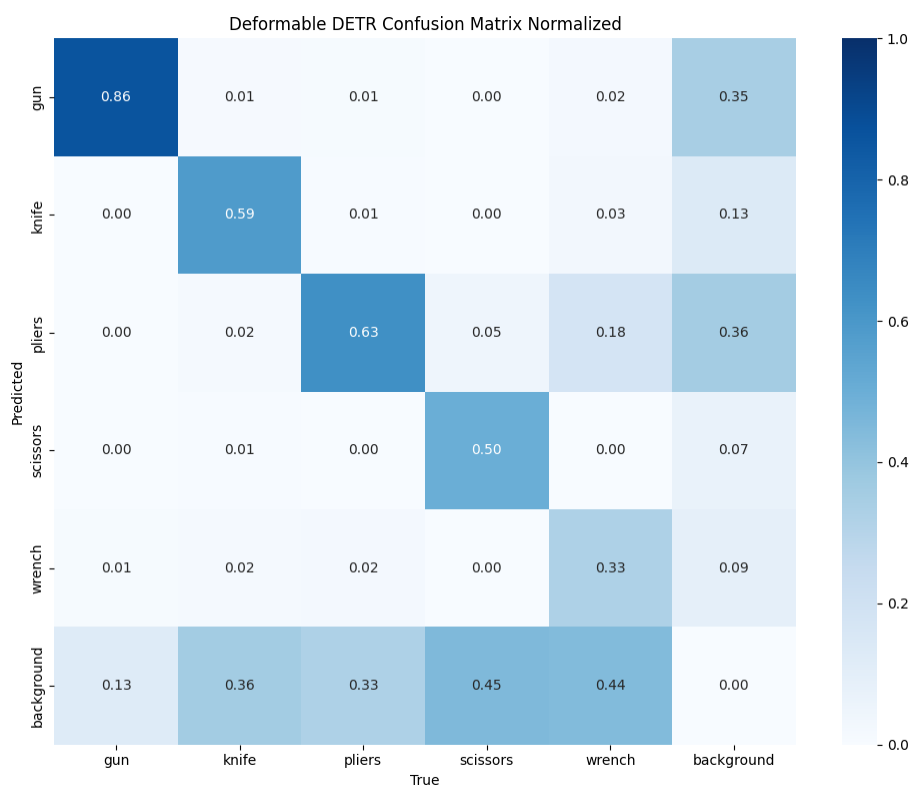


**Рисунок 4.12** – Нормалізована матриця помилок моделі DETR

Загалом матриця помилок підтверджує здатність моделі ефективно класифікувати виявлені об'єкти, проте вказує на наявність труднощів із повнотою їх виявлення для окремих категорій предметів.

### 3.5.5 Аналіз результатів детекції

Для якісного оцінювання роботи моделі було проаналізовано результати детекції на тестових рентгенівських зображеннях. На рисунку 3.13 зображено приклад роботи детекції заборонених об'єктів за допомогою моделі DETR.

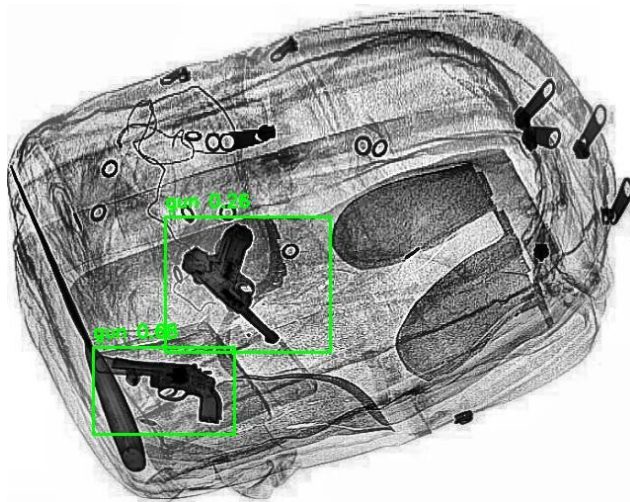


**Рисунок 3.13** – Перший приклад детекції заборонених об'єктів моделлю DETR

DETR коректно виявляє два об'єкти класу «gun» та правильно визначає їх розташування на рентгенівському зображенні. Незважаючи на відносно невисокі значення confidence score, модель успішно локалізує обидва об'єкти, що підтверджує її здатність працювати зі складними рентгенівськими сценами. Проте, модель взагалі не визначила об'єкт типу «knife» на зображенні, можна припустити, що через положення об'єкта на знімку.

На рисунку 3.14 представлено приклад характерної особливості моделі DETR – дублювання прогнозів для одного й того самого об'єкта.

Модель формує декілька обмежувальних рамок для схожих областей зображення, що призводить до повторного виявлення одного предмета. Така поведінка пов'язана з використанням механізму object queries, у якому декілька запитів можуть одночасно відповідати одному об'єкту. Крім того, спостерігаються випадки неточної локалізації, коли межі об'єкта визначаються лише частково.

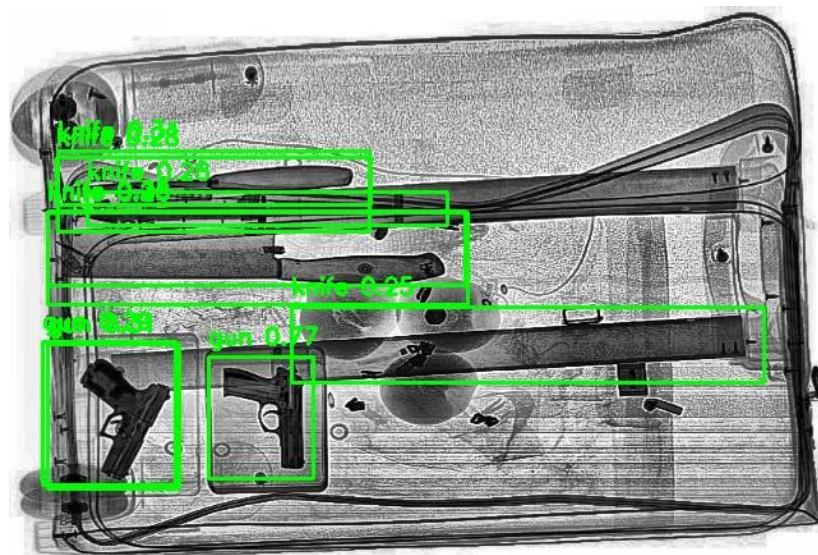


**Рисунок 3.14** – Приклад дублювання детекції та часткової локалізації об'єктів моделлю DETR

Загалом результати детекції демонструють, що модель DETR здатна успішно виявляти основні категорії заборонених предметів на рентгенівських зображеннях. Водночас аналіз прикладів показав наявність двох основних типів помилок: пропуск частини об'єктів та формування дубльованих прогнозів для одного й того самого предмета. Саме ці особливості частково пояснюють нижчі значення Recall та mAP порівняно з моделлю YOLOv8n. Незважаючи на це, DETR демонструє здатність до одночасного аналізу складних сцен та може розглядатися як перспективний підхід для задач автоматизованого контролю багажу.

### *Висновки до підрозділу 3.5*

Результати експериментів показали, що модель Deformable DETR може використовуватися для виявлення заборонених предметів на рентгенівських зображеннях, проте її ефективність виявилася нижчою порівняно з іншими

досліджуваними підходами. У процесі навчання модель стабільно збіжувалася, однак під час тестування частіше пропускала об'єкти та іноді формувала дубльовані детекції. Аналіз матриці помилок допоміг визначити категорії предметів, які розпізнавалися найкраще, а також виявити основні причини помилок. Отримані результати були враховані під час подальшого порівняння всіх трьох моделей.

### 3.6 Порівняльний аналіз моделей YOLOv8n, Faster R-CNN та DETR

#### 3.6.1 Порівняння кількісних показників моделей

Після проведення експериментів із трьома архітектурами виявлення об'єктів було виконано порівняльний аналіз отриманих результатів. Метою дослідження є визначення найбільш ефективної моделі для задачі автоматизованого виявлення заборонених предметів на рентгенівських зображеннях багажу авіапасажирів.

Для забезпечення коректності порівняння всі моделі навчалися на одному наборі даних та оцінювалися за однаковими критеріями. Основними показниками якості обрано метрики mAP50, mAP50–95 та Recall. Дані метрики дозволяють оцінити як точність локалізації об'єктів, так і повноту їх виявлення.

У таблиці 3.13 наведено результати оцінювання досліджуваних моделей.

**Таблиця 3.13** – Порівняння результати оцінювання дослідження моделей

| <i>Модель</i>   | <i>mAP50</i> | <i>mAP50-95</i> | <i>Recall</i> |
|-----------------|--------------|-----------------|---------------|
| YOLOv8n         | 0.810        | 0.580           | 0.623         |
| Faster R-CNN    | 0.817        | 0.525           | 0.579         |
| Deformable DETR | 0.489        | 0.259           | 0.362         |

Аналіз результатів показує, що найвище значення mAP50 продемонструвала модель Faster R-CNN. Проте її перевага над YOLOv8n є незначною та становить менше одного відсотка. Водночас за показником mAP50–95 найкращий результат отримала модель YOLOv8n, що свідчить про більш точну локалізацію об'єктів при використанні суворіших критеріїв оцінювання.

Особливо важливою для систем контролю багажу є метрика Recall, яка характеризує здатність моделі знаходити всі наявні небезпечні предмети. Найкращий результат також продемонструвала YOLOv8n. Це означає, що дана модель пропускає менше потенційно небезпечних об'єктів порівняно з альтернативними підходами.

Найнижчі результати за всіма досліджуваними метриками показала модель Deformable DETR. Отримані значення свідчать про те, що трансформерна архітектура в умовах проведеного експерименту поступається класичним підходам до виявлення об'єктів.

### *3.6.2 Аналіз отриманих результатів*

Для більш наочного представлення результатів порівняння було побудовано графічні залежності основних показників якості. Використання діаграм дозволяє швидше оцінити співвідношення між моделями та визначити сильні й слабкі сторони кожного підходу.

Першим показником, який було проаналізовано, є метрика mAP50. Вона характеризує загальну якість виявлення та класифікації об'єктів при порозі IoU 0,5.

Як видно з рисунка 3.15, найвищий результат продемонструвала модель Faster R-CNN. Проте відрив від YOLOv8n є мінімальним. Водночас модель

Deformable DETR демонструє суттєво нижчу якість детекції, що підтверджує результати, отримані під час аналізу числових показників.

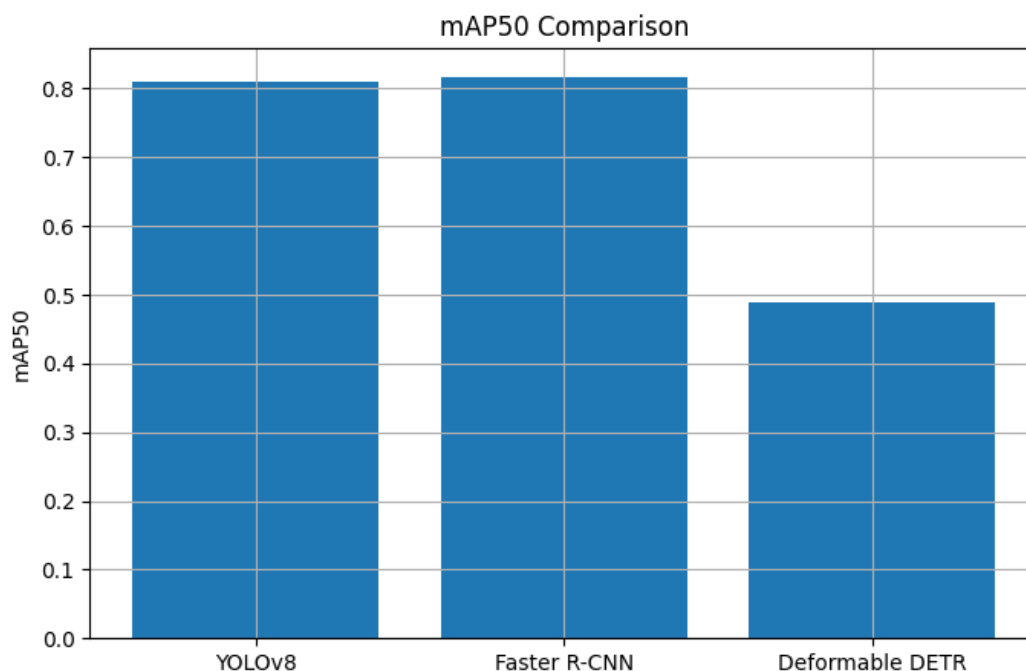


**Рисунок 3.15** – Порівняння моделей за показником mAP50

Для більш детального оцінювання точності локалізації було проаналізовано метрику mAP50–95, яка враховує широкий діапазон порогів IoU та є значно суворішою за mAP50 (рис. 3.16).

З рисунка 3.16 видно, що за даним показником лідером стала модель YOLOv8n. Це свідчить про більш точне визначення меж об'єктів та кращу локалізацію заборонених предметів на рентгенівських зображеннях. Faster R-CNN демонструє дещо нижчий результат, тоді як Deformable DETR суттєво поступається обом конкурентам.

Окрему увагу було приділено аналізу метрики Recall, яка є критично важливою для систем безпеки.

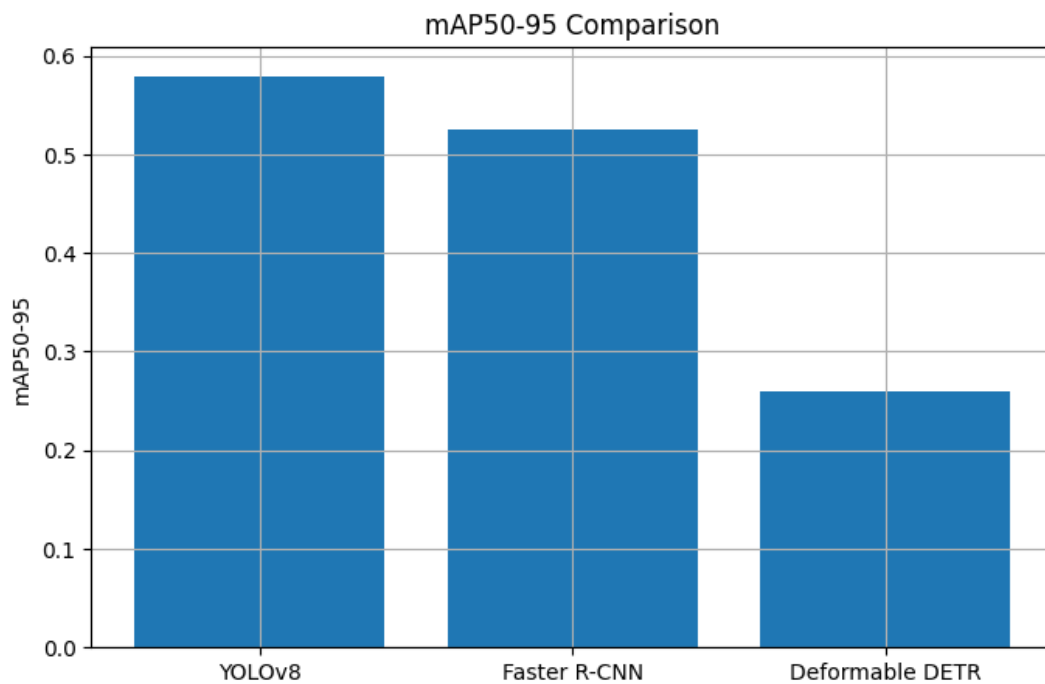


**Рисунок 3.16** – Порівняння моделей показником mAP50-95

На рисунку 3.17 наведена порівняльну діаграму метрики Recall для трьох обраних моделей.

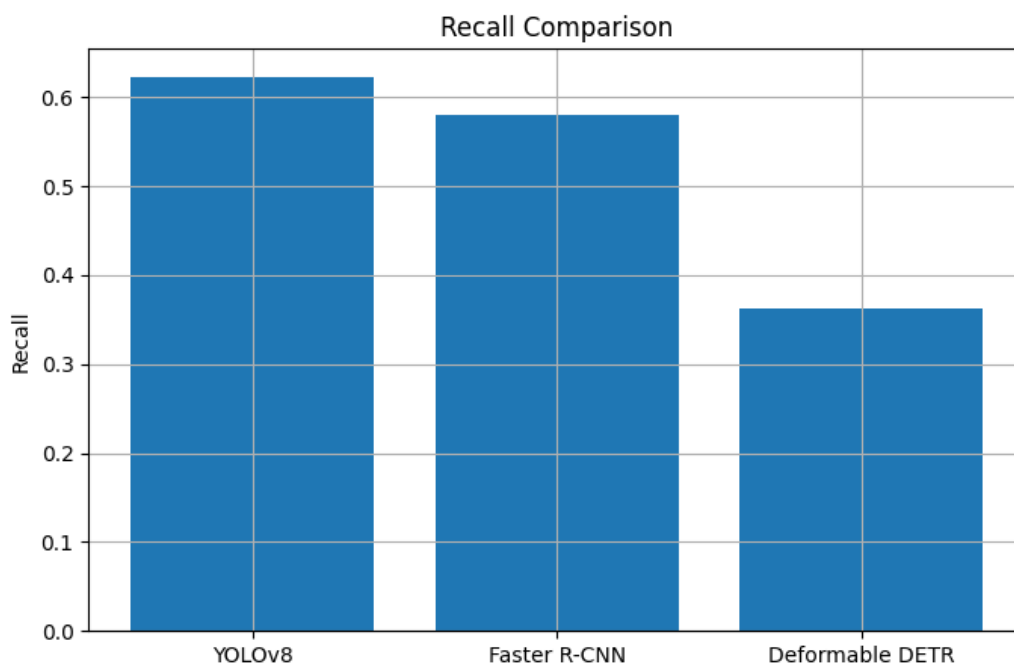


**Рисунок 3.17** – Порівняння моделей за показником Recall

Результати показують, що модель YOLOv8n забезпечує найвищу повноту виявлення об'єктів. Це означає, що вона рідше пропускає небезпечні предмети під час аналізу багажу. Faster R-CNN продемонструвала близький результат, тоді як Deformable DETR виявила значно меншу кількість об'єктів, що підтверджується її низьким значенням Recall.

Загалом результати графічного аналізу повністю підтверджують висновки, отримані під час порівняння кількісних показників. Найбільш збалансовані результати демонструє модель YOLOv8n, яка забезпечує високі показники якості за всіма досліджуваними критеріями.

### 3.6.3 Переваги та недоліки досліджуваних моделей

Окрім кількісних показників, важливим етапом дослідження є якісний аналіз особливостей кожної архітектури. Під час проведення експериментів були виявлені характерні переваги та недоліки моделей, які впливають на доцільність їх використання в системах автоматизованого контролю багажу.

У таблиці 3.14 наведено узагальнені результати такого аналізу.

**Таблиця 3.14** – Переваги та недоліки досліджуваних моделей

| <i>Модель</i>   | <i>Переваги</i>                                                         | <i>Недоліки</i>                                       |
|-----------------|-------------------------------------------------------------------------|-------------------------------------------------------|
| YOLOv8n         | Висока точність локалізації, найкращий Recall, стабільність результатів | Незначне відставання від Faster R-CNN за mAP50        |
| Faster R-CNN    | Найвищий mAP50, хороша якість локалізації                               | Нижчий Recall та складніша архітектура                |
| Deformable DETR | Сучасна трансформерна архітектура, глобальний аналіз сцени              | Низький Recall, дублювання детекцій, пропуск об'єктів |

Проведений аналіз показав, що YOLOv8n демонструє найбільш збалансоване поєднання точності та повноти виявлення об'єктів. Faster R-CNN забезпечує високі показники якості локалізації, проте поступається за здатністю знаходити всі потенційно небезпечні предмети. Deformable DETR, незважаючи на використання сучасного трансформерного підходу, у проведеному дослідженні показала найнижчу ефективність.

#### *3.6.4 Вибір моделі для реалізації інтелектуальної системи*

На основі проведеного порівняльного аналізу для реалізації інтелектуальної системи виявлення заборонених предметів у багажі авіапасажирів було обрано модель YOLOv8n.

Хоча Faster R-CNN продемонструвала дещо вищий показник mAP50, модель YOLOv8n показала найкращі результати за метриками mAP50–95 та Recall. Це свідчить про вищу точність локалізації об'єктів та кращу здатність виявляти заборонені предмети на рентгенівських зображеннях.

Крім того, під час аналізу результатів детекції YOLOv8n продемонструвала більш стабільну роботу та меншу кількість помилкових спрацьовувань порівняно з іншими досліджуваними моделями. Сукупність отриманих результатів дозволяє зробити висновок про доцільність використання саме YOLOv8n як основного модуля детекції в розробленій інтелектуальній системі.

### 3.7 Реалізація користувацького інтерфейсу системи

#### 3.7.1 Архітектура програмного інтерфейсу

Після завершення етапу навчання та оцінювання моделей було реалізовано програмний інтерфейс інтелектуальної системи виявлення заборонених предметів у багажі авіапасажирів. Основною метою інтерфейсу є забезпечення зручної взаємодії користувача з розробленою системою та надання можливості виконувати аналіз рентгенівських зображень без необхідності безпосередньої роботи з програмним кодом.

Інтерфейс реалізовано у вигляді веб-застосунку з використанням бібліотеки Gradio. Такий підхід дозволяє швидко розгортати програмні рішення для моделей машинного навчання та забезпечує доступ до функціоналу системи через стандартний веб-браузер.

Структура програмного інтерфейсу наведена на рисунку 3.18.

Робота системи здійснюється за таким принципом: користувач завантажує одне або декілька рентгенівських зображень, після чого обрана модель детекції виконує аналіз вхідних даних. Отримані результати передаються до модуля візуалізації, де формуються обмежувальні рамки навколо виявлених об'єктів, визначаються їх класи та рівень впевненості моделі. Після завершення обробки результати відображаються користувачу у графічному та табличному вигляді.

З метою проведення експериментального дослідження в інтерфейсі передбачено можливість вибору однієї з трьох реалізованих моделей: YOLOv8n, Faster R-CNN або Deformable DETR. Це дозволяє виконувати безпосереднє порівняння результатів детекції для різних архітектур у межах одного програмного середовища.

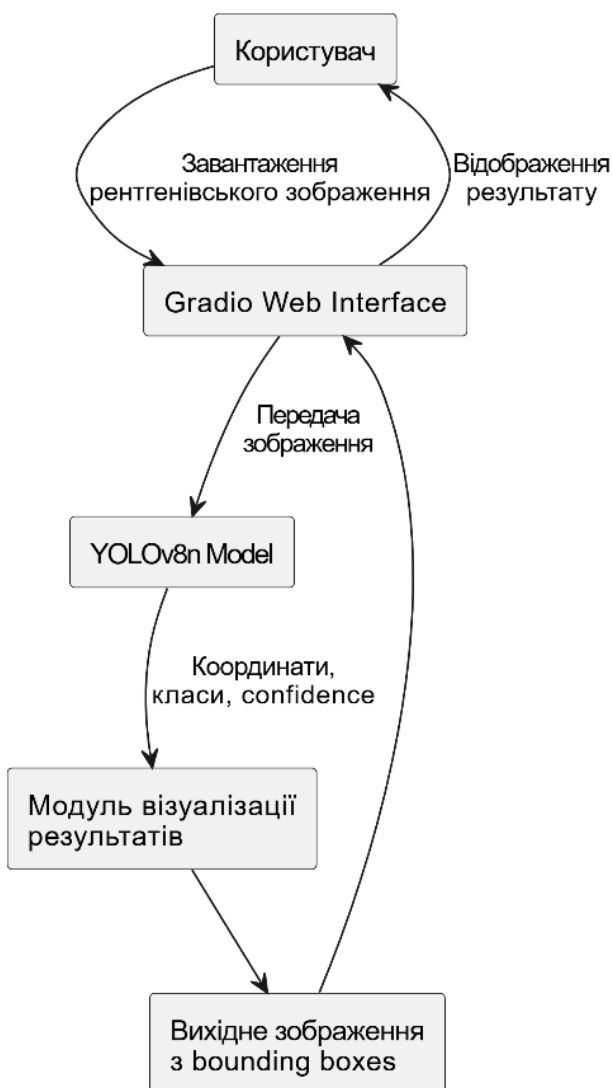


**Рисунок 3.18** – Архітектура програмного інтерфейсу системи

### 3.7.2 Реалізація веб-інтерфейсу засобами Gradio

Для реалізації користувацького інтерфейсу було використано бібліотеку Gradio, яка забезпечує створення інтерактивних веб-застосунків для моделей машинного навчання без необхідності розробки окремої серверної частини.

Використання Gradio дозволило інтегрувати модулі детекції об'єктів та засоби візуалізації результатів у єдину систему.

Загальний вигляд головного вікна програмного інтерфейсу наведено на рисунку 3.19.

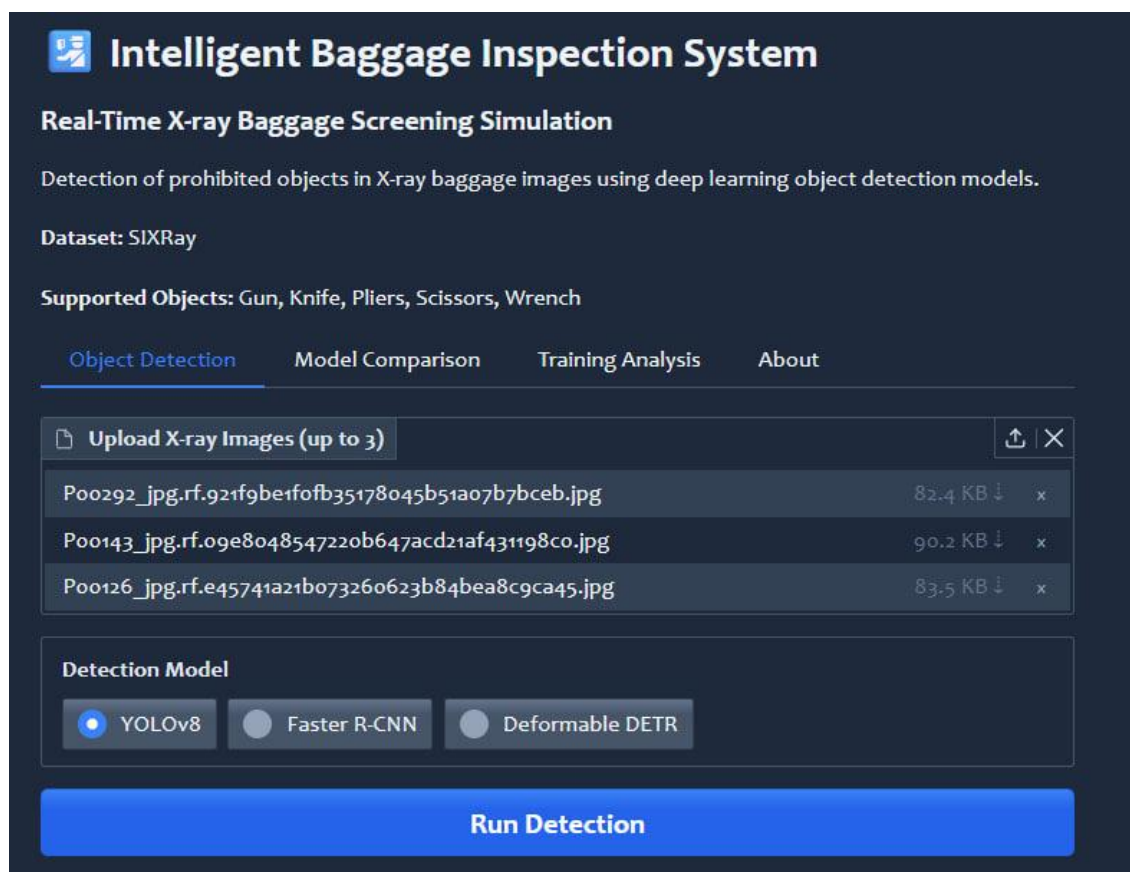


**Рисунок 3.19** – Головне вікно програмного інтерфейсу

Інтерфейс містить область завантаження рентгенівських зображень, блок вибору моделі детекції та кнопку запуску аналізу. Користувач може завантажити до трьох зображень одночасно та обрати модель, яка буде використовуватися для їх обробки.

Після запуску процедури аналізу система автоматично передає вхідні дані до відповідної моделі детекції та виконує прогнозування. Результати роботи моделі відображаються у вигляді зображень з нанесеними обмежувальними рамками та підписами виявлених класів.

Приклад роботи системи після виконання аналізу наведено на рисунку 3.20.

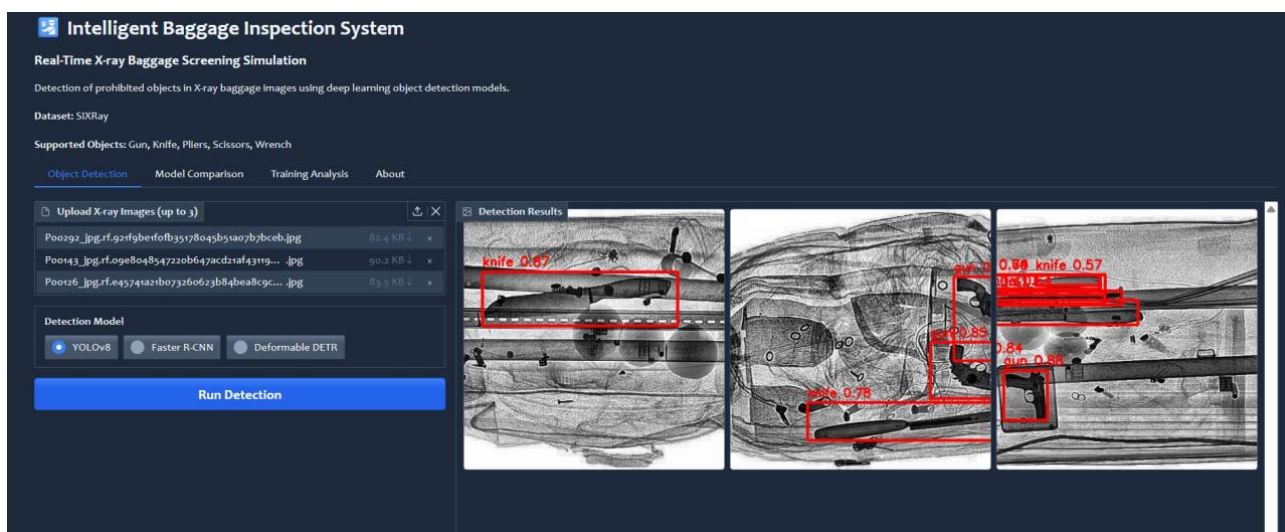


**Рисунок 3.20** – Результати детекції заборонених об'єктів

Для кожного виявленого об'єкта система відображає назву класу та відповідне значення впевненості моделі. Такий спосіб подання результатів дозволяє швидко оцінити характер виявлених предметів та рівень достовірності прогнозу.

Крім графічного представлення результатів, інтерфейс формує таблицю виявлених об'єктів із зазначенням рівня впевненості моделі. Приклад відповідного подання результатів наведено на рисунку 3.21.

На основі отриманих результатів система автоматично визначає факт наявності потенційно небезпечних предметів та формує відповідне повідомлення про рівень загрози. Такий підхід дозволяє підвищити інформативність результатів та спростити процес прийняття рішень оператором.

| Detected Objects |            |
|------------------|------------|
| Object           | Confidence |
| knife            | 0.874      |
| gun              | 0.879      |
| gun              | 0.849      |
| knife            | 0.78       |
| gun              | 0.884      |
| gun              | 0.842      |
| knife            | 0.773      |
| knife            | 0.696      |
| knife            | 0.645      |
| knife            | 0.643      |
| knife            | 0.574      |
| knife            | 0.405      |

| Detected Objects Count |  |
|------------------------|--|
| 12                     |  |

Threat Detected

Threat Level: HIGH

**Рисунок 3.21** – Табличне представлення результатів детекції

### 3.7.3 Аналіз роботи розробленої системи

Окрім функцій детекції об'єктів, розроблений програмний інтерфейс містить додаткові засоби для аналізу результатів проведених експериментів. Зокрема, реалізовано окремий розділ для відображення метрик досліджуваних моделей та побудованих порівняльних діаграм.

Приклад сторінки з результатами порівняльного аналізу моделей наведено на рисунку 3.22.

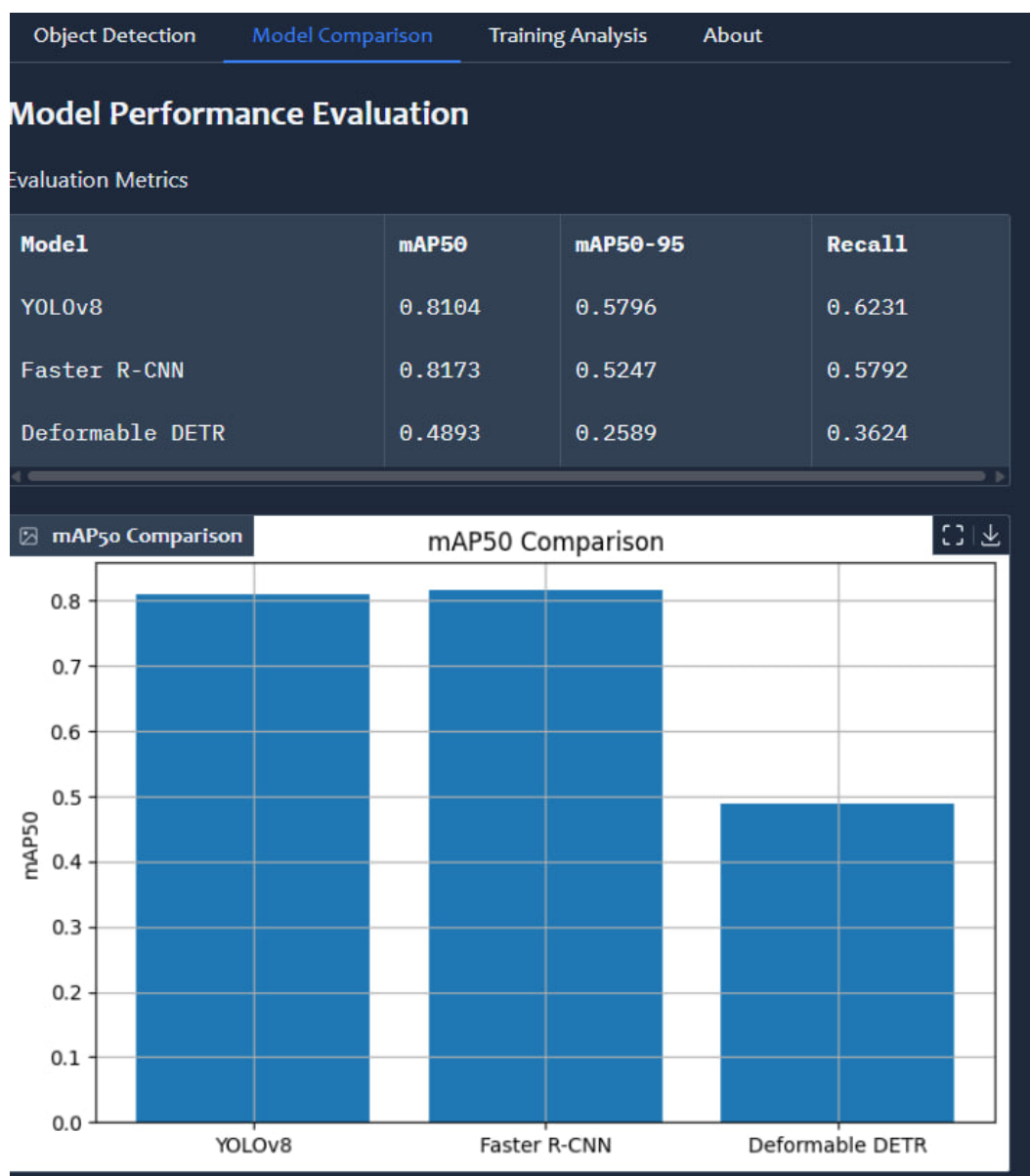


**Рисунок 3.22** – Відображення результатів порівняльного аналізу моделей

Завдяки наявності відповідного модуля користувач може ознайомитися з основними показниками якості моделей та візуально оцінити їх ефективність без необхідності використання додаткових інструментів аналізу даних.

Також у складі програмного інтерфейсу реалізовано інформаційну сторінку з відомостями про проєкт, використаний набір даних, досліджувані моделі та результати проведених експериментів.

Приклад інформаційної сторінки наведено на рисунку 3.23.

Object Detection
Model Comparison
Training Analysis
About

## Project Information

### Purpose

Development of an intelligent system for detecting prohibited objects in X-ray baggage images.

### Dataset

SIXRay Dataset

### Detected Classes

- Gun
- Knife
- Pliers
- Scissors
- Wrench

### Evaluated Models

- YOLOv8
- Faster R-CNN
- Deformable DETR

### Experimental Results

| Model           | mAP <sub>50</sub> | mAP <sub>50-95</sub> | Recall |
|-----------------|-------------------|----------------------|--------|
| YOLOv8          | 81.04%            | 57.96%               | 62.31% |
| Faster R-CNN    | 81.73%            | 52.47%               | 57.92% |
| Deformable DETR | 48.93%            | 25.89%               | 36.24% |

**Рисунок 3.23** – Інформаційна сторінка системи

Проведене тестування показало коректну роботу всіх основних компонентів програмного забезпечення. Система забезпечує повний цикл обробки рентгенівських зображень: від завантаження даних користувачем до формування результатів детекції та їх подальшого аналізу. Реалізований інтерфейс дозволяє використовувати розроблену систему без необхідності знання особливостей її внутрішньої реалізації та створює зручні умови для взаємодії з моделями машинного навчання.

### Висновки до розділу 3

У третьому розділі проведено експериментальне дослідження моделей детекції об'єктів для задачі виявлення заборонених предметів на рентгенівських зображеннях багажу. Для навчання та тестування використано датасет SIXray, який було розподілено на тренувальну, валідаційну та тестову вибірки. Перед початком експериментів зображення пройшли необхідну попередню обробку.

У середовищі Google Colaboratory досліджено три підходи до розв'язання задачі: YOLOv8n, Faster R-CNN та Deformable DETR. Для кожної моделі проаналізовано процес навчання, значення основних метрик якості та результати роботи на тестових даних.

Найкращі результати за сукупністю показників отримала модель YOLOv8n. Вона забезпечила високі значення Recall та mAP50–95, що є важливим для систем безпеки, де небажаним є пропуск потенційно небезпечних предметів. Faster R-CNN також показала хорошу якість детекції та досягла найвищого значення mAP50, однак загалом поступилася YOLOv8n. Найменш успішною виявилася модель Deformable DETR, для якої спостерігалася більша кількість пропущених об'єктів і дубльованих детекцій.

З урахуванням результатів порівняння саме YOLOv8n обрано як основну модель для розроблюваної системи. Вона продемонструвала найкраще співвідношення між точністю виявлення та повнотою знаходження об'єктів.

Окремим результатом роботи стало створення веб-застосунку з використанням бібліотеки Gradio. Розроблений інтерфейс дає змогу завантажувати рентгенівські зображення, запускати аналіз та переглядати результати детекції у зручному форматі. Під час тестування всі компоненти системи працювали коректно, що підтвердило можливість використання розробленого рішення як прототипу системи підтримки прийняття рішень під час контролю багажу.

Таким чином, у межах третього розділу виконано порівняння сучасних моделей детекції об'єктів і створено програмний прототип системи автоматичного виявлення заборонених предметів на рентгенівських зображеннях багажу авіапасажирів.

## ВИСНОВКИ

У дипломній роботі вирішено задачу розроблення інтелектуальної системи виявлення заборонених предметів у багажі авіапасажирів на основі методів комп'ютерного зору та глибокого навчання.

У процесі виконання роботи проведено аналіз сучасних підходів до автоматизованого контролю багажу, особливостей рентгенівських систем безпеки, наборів даних для дослідження рентгенівських зображень та сучасних моделей детекції об'єктів. Встановлено, що основними труднощами автоматичного аналізу X-ray зображень є перекриття предметів, наявність прихованих об'єктів, дисбаланс класів та низька контрастність окремих сцен.

На основі проведеного аналізу розроблено архітектуру інтелектуальної системи підтримки прийняття рішень, яка включає модулі попередньої обробки даних, виявлення об'єктів, оцінювання рівня загрози, формування рекомендацій та механізм ручної перевірки. Запропонована система реалізує підхід human-in-the-loop, за якого остаточне рішення щодо подальших дій приймає інспектор служби авіаційної безпеки.

Для експериментального дослідження використано набір даних SIXray. Виконано підготовку даних, налаштування процесу навчання та оцінювання моделей YOLOv8n, Faster R-CNN і Deformable DETR. Порівняння результатів здійснювалося за показниками Precision, Recall та mAP.

За результатами експериментів встановлено, що модель YOLOv8n продемонструвала найкращий баланс між точністю виявлення, повнотою детекції та швидкістю роботи. Faster R-CNN забезпечила високі показники локалізації об'єктів, проте поступилася за швидкістю. Модель Deformable DETR показала нижчу ефективність на використаному наборі даних та виявилася більш вимогливою до обчислювальних ресурсів.

Практичним результатом роботи стала реалізація програмного прототипу системи засобами бібліотеки Gradio. Розроблений вебзастосунок забезпечує

завантаження рентгенівських зображень, автоматичне виявлення заборонених предметів, візуалізацію результатів детекції та формування рекомендацій для оператора.

Отримані результати підтверджують доцільність використання сучасних моделей детекції об'єктів для автоматизації аналізу рентгенівських зображень багажу. Розроблена система може бути використана як прототип інтелектуальної системи підтримки прийняття рішень під час контролю безпеки в аеропортах.

Подальший розвиток роботи може бути пов'язаний із використанням більших наборів даних, дослідженням сучасних трансформерних архітектур, застосуванням мультимодальних підходів та підвищенням якості виявлення частково прихованих і перекритих об'єктів.

## ПЕРЕЛІК ПОСИЛАНЬ

1. Faits et chiffres. International Civil Aviation Organization. URL: <https://www.icao.int/fr/faits-et-chiffres> (дата звернення: 17.05.2026).
2. Dangerous Goods Guidance for Passengers. IATA. URL: <http://iata.org/en/programs/cargo/dgr/dgr-guidance-passengers/> (дата звернення: 17.05.2026).
3. Lu L. How Does an X-Ray Machine Work? LinkedIn. URL: <https://www.linkedin.com/pulse/how-does-x-ray-machine-work-lucas-lu-office-> (дата звернення: 18.05.2026).
4. Material Identification in Presence of Metal for Baggage Screening. PubMed. URL: <https://pubmed.ncbi.nlm.nih.gov/32390647/> (дата звернення: 18.05.2026).
5. Material Identification in Presence of Metal for Baggage Screening. Boston University Blogs. URL: [https://blogs.bu.edu/sanda/files/2020/08/Material\\_Identification\\_in\\_Presence\\_of\\_Metal\\_for\\_Baggage\\_Screening\\_final.pdf](https://blogs.bu.edu/sanda/files/2020/08/Material_Identification_in_Presence_of_Metal_for_Baggage_Screening_final.pdf) (дата звернення: 19.05.2026).
6. DEBISim Simulation Pipeline. Purdue University. URL: [https://engineering.purdue.edu/RVL/Publications/AManerikar\\_DEBISim\\_Simulation\\_Pipeline\\_2021\\_JOXST.pdf](https://engineering.purdue.edu/RVL/Publications/AManerikar_DEBISim_Simulation_Pipeline_2021_JOXST.pdf) (дата звернення: 19.05.2026).
7. What Do Airport X-Ray Scanners Actually Show? ScienceInsights. URL: <https://scienceinsights.org/what-do-airport-x-ray-scanners-actually-show> (дата звернення: 20.05.2026).
8. Using Eye Movements to Understand How Security Screeners Search for Threats in X-Ray Baggage. URL: <https://www.mdpi.com/2411-5150/3/2/24> (дата звернення: 20.05.2026).

9. Andriyanov N. Deep Learning for Detecting Dangerous Objects in X-rays of Luggage. URL: <https://www.mdpi.com/2673-4591/33/1/20> (дата звернення: 21.05.2026).
10. SIXray: A Large-scale Security Inspection X-ray Benchmark for Prohibited Item Discovery in Overlapping Images. arXiv. URL: <https://arxiv.org/abs/1901.00303> (дата звернення: 21.05.2026).
11. SIXray: A Large-scale Security Inspection X-ray Benchmark for Prohibited Item Discovery in Overlapping Images. CVPR Open Access. URL: [https://openaccess.thecvf.com/content\\_CVPR\\_2019/papers/Miao\\_SIXray\\_A\\_Large-Scale\\_Security\\_Inspection\\_X-Ray\\_Benchmark\\_for\\_Prohibited\\_Item\\_CVPR\\_2019\\_paper.pdf](https://openaccess.thecvf.com/content_CVPR_2019/papers/Miao_SIXray_A_Large-Scale_Security_Inspection_X-Ray_Benchmark_for_Prohibited_Item_CVPR_2019_paper.pdf) (дата звернення: 22.05.2026).
12. Transfer Learning Using Convolutional Neural Networks for Object Classification within X-ray Baggage Security Imagery. URL: <https://breckon.org/toby/publications/papers/akcay16transfer.pdf> (дата звернення: 22.05.2026).
13. The Good, the Bad and the Ugly: Evaluating Convolutional Neural Networks for Prohibited Item Detection Using Real and Synthetically Composited X-ray Imagery. arXiv. URL: <https://arxiv.org/abs/1909.11508> (дата звернення: 23.05.2026).
14. DCNNs: A Transfer Learning Comparison of Full Weapon Family Threat Detection for Dual-Energy X-Ray Baggage Imagery. arXiv. URL: <https://arxiv.org/abs/2006.13065> (дата звернення: 23.05.2026).
15. Operationalizing Convolutional Neural Network Architectures for Prohibited Object Detection in X-Ray Imagery. arXiv. URL: <https://arxiv.org/abs/2110.04906> (дата звернення: 24.05.2026).
16. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. URL: [https://papers.nips.cc/paper\\_files/paper/2015/file/14bfa6bb14875e45bba028a21ed38046-Paper.pdf](https://papers.nips.cc/paper_files/paper/2015/file/14bfa6bb14875e45bba028a21ed38046-Paper.pdf) (дата звернення: 24.05.2026).

17. You Only Look Once: Unified, Real-Time Object Detection. arXiv. URL: <https://arxiv.org/abs/1506.02640> (дата звернення: 25.05.2026).
18. SSD: Single Shot MultiBox Detector. Google Research. URL: <https://research.google/pubs/ssd-single-shot-multibox-detector/> (дата звернення: 25.05.2026).
19. End-to-End Object Detection with Transformers. arXiv. URL: <https://arxiv.org/abs/2005.12872> (дата звернення: 26.05.2026).
20. Towards Real-world X-ray Security Inspection: A High-Quality Benchmark and Lateral Inhibition Module for Prohibited Items Detection. arXiv. URL: <https://arxiv.org/abs/2108.09917> (дата звернення: 26.05.2026).
21. Metrics and Scoring: Quantifying the Quality of Predictions. Scikit-learn. URL: [https://scikit-learn.org/stable/modules/model\\_evaluation.html](https://scikit-learn.org/stable/modules/model_evaluation.html) (дата звернення: 27.05.2026).
22. Intersection over Union (IoU) for Object Detection. PyImageSearch. URL: <https://pyimagesearch.com/2016/11/07/intersection-over-union-iou-for-object-detection/> (дата звернення: 27.05.2026).
23. COCO: Common Objects in Context. URL: <https://cocodataset.org/#detection-eval> (дата звернення: 28.05.2026).
24. confusion\_matrix. Scikit-learn. URL: [https://scikit-learn.org/stable/modules/generated/sklearn.metrics.confusion\\_matrix.html](https://scikit-learn.org/stable/modules/generated/sklearn.metrics.confusion_matrix.html) (дата звернення: 28.05.2026).
25. Advancements in Machine Learning Techniques for Threat Item Detection in X-ray Images: A Comprehensive Survey. SpringerLink. URL: <https://link.springer.com/article/10.1007/s13735-024-00348-2> (дата звернення: 29.05.2026).
26. Towards More Efficient Security Inspection via Deep Learning: A Task-Driven X-ray Image Cropping Scheme. PMC. URL: <https://pmc.ncbi.nlm.nih.gov/articles/PMC9031191/> (дата звернення: 29.05.2026).

27. Object Detection Models Overview. GeeksforGeeks. URL: <https://www.geeksforgeeks.org/computer-vision/object-detection-models/> (дата звернення: 30.05.2026).
28. GitHub – MeioJane/SIXray: The SIXray Dataset. URL: <https://github.com/MeioJane/SIXray> (дата звернення: 30.05.2026).
29. Akcay S., Kundegorski M., Devereux M., Breckon T. Transfer Learning Using Convolutional Neural Networks for Object Classification within X-ray Baggage Security Imagery // IEEE International Conference on Image Processing. 2016.
30. Liang K., Yang L., Wang Y. Toward Automatic Threat Recognition for Airport X-ray Baggage Screening with Deep Convolutional Object Detection // Proceedings of the International Conference on Pattern Recognition. 2018.
31. Apostolopoulos I. Deep Learning for Detecting Dangerous Objects in X-rays of Luggage // Engineering Proceedings. 2023. Vol. 33.
32. Illicit Object Detection in X-ray Imaging Using Deep Learning Techniques: A Comparative Evaluation // arXiv preprint. 2025.
33. Jaccard N., Rogers T., Morton E., Griffin L. Automated Detection of Threat Objects Using Deep Learning in Security X-ray Images // arXiv preprint. 2021.
34. Mery D. X-ray Testing and Inspection for Security Applications // Springer International Publishing. 2015.
35. Miao C., Xie L., Wan F. SIXray: A Large-scale Security Inspection X-ray Benchmark for Prohibited Item Discovery in Overlapping Images // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2019.
36. Zhao B., Zhang L., Chen Z. PIDray: A Large-Scale X-ray Benchmark for Real-World Prohibited Item Detection // International Journal of Computer Vision. 2023.
37. Wei J., Cao H., Han Y. Towards Real-world X-ray Security Inspection: A High-Quality Benchmark and Lateral Inhibition Module for Prohibited Items Detection // arXiv preprint. 2021.

38. Wei J., Zhang H., Cao H. OPIXray: Occluded Prohibited Items X-ray Benchmark for Security Inspection // arXiv preprint. 2020.
39. Mery D., Rizzo V., Zscherpel U. GDXray: The Database of X-ray Images for Nondestructive Testing // Journal of Nondestructive Evaluation. 2015.
40. Synthetic X-ray Image Generation for Security Applications // Scientific Data. Nature Publishing Group. 2025.
41. Redmon J., Farhadi A. YOLO: You Only Look Once — Unified, Real-Time Object Detection // arXiv preprint arXiv:1506.02640. 2015.
42. Carion N., Massa F., Synnaeve G. End-to-End Object Detection with Transformers // European Conference on Computer Vision (ECCV). 2020.
43. Ren S., He K., Girshick R., Sun J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks // IEEE Transactions on Pattern Analysis and Machine Intelligence. 2017.
44. Jaccard N., Rogers T., Morton E., Griffin L. Evaluation of a Dual Convolutional Neural Network Architecture for Object-wise Anomaly Detection in Cluttered X-ray Security Imagery // arXiv preprint arXiv:1904.05304. 2019.
45. Redmon J., Divvala S., Girshick R., Farhadi A. You Only Look Once: Unified, Real-Time Object Detection // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016.
46. GeeksforGeeks. Object Detection Models Overview. URL: <https://www.geeksforgeeks.org/computer-vision/object-detection-models/>
47. A Lightweight Xray-YOLO-Mamba Model for Prohibited Item Detection in X-ray Images // Scientific Reports. Nature Publishing Group. 2025.
48. Liu W., Anguelov D., Erhan D. SSD: Single Shot MultiBox Detector // European Conference on Computer Vision (ECCV). 2016.
49. Lin T.-Y., Goyal P., Girshick R. Focal Loss for Dense Object Detection // IEEE International Conference on Computer Vision (ICCV). 2017.
50. Carion N., Massa F., Synnaeve G. Detection Transformer (DETR) // ECCV. 2020.

51. Hybrid CNN-Transformer Architectures for X-ray Security Screening // arXiv preprint. 2025
52. COCO Dataset Evaluation Metrics. URL: <https://cocodataset.org/#detection-eval>
53. Comprehensive Review of AI-based Baggage Screening Systems // Scientific Reports. 2023.
54. Radiographic Image Contrast Analysis in Security X-ray Imaging // PubMed. 2020.
55. Explainable Artificial Intelligence for Safety-Critical Applications // Nature Machine Intelligence. 2021.
56. GitHub – MeioJane/SIXray: The SIXray dataset. *GitHub*. URL: <https://github.com/MeioJane/SIXray>.

## ДОДАТОК А ЛІСТИНГ КОДУ

### A.1 Імпорт бібліотек та налаштування середовища

```
import cv2
import json
import torch
import random
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from tqdm import tqdm
from pathlib import Path
from ultralytics import YOLO
import torchvision
from torchvision.models.detection import fasterrcnn_resnet50_fpn
from torch.utils.data import Dataset, DataLoader
from transformers import AutoImageProcessor, DeformableDetrForObjectDetection
from pycocotools.coco import COCO
from pycocotools.cocoeval import COCOeval
CLASSES = ['gun', 'knife', 'pliers', 'scissors', 'wrench']
NUM_CLASSES = len(CLASSES)
id2label = {i:c for i,c in enumerate(CLASSES)}
label2id = {v:k for k,v in id2label.items()}

DEVICE = torch.device(
    "cuda" if torch.cuda.is_available()
    else "cpu"
)
```

### A.2 Підготовка даних та конфігурація YOLOv8

```
from pathlib import Path
DATASET_DIR = Path('/content/dataset/sixray_v3')
YOLO_YAML = DATASET_DIR / 'data.yaml'
yaml_content = f"""
path: {DATASET_DIR}
train: train/images
val: valid/images
test: test/images
names:
  0: gun
  1: knife
  2: pliers
  3: scissors
  4: wrench
"""
with open(YOLO_YAML, 'w') as f:
    f.write(yaml_content)
print(open(YOLO_YAML).read())
```

### A.3 Навчання моделі YOLOv8

```
yolo_train_model = YOLO('yolov8n.pt')
YOLO_EPOCHS = 40
yolo_results = yolo_train_model.train(
    data=str(YOLO_YAML),
    epochs=YOLO_EPOCHS,
    imgsz=640,
    batch=16,
    device=0,
```

```

optimizer='AdamW',
lr0=1e-3,
weight_decay=1e-4,
cos_lr=True,
pretrained=True,
patience=15,
cache=True,
workers=2,
project=str(RESULTS_DIR),
name='yolo_training'
)
print('YOLO training completed')

# Load trained YOLO weights

YOLO_WEIGHTS = MODELS_DIR / 'best_yolo.pt'

yolo_model = YOLO(str(YOLO_WEIGHTS))

print('Trained YOLO loaded')

```

#### A.4 Реалізація датасету для Faster R-CNN

```

class DetectionDataset(Dataset):
    def __init__(self, image_dir, label_dir):
        self.image_dir = Path(image_dir)
        self.label_dir = Path(label_dir)
        self.images = sorted(list(self.image_dir.glob('*')))
    def __len__(self):
        return len(self.images)
    def __getitem__(self, idx):
        img_path = self.images[idx]
        image = cv2.imread(str(img_path))
        image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
        h, w = image.shape[:2]
        boxes = []
        labels = []
        label_path = self.label_dir / f'{img_path.stem}.txt'
        if label_path.exists():
            with open(label_path) as f:
                for line in f.readlines():
                    cls_id, x, y, bw, bh = map(float, line.strip().split())
                    x1 = (x - bw/2) * w
                    y1 = (y - bh/2) * h
                    x2 = (x + bw/2) * w
                    y2 = (y + bh/2) * h
                    boxes.append([x1,y1,x2,y2])
                    labels.append(int(cls_id) + 1)
        target = {
            'boxes': torch.as_tensor(boxes, dtype=torch.float32).reshape(-1, 4),
            'labels': torch.tensor(labels, dtype=torch.int64),
            'image_id': torch.tensor([idx])
        }
        image = torchvision.transforms.functional.to_tensor(image)
        return image, target
    def collate_fn(batch):
        return tuple(zip(*batch))

def collate_fn(batch):
    return tuple(zip(*batch))

train_dataset = DetectionDataset(
    DATASET_DIR / 'train/images',
    DATASET_DIR / 'train/labels'
)

```

```

)
test_dataset = DetectionDataset(
    DATASET_DIR / 'test/images',
    DATASET_DIR / 'test/labels'
)
train_loader = DataLoader(
    train_dataset,
    batch_size=4,
    shuffle=True,
    collate_fn=collate_fn
)
test_loader = DataLoader(
    test_dataset,
    batch_size=4,
    shuffle=False,
    collate_fn=collate_fn
)

```

#### A.5 Побудова та навчання Faster R-CNN

```

import torchvision
from torchvision.models.detection import fasterrcnn_resnet50_fpn
faster_rcnn = fasterrcnn_resnet50_fpn(
    weights="DEFAULT"
)
in_features = (
    faster_rcnn
    .roi_heads
    .box_predictor
    .cls_score
    .in_features
)
faster_rcnn.roi_heads.box_predictor = (
    torchvision.models.detection.faster_rcnn.FastRCNNPredictor(
        in_features,
        NUM_CLASSES + 1
    )
)
faster_rcnn.to(DEVICE)
print("Faster R-CNN created")

RCNN_EPOCHS = 10
optimizer = torch.optim.AdamW(
    faster_rcnn.parameters(),
    lr=1e-4,
    weight_decay=1e-4
)
loss_history = []
for epoch in range(RCNN_EPOCHS):
    faster_rcnn.train()
    epoch_loss = 0.0
    pbar = tqdm(train_loader)
    for images, targets in pbar:
        images = [i.to(DEVICE) for i in images]
        targets = [
            {k: v.to(DEVICE) for k, v in t.items()}
            for t in targets
        ]
        loss_dict = faster_rcnn(images, targets)
        loss = sum(loss for loss in loss_dict.values())
        optimizer.zero_grad()
        loss.backward()

```

```

optimizer.step()
epoch_loss += loss.item()
pbar.set_description(
    f"RCNN Epoch {epoch+1}/{RCNN_EPOCHS} Loss {loss.item():.4f}"
)
avg_loss = epoch_loss / len(train_loader)
loss_history.append(avg_loss)
print(
    f"Epoch {epoch+1}/{RCNN_EPOCHS} | "
    f"Average Loss: {avg_loss:.4f}"
)
# Зберігаємо після КОЖНОЇ епохи
torch.save(
    faster_rcnn.state_dict(),
    MODELS_DIR / f"faster_rcnn_epoch_{epoch+1}.pth"
)
np.save(
    RESULTS_DIR / "faster_rcnn_loss.npy",
    np.array(loss_history)
)
print("Faster R-CNN training complete")

```

#### A.6 Реалізація датасету для Deformable DETR

```

class DETRDataset(Dataset):
    def __init__(self, image_dir, label_dir, processor):
        self.image_dir = Path(image_dir)
        self.label_dir = Path(label_dir)
        self.processor = processor
        self.images = sorted(
            list(self.image_dir.glob('*'))
        )
    def __len__(self):
        return len(self.images)
    def __getitem__(self, idx):
        img_path = self.images[idx]
        image = cv2.imread(str(img_path))
        image = cv2.cvtColor(
            image,
            cv2.COLOR_BGR2RGB
        )
        h, w = image.shape[:2]
        annotations = []
        label_path = (
            self.label_dir /
            f"{img_path.stem}.txt"
        )
        if label_path.exists():
            with open(label_path) as f:
                for line in f.readlines():
                    cls_id, x, y, bw, bh = map(
                        float,
                        line.strip().split()
                    )
                    x1 = (x - bw/2) * w
                    y1 = (y - bh/2) * h
                    annotations.append({
                        "bbox": [
                            x1,
                            y1,
                            bw * w,
                            bh * h

```

```

        ],
        "category_id": int(cls_id),
        "area": bw*w*bh*h,
        "iscrowd": 0
    })
    target = {
        "image_id": idx,
        "annotations": annotations
    }
    encoding = self.processor(
        images=image,
        annotations=target,
        return_tensors="pt"
    )
    pixel_values = encoding["pixel_values"].squeeze(0)
    labels = encoding["labels"][0]
    return pixel_values, labels

def detr_collate_fn(batch):
    pixel_values = [item[0] for item in batch]
    encoding = processor.pad(
        pixel_values,
        return_tensors="pt"
    )
    labels = [item[1] for item in batch]
    return {
        "pixel_values": encoding["pixel_values"],
        "pixel_mask": encoding["pixel_mask"],
        "labels": labels
    }

detr_train_dataset = DETRDataset(
    DATASET_DIR / "train/images",
    DATASET_DIR / "train/labels",
    processor
)
detr_train_loader = DataLoader(
    detr_train_dataset,
    batch_size=4,
    shuffle=True,
    collate_fn=detr_collate_fn
)

A.7 Побудова та навчання Deformable DETR

from transformers import (
    AutoImageProcessor,
    DeformableDetrForObjectDetection
)
processor = AutoImageProcessor.from_pretrained(
    "SenseTime/deformable-detr"
)
detr_model = DeformableDetrForObjectDetection.from_pretrained(
    "SenseTime/deformable-detr",
    num_labels=NUM_CLASSES,
    ignore_mismatched_sizes=True,
    id2label=id2label,
    label2id=label2id
)
detr_model.to(DEVICE)
print("DETR loaded")

```

```

detr_model = DeformableDetrForObjectDetection.from_pretrained(
    "SenseTime/deformable-detr",
    num_labels=NUM_CLASSES,
    ignore_mismatched_sizes=True,
    id2label=id2label,
    label2id=label2id
)
detr_model.to(DEVICE)
print("DETR loaded")

DETR_EPOCHS = 15
optimizer_detr = torch.optim.AdamW(
    detr_model.parameters(),
    lr=2e-5,
    weight_decay=1e-4
)
loss_history = []
for epoch in range(DETR_EPOCHS):
    detr_model.train()
    epoch_loss = 0.0
    pbar = tqdm(
        detr_train_loader,
        desc=f"DETR Epoch {epoch+1}/{DETR_EPOCHS}"
    )
    for batch in pbar:
        pixel_values = batch["pixel_values"].to(DEVICE)
        pixel_mask = batch["pixel_mask"].to(DEVICE)
        labels = [
            {k: v.to(DEVICE) for k, v in t.items()}
            for t in batch["labels"]
        ]
        outputs = detr_model(
            pixel_values=pixel_values,
            pixel_mask=pixel_mask,
            labels=labels
        )
        loss = outputs.loss
        optimizer_detr.zero_grad()
        loss.backward()
        optimizer_detr.step()
        epoch_loss += loss.item()
        pbar.set_postfix(
            loss=f"{loss.item():.4f}"
        )
    avg_loss = epoch_loss / len(detr_train_loader)
    loss_history.append(avg_loss)
    # SAVE AFTER EVERY EPOCH
    torch.save(
        detr_model.state_dict(),
        MODELS_DIR / "best_detr.pth"
    )
    torch.save(
        {
            "epoch": epoch + 1,
            "model_state_dict": detr_model.state_dict(),
            "optimizer_state_dict": optimizer_detr.state_dict(),
            "loss_history": loss_history
        },
        MODELS_DIR / "detr_checkpoint.pth"
    )
    np.save(
        RESULTS_DIR / "detr_loss.npy",

```

```

        np.array(loss_history)
    )
    print(
        f"Epoch {epoch+1}/{DETR_EPOCHS} | "
        f"Average Loss: {avg_loss:.4f}"
    )
print("Deformable DETR training complete")

A.8 Оцінювання моделей та COCO-метрики
def build_coco_gt():
    images = []
    annotations = []
    ann_id = 0
    image_paths = sorted(
        list((DATASET_DIR / 'test/images').glob('*'))
    )
    for idx, img_path in enumerate(image_paths):
        image = cv2.imread(str(img_path))
        h, w = image.shape[:2]
        images.append({
            'id': idx,
            'file_name': img_path.name,
            'width': w,
            'height': h
        })
        label_path = DATASET_DIR / 'test/labels' / f'{img_path.stem}.txt'
        if label_path.exists():
            with open(label_path) as f:
                for line in f.readlines():
                    cls_id, x, y, bw, bh = map(float, line.strip().split())
                    bw_abs = bw * w
                    bh_abs = bh * h
                    x1 = (x - bw/2) * w
                    y1 = (y - bh/2) * h
                    annotations.append({
                        'id': ann_id,
                        'image_id': idx,
                        'category_id': int(cls_id) + 1,
                        'bbox': [x1, y1, bw_abs, bh_abs],
                        'area': bw_abs*bh_abs,
                        'iscrowd': 0
                    })
                    ann_id += 1

    categories = [
        {'id': i + 1, 'name': c}
        for i, c in enumerate(CLASSES)
    ]
    coco = {
        'images': images,
        'annotations': annotations,
        'categories': categories
    }
    with open(RESULTS_DIR / 'gt_coco.json', 'w') as f:
        json.dump(coco, f)

build_coco_gt()
def coco_eval(predictions, model_name):
    pred_path = RESULTS_DIR / f'{model_name}_preds.json'
    with open(pred_path, 'w') as f:
        json.dump(predictions, f)
    coco_gt = COCO(str(RESULTS_DIR / 'gt_coco.json'))
    coco_dt = coco_gt.loadRes(str(pred_path))
    evaluator = COCOeval(coco_gt, coco_dt, 'bbox')

```

```

evaluator.evaluate()
evaluator.accumulate()
evaluator.summarize()
return {
    'model': model_name,
    'mAP50_95': float(evaluator.stats[0]),
    'mAP50': float(evaluator.stats[1]),
    'recall': float(evaluator.stats[8])
}

comparison_df = pd.DataFrame([
    {
        "model": "YOLOv8",
        "mAP50": 0.8104,
        "mAP50_95": 0.5796,
        "recall": 0.6231
    },
    {
        "model": "Faster R-CNN",
        "mAP50": 0.8173,
        "mAP50_95": 0.5247,
        "recall": 0.5792
    },
    {
        "model": "Deformable DETR",
        "mAP50": 0.4893,
        "mAP50_95": 0.2589,
        "recall": 0.3624
    }
])
display(comparison_df)
comparison_df.to_csv(
    RESULTS_DIR / "comparison_metrics.csv",
    index=False
)

```

#### A.9 Вебзастосунок Gradio

```

CLASSES = [
    "gun",
    "knife",
    "pliers",
    "scissors",
    "wrench"
]

import gradio as gr
import pandas as pd
comparison_df = pd.DataFrame({
    "Model": [
        "YOLOv8",
        "Faster R-CNN",
        "Deformable DETR"
    ],
    "mAP50": [
        0.8104,
        0.8173,
        0.4893
    ],
    "mAP50-95": [
        0.5796,
        0.5247,
        0.2589
    ]
})

```

```

    ],
    "Recall": [
        0.6231,
        0.5792,
        0.3624
    ]
})
with gr.Blocks(
    theme=gr.themes.Glass(),
    title="Intelligent Baggage Inspection System"
) as demo:
    gr.Markdown("""
# Intelligent Baggage Inspection System
### Real-Time X-ray Baggage Screening Simulation
Detection of prohibited objects in X-ray baggage images using deep learning object
detection models.
**Dataset:** SIXRay
**Supported Objects:** Gun, Knife, Pliers, Scissors, Wrench
""")
    with gr.Tabs():
        # OBJECT DETECTION
        with gr.Tab("Object Detection"):
            with gr.Row():
                with gr.Column(scale=1):
                    input_images = gr.File(
                        file_count="multiple",
                        file_types=["image"],
                        label="Upload X-ray Images (up to 3)"
                    )
                    model_selector = gr.Radio(
                        [
                            "YOLOv8",
                            "Faster R-CNN",
                            "Deformable DETR"
                        ],
                        value="YOLOv8",
                        label="Detection Model"
                    )
                    detect_btn = gr.Button(
                        "Run Detection",
                        variant="primary"
                    )
                with gr.Column(scale=2):
                    output_gallery = gr.Gallery(
                        label="Detection Results",
                        columns=3,
                        height=450
                    )
            detected_table = gr.Dataframe(
                headers=[
                    "Object",
                    "Confidence"
                ],
                label="Detected Objects"
            )
            object_count = gr.Number(
                label="Detected Objects Count"
            )
            threat_status = gr.Markdown()
            detect_btn.click(
                fn=predict,

```

```

        inputs=[
            input_images,
            model_selector
        ],
        outputs=[
            output_gallery,
            detected_table,
            threat_status,
            object_count
        ]
    )
    # MODEL COMPARISON
    with gr.Tab("Model Comparison"):
        gr.Markdown("""
## Model Performance Evaluation
""")
        gr.Dataframe(
            value=comparison_df,
            interactive=False,
            label="Evaluation Metrics"
        )
        gr.Image(
            value=str(
                RESULTS_DIR /
                "map50_comparison.png"
            ),
            label="mAP50 Comparison"
        )
        gr.Image(
            value=str(
                RESULTS_DIR /
                "map50_95_comparison.png"
            ),
            label="mAP50-95 Comparison"
        )
        gr.Image(
            value=str(
                RESULTS_DIR /
                "recall_comparison.png"
            ),
            label="Recall Comparison"
        )
    # TRAINING ANALYSIS
    with gr.Tab("Training Analysis"):
        gr.Markdown("""
## Training Curves
""")
        gr.Image(
            value=str(
                RESULTS_DIR /
                "faster_rcnn_loss.png"
            ),
            label="Faster R-CNN Training Loss"
        )
        gr.Image(
            value=str(
                RESULTS_DIR /
                "detr_loss.png"
            ),
            label="Deformable DETR Training Loss"
        )

```

```

        gr.Image(
            value=str(
                RESULTS_DIR /
                "yolo_training/results.png"
            ),
            label="YOLOv8 Training Metrics"
        )
    # ABOUT
    with gr.Tab("About"):
        gr.Markdown("""
# Project Information
## Purpose
Development of an intelligent system for detecting prohibited objects in X-ray baggage
images.
## Dataset
SIXRay Dataset
## Detected Classes
- Gun
- Knife
- Pliers
- Scissors
- Wrench
## Evaluated Models
- YOLOv8
- Faster R-CNN
- Deformable DETR
## Experimental Results
| Model | mAP50 | mAP50-95 | Recall |
|-----|-----|-----|-----|
| YOLOv8 | 81.04% | 57.96% | 62.31% |
| Faster R-CNN | 81.73% | 52.47% | 57.92% |
| Deformable DETR | 48.93% | 25.89% | 36.24% |
## Bachelor Thesis Project
""")
demo.launch(
    share=True,
    debug=False
)

```