ПОЛТАВСЬКИЙ ДЕРЖАВНИЙ АГРАРНИЙ УНІВЕРСИТЕТ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ЕКОНОМІКИ, УПРАВЛІННЯ, ПРАВА ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КАФЕДРА ІНФОРМАЦІЙНИХ СИСТЕМ ТА ТЕХНОЛОГІЙ

Освітньо-професійна програма Інформаційні управляючі системи та технології
Спеціальність 126 Інформаційні системи та технології
Ступінь вищої освіти Магістр

ДОПУСКАЄТЬСЯ ДО ЗАХИСТУ
Завідувач кафедри
_____ Юрій УТКІН
«15» грудня 2022 року

КВАЛІФІКАЦІЙНА РОБОТА

на тему: **«Модель обробки даних отриманих з БПЛА для вирішення окремих завдань сучасного аграрного підприємства»**

виконав здобувач вищої освіти денної форми навчання
Кириченко Юрій Володимирович

Керівник кваліфікаційної роботи
доцент, к. т. н. Ігор СЛЮСАРЬ

Полтава – 2022 року

ЗМІСТ

ВСТУП 6
РОЗДІЛ 1. КОНЦЕПЦІЯ SMART FARM 9
1.1 Технологічний базис концепції Smart Farm 9
1.2 Аналіз напрямів застосування БПЛА в рамках реалізації концепції Smart Farm 14
1.3 Огляд основних послуг на базі дронів для агросектору 21
1.4 Актуальні окремі завдання сучасного аграрного підприємства .. 22
Висновки до розділу 1 24
РОЗДІЛ 2. МОДЕЛЬ ОБРОБКИ ДАНИХ З БПЛА ДЛЯ РЕАЛІЗАЦІЇ OBJECT DETECTION 25
2.1 Принципи роботи Object Detection 25
2.2 Архітектура R-CNN 29
2.3 Архітектура YOLO 32
2.4 Варіант використання YOLO для Object Detection пожеж 39
Висновки до розділу 2 49
РОЗДІЛ 3. РЕКОМЕНДАЦІЙ ЩОДО ВИКОРИСТАННЯ ЗАПРОПОНОВАНОЇ МОДЕЛІ ОБРОБКИ ДАНИХ, ЩО ОТРИМАНІ З БПЛА 51
3.1 Анотація зображень 51
3.2 Порівняльна оцінка точності нейронних мереж Object Detection 55
3.3 Техніко-економічне обґрунтування прийнятих рішень 59
Висновки до розділу 3 62
ВИСНОВКИ 63
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ 65
ДОДАТКИ 72

ВСТУП

Актуальність теми кваліфікаційної роботи підтверджується необхідністю реалізації концепція «Smart Farm» передбачає контроль широкого спектру сільськогосподарських параметрів на основі технологій IoT. При цьому, використання БПЛА в аграрному секторі у якості платформ для зондування та/або зв'язку з сенсорним сегментом IoT має великий потенціал для створення екосистем IoT. Однак, питання розробки рекомендацій щодо інтеграції БПЛА і екосистеми IoT на основі штучного інтелекту потребують додаткових досліджень. Все це свідчить про актуальність теми роботи.

Зв'язок роботи з науковими програмами, темами. Робота відповідає дослідженням в рамках науково-дослідної роботи «Управління стратегією інноваційного розвитку підприємств в контексті підвищення їх конкурентоспроможності на аграрному ринку, сталого розвитку та забезпечення продовольчої безпеки держави» (2021 р.), що фінансувалась господарськими договорами із замовниками, Концепції розвитку штучного інтелекту в Україні (розпорядження Кабінету Міністрів України № 1787-р від 29.12.2021), тематиці досліджень Навчально-дослідної лабораторії інтелектуальних систем, комп'ютерних мереж та інтернет речей Кафедри інформаційних систем та технологій Полтавського державного аграрного університету.

Метою кваліфікаційної роботи є підвищення ефективності аграрного підприємства за рахунок використання інтелектуальних функцій на основі нейронних мереж.

Завданнями кваліфікаційної роботи є:

- аналіз варіантів використання БПЛА при реалізації концепції «Smart Farm»;
- аналіз особливостей існуючих підходів щодо виконання процедури Object Detection;

– аналіз архітектури нейронних мереж типу Yolo;

– розроблення моделей нейронних мереж Object Detection;

– визначення точності запропонованих моделей нейронних мереж;

– формування рекомендацій щодо використання синтезованих архітектур нейронних мереж.

Об'єктом дослідження є процес детектування об'єктів за допомогою штучних нейронних мереж.

Предметом дослідження є точність нейронних мереж Object Detection.

Методами дослідження є: аналіз, синтез нейронних мереж Object Detection, вбудовані в фреймворк Keras математичні бібліотеки.

Інформаційна база кваліфікаційної роботи сформована з ресурсів, що містять інформацію про методи сегментації зображень, нейронні мережі та їх компоненти, що використовуються для виконання Object Detection, а також інструментарій для розробки та дослідження згорткових нейронних мереж.

Елементи наукової новизни роботи полягають в архітектур нейронних мереж для реалізації функцій Object Detection, порівняльній оцінці точності нейронних мереж на основі модифікованих архітектур Yolo.

Практична значущість роботи полягає у формуванні рекомендацій щодо використання запропонованих архітектур нейронних мереж в інтересах автоматизованого моніторингу пожежної безпеки посівів, які можуть бути використані для подальших досліджень за даною тематикою та при реалізації концепції Smart Farm.

Апробація результатів відбувалася в рамках IV Міжнародної науково-практичної конференції, присвяченої 50-й річниці кафедрі Інформаційних систем та технологій «Інтеграція інформаційних систем та інтелектуальних технологій в умовах трансформації інформаційного суспільства» (жовтень 2021 р., м. Полтава), Щорічної студентської наукової конференції Полтавського державного аграрного університету (листопад 2022 р., м. Полтава).

За результатами досліджень здійснено 3 публікації тез доповідей.

Структура кваліфікаційної роботи логічно пов'язана з завданнями досліджень і містить вступ, три розділи основної частини, висновки, список використаних джерел, додатки. Загальний обсяг пояснювальної записки кваліфікаційної роботи складає 72 сторінки формату А4. Вона містить 42 рисунки і 1 таблиця.

РОЗДІЛ 1
КОНЦЕПЦІЯ SMART FARM

1.1 Технологічний базис концепції Smart Farm

Постійно зростаюче населення вимагає збільшення виробництва продуктів харчування. Це спонукає до розвитку системи сільського господарства. Технічний прогрес у сільському господарстві збільшує потреби зростаючого населення. Протягом останніх кількох років обсяги інвестицій у сільське господарство збільшуються, щоб до 2050 р. забезпечити приріст посівів принаймні на 70 % [2], оскільки посівні площі будуть зменшуватися.

Практично з кожного аспекту сільського господарства можна отримати вигоду за допомогою технологічних досягнень. Розумне сільське господарство не тільки допомагає постачати продовольством зростаюче населення, але й сприяє розвитку останніх тенденцій, таких як органічне землеробство тощо [3].

Розумні ферми та точне землеробство передбачають інтеграцію передових технологій у існуючі методи ведення сільського господарства з метою підвищення ефективності виробництва та якості сільськогосподарської продукції. Як додаткову перевагу вони також покращують умови життя працівників, скорочуючи важку ручну працю та стомлюючі щоденні завдання.

Таким чином, Smart Farm – це концепція, що стосується управління сільгоспугіддями, з використанням сучасних інформаційних технологій для збільшення кількості та якості продукції, при оптимізації необхідної людської праці. Це означає управління фермами з використанням сучасних інформаційно-комунікаційних технологій для досягнення вищих результатів продуктивності при оптимізації необхідної робочої сили [4]. До таких технологій слід віднести:

– робототехніка – БПЛА, автоматизовані виробничі машини та ін.;

– датчики – для виміру води, температури, ґрунту, вологості, світла та ін.;

– аналіз даних – сучасний аналіз даних, алгоритми обробки зображень та ін.;

– програмне забезпечення: спеціальні програмні рішення, які націлені на конкретні типи ферм або використовують платформи, які можуть керувати різними сільськогосподарськими процесами та системами, не знаючи деталей системи.

– підключення: Wi-Fi, 5G, LTE, стільникові мережі, супутники, GPS та ін.

Пристрої IoT можуть контролювати поля, посіви та навколишнє середовище. Набір обчислювальних пристроїв IoT підключено через Інтернет, і всі цифрові та механічні компоненти мають унікальні ідентифікатори (UID), і вони можуть передавати дані через мережу без взаємодії «людина-комп'ютер» або «людина-людина» [5]. Датчики, зв'язок, обробка даних та інтерфейс користувача є основою IoT. Пристрої IoT сприймають навколишнє середовище за допомогою датчиків (наприклад, датчиків камери), надсилають дані безпосередньо на контролер або хмару, отримують відповіді від контролера хмарних серверів і виконують роботу відповідно.

Інноваційні, автономні «агроботи» та дрони корисні, але те, що справді зробить майбутню ферму «розумною», стане тим, що об'єднає всю цю технологію це IoT. IoT став чимось на зразок всеосяжного терміну для ідеї про те, щоб комп'ютери, машини, обладнання та пристрої всіх типів були з'єднані один з одним, обмінювалися даними таким чином, щоб вони могли працювати як так звану розумну структуру. Вже зараз спостерігаємо використання технологій IoT у багатьох сферах, таких як пристрої для розумного дому та цифрові помічники, розумні житлові комплекси та офіси, розумні готелі та підприємства. Smart Farm будуть мати вбудовані датчики на кожному етапі процесу та на кожному обладнанні. Датчики, встановлені на полях, будуть збирати дані про рівень освітленості, стан ґрунту, іригацію, якість повітря та

погоду. Ці дані будуть повернуті власнику або безпосередньо «агроботам» у полі. Команди роботів перетинатимуть поля, реагуючи на потреби сільськогосподарських культур, і виконуватимуть функції прополювання, поливу, обрізки або збору врожаю, керуючись власним збором з датчиків.

Основна концепція впровадження автономної робототехніки на Smart Farm має на меті зниження залежності від ручної праці за одночасного підвищення ефективності, продуктивності та якості продукції.

Дуже важливим чинником для всіх «агроботов» є наявність надійної магістралі датчиків IoT, вбудованих в інфраструктуру господарства. Ключ до

Smart Farm залежить від здатності всіх машин і датчиків взаємодіяти один з

одним і з фермером, навіть якщо вони працюють автономно.

БПЛА також є пристроями Інтернету речей, оскільки вони оснащені камерами, GPS тощо та контролюються користувачем або віддаленим сервером за допомогою підключення до Інтернету. БПЛА або дрони – це одноразові або відновлювані літаючі об'єкти, які можуть літати самостійно без пілота за допомогою дистанційного керування на землі або за програмами попереднього прибуття. БПЛА оснащені різними датчиками, які можуть виконувати різні завдання, які роблять БПЛА інтелектуальними, наприклад камери RGB, LIDAR, RADAR, інфрачервоні камери тощо [6]. БПЛА класифікуються за формою, вагою, радіусом дії, швидкістю та іншими факторами. Наприклад, БПЛА класифікуються та підсумовуються за формою їхніх крил:

– БПЛА з нерухомим крилом підходять для великих полів; на них менше впливає вітер, тривалі перельоти, велика висота; їм потрібен вертикальний вогонь.

– Гвинтокрилі БПЛА підходять для невеликих поверхонь для зльоту та посадки, вимагаючи постійного вогню та вогню під різними кутами.

БПЛА можна використовувати для виконання небезпечних і ризикованих місій для людини. Спочатку винайдені для військових застосувань, сьогодні БПЛА успішно використовуються в кількох цивільних цілях, таких як сільське господарство, поліція, спостереження, рекреаційні цілі тощо [7]. Безпілотники використовуються для аерофотозйомки, моніторингу переміщень цивільного населення під час громадських зібрань, моніторингу зібрань і оголошення про правила карантину в районах, уражених COVID-19, щоб протистояти пандемії COVID-19 [8, 9].

З розвитком технологій БПЛА тепер БПЛА можуть робити те, що раніше вимагало гелікоптерів. Переваги UVA в розумному сільському господарстві головним чином забезпечують мобільність у сільському господарстві та можуть працювати за змінних погодних умов. Використання цих БПЛА для сканування сільськогосподарського поля з вищесказаного пропонує багато можливостей. Отримані різні типи зображень дозволяють Розумним фермам збирати докладніші дані, ніж будь-коли раніше. Можливість регулярно виконувати всі ці польові дослідження покращує планування схем посіву насіння, іригації та картографування розташування як у 2D, так і в 3D. Маючи всі ці дані, власники можуть оптимізувати кожен аспект управління земельними та сільськогосподарськими культурами на своїй Smart фермі.

Такий підхід виводить на новий якісний рівень моніторинг та аналіз у реальному часі. Існують переваги використання невеликого парку безпілотників замість групи робітників, які проводять годинник на ногах або в транспортному засобі, що подорожує полем, щоб візуально перевірити умови посіву. Саме тут важлива підключена Smart Farm, оскільки всі ці дані потрібні. Власники можуть переглядати інформацію та здійснювати особисті поїздки в поля лише тоді, коли є конкретна проблема, яка потребує їхньої уваги, замість витрачати час на турботу про здорові рослини. Безпілотники можна використовувати не тільки як камери, вони знаходять застосування у посадці та обприскуванні.

Однак досліджувати отримані з БПЛА зображення людині непросто. Глибоке навчання вступає в дію в цих областях. Згорткові нейронні мережі (CNN), ймовірно, є найбільш широко використовуваною архітектурою глибокого навчання [10]. Нещодавнє зростання глибокого навчання пояснюється величезною популярністю та ефективністю ConvNets. Недавній інтерес до CNN почався з AlexNet у 2012 р., і відтоді він експоненціально зріс [11]. За допомогою ConvNets можна легко здійснити виявлення об'єктів [12] і сегментацію зображення [13].

На даний момент глибоке навчання мало впроваджено в галузі сільського господарства, але все швидко змінюється [14]. Як відомо, глибоке навчання – це підмножина алгоритмів машинного навчання, наприклад моделі навчання. Ці моделі можуть навчитися виконувати аналіз шаблонів або класифікацію безпосередньо з зображень, текстів, звуків або відео. Ці моделі іноді досягають людського рівня точності. Для навчання цих моделей використовується великий набір позначених даних; також архітектури нейронних мереж можуть мати багато рівнів [15].

Глибоке навчання є основною технологією, яка лежить в основі успіху безпілотних автомобілів як БПЛА. Він використовується в пристроях для керування мовленням, таких як гучномовці (Alexa, Google Home), планшети, телефони, телевізори та ін. Глибоке навчання досягає результатів, які раніше були неможливими, і воно має багато можливостей. Деякими прикладами є віртуальні помічники, перекладачі, бачення для безпілотних вантажівок, дронів, автономних автомобілів, розпізнавання обличчя та ін. [19, 20].

Таким чином, поєднання штучного інтелекту на базі глибокого навчання та БПЛА може значно підвищити ефективність сільського господарства [14]. Надалі доцільно розглянути завдання, що можна вирішувати за допомогою БПЛА.

1.2 Аналіз напрямів застосування БПЛА в рамках реалізації концепції Smart Farm

Згідно [16], найбільшого поширення у світі набули такі варіанти використання БПЛА.

1. Виявлення та класифікація рослин. На великих сільськогосподарських полях людям дуже важко виявити такі рослини, як трава та бур'яни. Якщо справжні посіви та бур'яни не будуть виявлені та класифіковані, це може вплинути на прибуток, а також на навколишнє середовище через надмірне використання добрив, пестицидів та інсектицидів. Багато рослин є сировиною для медичної промисловості, і ідентифікація цих рослин є виснажливою роботою для людини. Багато рідкісних рослин потребують збереження та ідентифікації. Тобто можна використовувати БПЛА для збору інформації, а використання машинного навчання повинно бути дуже простим, економічним і менш трудомістким. Пояснення класифікації рослин за допомогою БПЛА наведено на рис. 1.1. БПЛА роблять знімки, потім ці зображення проходять через різні алгоритми класифікації рослин, потім рослини класифікуються відповідно [17].

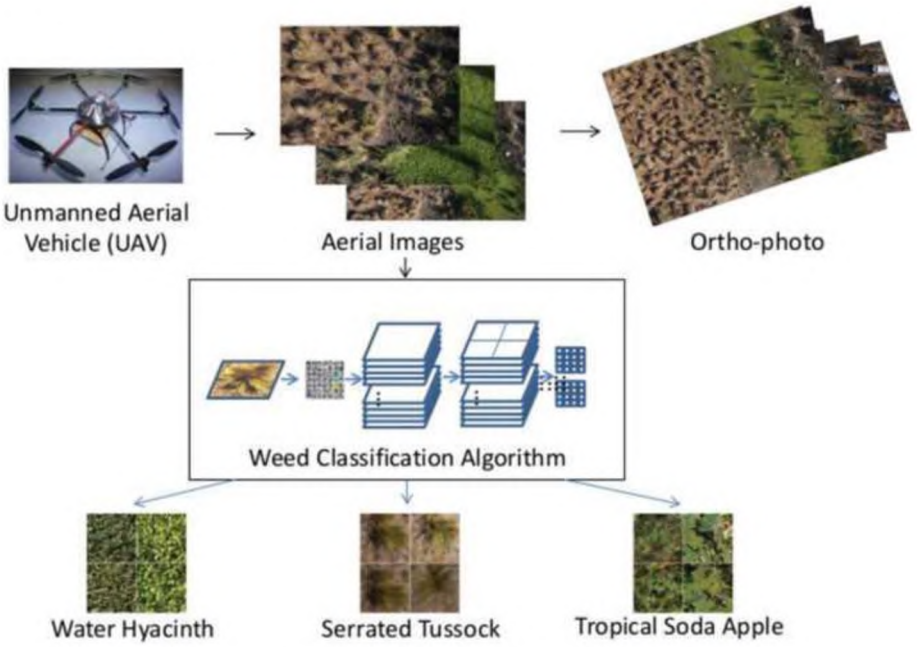

Рисунок 1.1 – Виявлення рослин за допомогою зображень БПЛА

В [18] розглянуто, як зображення БПЛА використовуються для вивчення рослинності для ідентифікації трави на луках. В основі лежить платформа з подвійною камерою на основі БПЛА [19]. Використовуючи класифікацію *k*-середніх, проводиться оцінка кількість квіток ріпаку. В [17] запропонований алгоритм на основі методів навчання для ідентифікації бур'янів на полях за допомогою зображень БПЛА. Автори стверджують, що досягли 90% точності. В [20] наведена інтегрована конвеєрна методологія з використанням дерев рішень із посиленням градієнта для картографування трав і рослинності в посушливих землях із точністю близько 96 %. В [21] описана вдосконалена процедура класифікації ярих культур за допомогою методів машинного навчання на основі ієрархічних моделей, що дає більшу продуктивність. У [22] запропонована архітектура CNN для класифікації рослин із послідовностей зображень з точністю 97,47 % для класифікації 16 видів рослин. В [23] наведено алгоритм для реорганізації листка рослини за допомогою машини опорних векторів, який може класифікувати 32 види рослин. У [24] CNN продемонструвала середнью точність 86,2 % при класифікації 22 види рослин, включаючи бур'яни та зернові культури. В [25] наведено результати експериментів з чотирма різними моделями глибокого навчання з чотирма відкритими наборами даних: Flavia, Swedish Leaf, UCI Leaf і PlantVillage. У [26] продемонстровано приклад моделі глибокого навчання CNN для ідентифікації рослини за морфологічними моделями їхніх листкових жилок. Для вирішення вказаного завдання в [27] використані гібридні моделі, що забезпечують кращу продуктивність, ніж чисті CNN, автокодер (AE) або машини опорних векторів (SVM) для листової класифікації. Досить цікавою є система [28], яка може розрізняти бур'яни та цукрові буряки за допомогою CNN. В [29] автори запропонували набір даних BJFU100 і модель глибокого навчання ResNet26, що досягла 91,78 % точності та опублікували свій набір даних для академічного використання. У [30] наведені результати експериментів з R-CNN, YOLOv3 і Retina Net, які показали, що RetinaNet досягла найвищої точності з середнім значенням 92,64 %. Існує і багато інших

робіт за даним напрямом використання БПЛА.

2. Оцінка здоров'я рослин. Оскільки технологія продовжує вдосконалюватися, вона також потребує впровадження систем автоматизації на сільськогосподарських полях. Щоб розвивати високотехнологічні сільськогосподарські системи, потрібно зосередитися на здоров'ї рослин, де можна виділити кілька напрямів:

– моніторинг здоров'я посівів;

– виявлення хвороб і пошкоджень культур;

– сортування та аналіз форми плодів сільськогосподарської продукції.

Моніторинг стану рослин. Робота [31] представляє використання машинного навчання для розробки методології, яка аналізує цифрові зображення цитрусових рослин, зібраних з БПЛА. Наземні датчики, включаючи вимірювач водного потенціалу, хлорофілометр і спектрорадіометр, використовуються для моніторингу стану ферми. Тут за допомогою цього методу застосували техніку точного землеробства. Зібрані зображення та основні правдиві дані потім використовуються як навчальні дані для моделей машинного навчання. У 2015 р. опублікована реалізація автоматизованого рішення для еволюції здоров'я рослин на кукурудзяних полях [32], яка використовує невеликі БПЛА та алгоритми CV, які працюють з інформацією у візуальному спектрі RGB. Це дає численні результати щодо використання комерційних датчиків RGB для надання фермерам важливої інформації про стан їхніх полів. Головним мотивом їх досліджень є зменшення внесення добрив і підвищення врожайності. Автори досягли 82 % продуктивності моделі.

3. Виявлення хвороб і пошкоджень культур. В якості прикладу слід розглянути роботу [33], в якій описано експеримент з посівами пшениці на двох полях одне було заражене гельмінтоспоріозом (HLB), а інший – нормальним. ГЛБ-інфекція призвела до жовтизни листя на початковій стадії. БПЛА (Phantom 4) з бортовою камерою RGB використовувався для зйомки зображень 4000x3000 пікселів з висоти 80 м. нейронна мережа CNN досягла

точності 91,43 %, а похибка становила 0,83 %, перевершуючи інші за рівнем точності та стабільності. В [34] висвячені результати експериментів з октокоптером DJI S1000 і 5-діапазонною мультиспектральною камерою. Систему випробували на озимій пшениці, зараженій інокулятом жовтої іржі. БПЛА робив знімки на відстані від 16 до 24 метрів із роздільною здатністю від 1 до 1,5 см/піксель. Розроблена система досягла середньої точності 89,2 %, запам'ятовування 89,4 % і точності 89,3 %. В [35] збирали гіперспектральні дані за допомогою DJI Matrice 600Pro Hexacopter і гіперспектральної камери. Вони використовували ARI і TCARI 1, які точно виявляли рослини з раковою інфекцією. В [36] описаний експеримент з двома БПЛА, оснащеними датчиками RGB (12 МП). Вони зробили 40 зображень розміром 4000 x 3000 пікселів. Використовуючи GoogLeNet, вони виявили фузаріоз з точністю понад 90 %. Згідно [37], запроваджена система виявлення хвороб рослин за допомогою системи обробки зображень використовує алгоритм RELIEF-F для виявлення різних захворювань.

На рис. 1.2 показано кроки для виявлення хвороб рослин і класифікації хвороб.

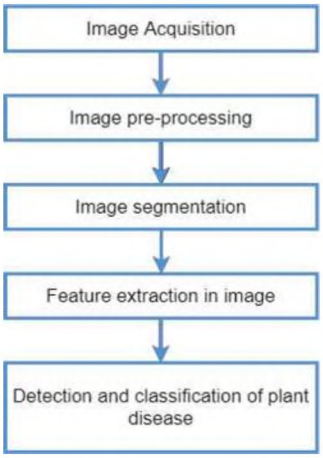

Рисунок – 1.2 Основні етапи виявлення та виявлення хвороб рослин [37]

У [38] наведена модель CNN для виявлення та діагностики захворювань рослин, використовуючи зображення листя здорових і хворих рослин, використовуючи методології глибокого навчання. На основі відкритої бази даних, яка містить 87848 зображень 25 різних рослин, наборів із 58 різних

класів комбінацій (рослин, хвороб). При цьому нейронна мережа досягла 99,53 % успіху. У [39] наведена система для виявлення різних хвороб листя рослин за допомогою CV та нечіткої логіки. В [40] досліджена бактеріальну хворобу Talya (вражає фрукти), а також запропонована вебсистема обробки зображень, яка швидко змінює розмір зображення. Потім ознаки виділяються на основі кольору, морфології та алгоритмів кластеризації (CCV) з *k*-середніми. Загалом, досягається точність 82 %. В [41] описана систему обробки зображень і машинного навчання (загальної точності 95 %) для діагностики хвороб рослин картоплі на основі набору даних PlantVillage і досягають. В [42] наведена техніка для ідентифікації та класифікації грибкових захворювань. Вивчались чотири різні сільськогосподарські/садівничі культури: овочеві культури, фруктові культури, зернові культури та комерційні культури. Пропонована робота спрямована на розробку методології класифікації та ідентифікації симптомів грибкових захворювань садових і сільськогосподарських культур.

4. Розумна боротьба зі шкідниками та травами. Шкідники – головна проблема сільському господарстві. Ефективне використання пестицидів необхідне як довкілля, так сільськогосподарських культур. Надмірне використання пестицидів також є неекономічним. Перевага розпилення пестицидів за допомогою БПЛА полягає у прискоренні процесу та запобіганні надмірному поглинанню пестицидів ґрунтом. Однак через швидкість і напрям вітру велика кількість пестицидів може переноситися на прилеглі поля. З підвищеною точністю можна обмежити втрати пестицидів під час розпилення зверху. Наприклад, в [43] запропонована архітектура, яка використовує БПЛА з системою поєднаних розпилювачів, які можуть спілкуватися з бездротовою мережею датчиків. Ця система має на меті надсилати відгуки про погодні умови та рівень обприскування на полі цільової культури. На основі інформації БПЛА відповідним чином застосовує оптимізований маршрут. На рис. 1.3 зображено БПЛА з розпилювачем [44]. Розпилювачі БПЛА працюють краще в порівнянні; вони є більш економічно ефективними та безвідходними. Вони досягають оптимальної ефективності контролю за допомогою БПЛА при

16,8 л/га (з насадками LU120-02, -03) із системним інсектицидом і при 28,1 л/га з контактним інсектицидом і фунгіцидом.

Рисунок 1.3 – БПЛА від CNR для лікування процесійної молі

В [45] автори провели порівняльне дослідження серед архітектур глибокого навчання VGG-19, ResNet-50, Inception-v3, VGG-16 і Xception із 5000 зображеннями шкідників сої та виявили, що архітектура ResNet-50 досягла точності до 93,82 %. В свою чергу, в [46] наведена систем для розпізнавання шкідників на основі аналізу помітності зображення та моделі глибокого навчання для класифікації видів комах на китайських чайних полях. У [47] запропонований мобільний додаток під назвою BioLeaf, який може вимірювати пошкодження листя сої. Додаток було протестовано на реальних зображеннях, зібраних із соєвих полів, і інструмент можна використовувати для культур різної ширини та вузького листя. В [48] проведено дослідження раннього моніторингу злакових бур'янів на полях кукурудзи з використанням зображень БПЛА та алгоритмів OBIA (рис. 1.4).

5. Аналіз поля та оцінка врожайності. Оцінка врожайності сільськогосподарських культур є важливим завданням в управлінні та маркетингу товарів. Точне прогнозування врожайності допомагає фермерам покращити якість їхнього врожаю. Це також допомагає зменшити експлуатаційні витрати, приймаючи кращі рішення щодо інтенсивності збору врожаю та необхідної праці. Загалом оцінки врожайності здійснюються з використанням попередніх даних, а працівники вручну підраховують плоди у

вибраних місцях поля. Існує два типи комп'ютерних методів оцінки продуктивності: регіональні методи та методи підрахунку.

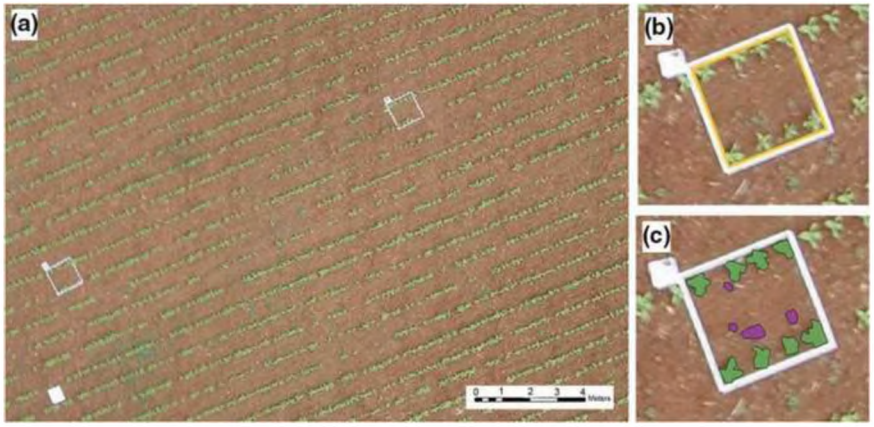

Рисунок – 1.4 Приклад отриманого БПЛА зображення поля соняшника з аналізом зображення, яке відокремлює бур'яни від соняшників

В [49] розглянута автоматична система для оцінки врожайності яблук, у якій зображення яблуневих садів фіксуються вночі для зменшення непередбачуваного природного освітлення вдень. В [50] використано кольорові зображення, щоб підрахувати кількість яблук у садах при природному освітленні, модель може виявити яблука з точністю понад 85 %. В [51] наведена методика оцінки врожайності насіння ріпаку за допомогою даних дистанційного зондування методом спектрального аналізу суміші. Оцінка врожайності є важливою задачею в сільському господарстві; точніша оцінка приносить більше прибутку. У майбутньому БПЛА повинні автоматично й точніше оцінювати ресурси та врожай у спосіб, який буде менш трудомістким і більш прибутковим.

Таким чином, можна виділити такі основні домени: виявлення та класифікація рослин, оцінка здоров'я рослин, інтелектуальна боротьба зі шкідниками та травами, а також аналіз поля та оцінка врожайності.

1.3 Огляд основних послуг на базі дронів для агросектору

Надалі доцільно розглянути типовий набір послуг для агропідприємств, що можна зустріти на сучасному ринку [52].

1. Моніторинг дронами. Великим та середнім підприємствам важко керувати всіма роботами на своїх землях в десятки тисяч гектарів. Контролювати всі технологічні операції та процеси та оперативно реагувати на визначені проблемні ситуації іноді просто неможливо. Проблема контролю та потреби постійної присутності в полі – саме це вирішується за допомогою комплексних послуг моніторингу дронами. Впродовж усього сезону пропонується оперативний інструмент моніторингу для візуальної оцінки стану посівів.

2. Карти внесення азоту (гербіцидів, СЗР). Внесення засобів захисту дронами дає можливість точково обробляти необхідні дільниці.

Карти внесення гербіцидів формуються на основі інформації про посів та карти розвитку бур'янів, отриманої за даними дистанційного сканування. Вивчаючи ці вступні, фахівці формують карти необхідних доз гербіциду у різних частинах поля. Створення карт розвитку бур'янів дозволяє вчасно визначити необхідність використання гербіцидів та оптимально розподілити їх по полю. Карти диференційованого внесення гербіцидів можуть застосовуватися як для ручного, так і для автоматичного регулювання доз внесення.

Таким чином, пропонується мати карти-рекомендації безпосередньо перед запланованим внесенням добрив.

3. Візуальні і NDVI карти полів. Регулярний моніторинг тисяч гектар для власників сільськогосподарських підприємств скрутний, і керівництву необхідний зручний формат відображення даних зі статистичною основою, що відображає проблемні ділянки, що потребують термінової уваги. Візуальні карти полів є найпростішим та зрозумілішим застосуванням БПЛА

у сільському господарстві. Це зображення поля у звичайних кольорах, що дозволяє розглянути стан посіву, оцінити неоднорідність поля та виявити характерні проблемні зони. Карти NDVI дозволяють виявити відмінності у стані рослин, які візуально не видно.

4. Обмір полів. Дистанційне сканування полів у поєднанні з наземними контрольними вимірами дозволяє проводити точний обмір розмірів, площ і висот полів з урахуванням спотворюючих факторів, таких як рельєф, опори ЛЕП, дороги та ін.

5. Трихограма. Розселення трихограми на порядок дешевше за хімічний захист рослин, а існуючі технології дозволяють мінімізувати витрати. Фахівці можуть виконувати розселення трихограми з дрону, що дозволяє боротися більш ніж з 70 видами шкідників.

1.4 Актуальні окремі завдання сучасного аграрного підприємства

Доступність аеровізуальних даних та останні досягнення в алгоритмах виявлення об'єктів (це завдання має назву – функції Object Detection) дозволяють застосовувати такі технології в інтересах агросектору.

В свою чергу, згідно [53], станом на початок серпня 2022 р. на території України було ліквідовано 280 пожеж на сільгоспугіддях (рис. 1.5). Їхня загальна площа становила понад 5000 гектарів. Працівниками Державної служби надзвичайних ситуацій було врятовано понад 1500 гектарів зернових від пожеж, що спалахують від обстрілу.

Це свідчить про необхідність вирішення завдання своєчасного виявлення джерела займання, що є важливим питанням захисту людей, тварин і територій від пожеж і вчасно вжити необхідних заходів безпеки.

Таким чином, в інтересах автоматизованого моніторингу пожежної безпеки посівів доцільно використовувати БПЛА.

Рисунок 1.5 – Пожежа на сільгоспугіддях

При здійсненні детектування пожежі за допомогою застосування алгоритмів CV необхідно мати знання про особливості диму та вогню, а також розуміти, які ознаки можна витягти з послідовності аерофотознімків, що містять пожежу або дим. Просторовими ознаками диму (потрібний один кадр відеопотоку для отримання значення ознаки) є колірна маску диму, прозорість та варіативність форми та розміру (фрактальність, турбулентність, нерівність меж, хаотичність). Часовими ознаками диму (необхідно два кадри і більше) є наявність руху та його аналіз, часові зміни кордонів, до яких відносяться мерехтіння та рух кордонів області задимлення. Основним недоліком робіт з детектування диму можна вважати відсутність достатніх досліджень впливу погодних умов на якість детектування диму за відеопотоком та вузькоспрямованість методів та алгоритмів, які в основному спрямовані на виділення лише ближнього диму або визначення лише далекого диму. Як наслідок, здійснювати інтелектуальну функцію моніторингу пожежної безпеки посівів за допомогою БПЛА необхідно реалізовувати на моделях глибокого навчання нейронних мереж Object Detection. В даному контексті, в якості

основних об’єктів розглядаються вогонь та дим. Враховуючи складність детектування диму можливо передбачити рішення, що містить одну нейронну мережу для виявлення диму, а іншу – для виявлення вогню.

Висновки до розділу 1

Розумне сільське господарство та точне землеробство передбачають інтеграцію передових технологій в існуючі методи ведення сільського господарства для ефективності виробництва та якості сільськогосподарської продукції. Розумні ферми передбачають використання автономних роботів, дронів або БПЛА, датчиків та платформ IoT.

БПЛА в рамках реалізації концепції Smart Farm може використовуватись за кількома напрямами, серед яких слід виділити: виявлення та класифікація рослин; оцінка здоров'я рослин (моніторинг здоров'я посівів; виявлення хвороб і пошкоджень культур; сортування та аналіз форми плодів сільськогосподарської продукції); виявлення хвороб і пошкоджень культур; розумна боротьба зі шкідниками та травами; аналіз поля та оцінка врожайності.

Доступність аеровізуальних даних та останні досягнення в алгоритмах Object Detection дозволяють застосовувати такі технології в інтересах агросектору. Однак досліджувати отримані з БПЛА зображення людині непросто. Глибоке навчання вступає в дію в цих областях. Це свідчить про необхідність вирішення завдання своєчасного виявлення джерела займання, що є важливим питанням захисту людей, тварин і територій від пожеж і вчасно вжити необхідних заходів безпеки. Таким чином, в інтересах автоматизованого моніторингу пожежної безпеки посівів доцільно використовувати БПЛА.

РОЗДІЛ 2
МОДЕЛЬ ОБРОБКИ ДАНИХ З БПЛА ДЛЯ РЕАЛІЗАЦІЇ OBJECT DETECTION

2.1 Принципи роботи Object Detection

Згідно [54], пожежа – одне з лих, що завдає найбільшої шкоди ресурсам. Дим, зазвичай, утворюється перед полум'ям, і через значне розширення диму його легше зафіксувати камерами. Тому виявлення диму є життєво важливим для запобігання пожежам [55].

Виявлення пожеж можна розділити на два основні типи: традиційний метод виявлення та метод виявлення на основі глибокого навчання.

При цьому, беручи до уваги час, витрачений на навчання традиційного алгоритму згорткової нейронної мережі, в [56] розширено набір даних шляхом синтезу диму від пожежі та реалізоване виявлення його за алгоритму Faster R-CNN, усунувши складний процес ручного вилучення ознак у традиційних методах відеодетектування. В [57], щоб підвищити надійність CNN і уникнути перенавчання, застосовано стратегію оптимізації для покращення функції втрат у множинному згортковому ядрі та пакетної нормалізації. В [58] зазначено, що для однієї моделі нейронної мережі важко досягти вилучення функцій у багатоскладних сценаріях, і кожна окрема нейронна мережа може виконувати різні функції. Це реалізовано шляхом використання трьох глибоких нейронних мереж. Таким чином, ці результати підтверджують можливість використання Object Detection для виявлення диму та вогню посівів під час моніторингу земель за допомогою БПЛА.

Object Detection – це визначення об'єктів на картинці / кадрі, тобто алгоритм або нейронна мережа визначають об'єкт і записують його позицію та bounding boxes (параметри прямокутників навколо об'єктів) – рис. 2.1. Для опису розташування об'єктів зазвичай використовується обмежувальна рамка (bounding box). Обмежуюча рамка є прямокутною рамкою, яка визначається за

(x, y) координатами лівого верхнього кута і (x, y) координатами правого нижнього кута прямокутника (рис. 2.2).

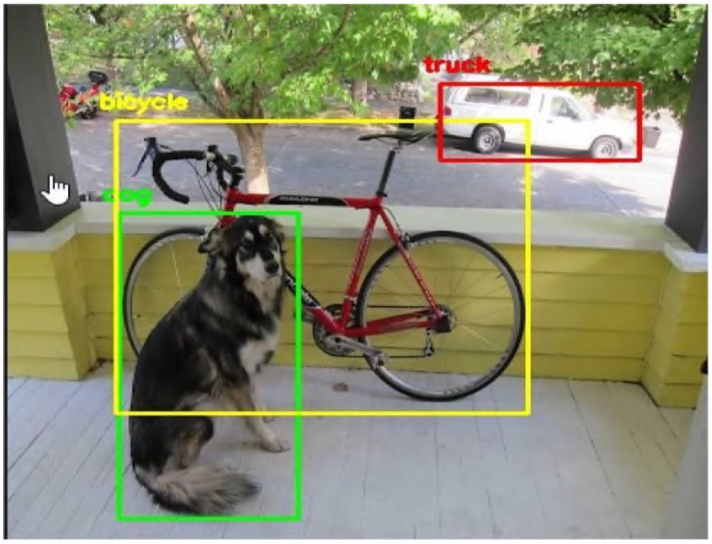

Рисунок 2.1 – Object Detection

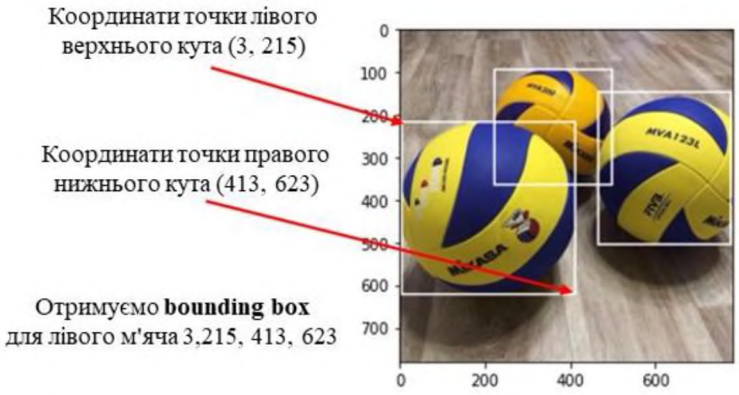

Рисунок 2.2 – Bounding box в завданні Object Detection

Bounding box може бути представлена 4 координатами у двох форматах:

– центрований (cx, cy, w, h);

– простий ($x_{\min}$, $y_{\min}$, $x_{\max}$, $y_{\max}$).

Щоб виділити об'єкт на зображенні (щодо інших об'єктів), потрібно ввести якісь критерії, наприклад колір, текстура або розмір. У реальних завданнях розподіл на такі найпростіші критерії дуже погано працює. Тому виникла потреба в алгоритмі, який би добре розділяв об'єкти на картинці один від одного з урахуванням усіх складнощів.

Алгоритм селективного пошуку (Selective Search). В основі алгоритму Selective search лежить використання методу ієрархічного угруповання схожих

регіонів на основі відповідності кольору, текстури, розміру або форми та графів. Для цих графів вершиною є інтенсивність поточного пікселя, а ребра поєднують пару сусідніх пікселів. Абсолютна різниця інтенсивностей пікселів вершин використовується як вага ребра. За допомогою графа виділяються фрагменти, які потім групуються згідно з наступним принципом: ребра між двома вершинами в одній групі повинні мати менші ваги, а ребра між двома вершинами в різних групах повинні мати більшу вагу. Після того як два найбільш схожі регіони згруповані, між отриманим регіоном та його сусідами обчислюється нова схожість. Процес угруповання найбільш схожих областей повторюється до того часу, поки все зображення стане однією областю. Таким чином, алгоритм Selective Search реалізується за допомогою детальної сегментації зображення в залежності від інтенсивності пікселів з використанням методу сегментації на основі графів і вибіркового пошуку (рис. 2.3).

На даний час є кілька поширених архітектур нейронних мереж, що створені для визначення об'єктів. Вони в основному поділяються на «дворівневі», такі як R-CNN, Fast R-CNN та Faster R-CNN, та «однорівневі», такі як YOLO.

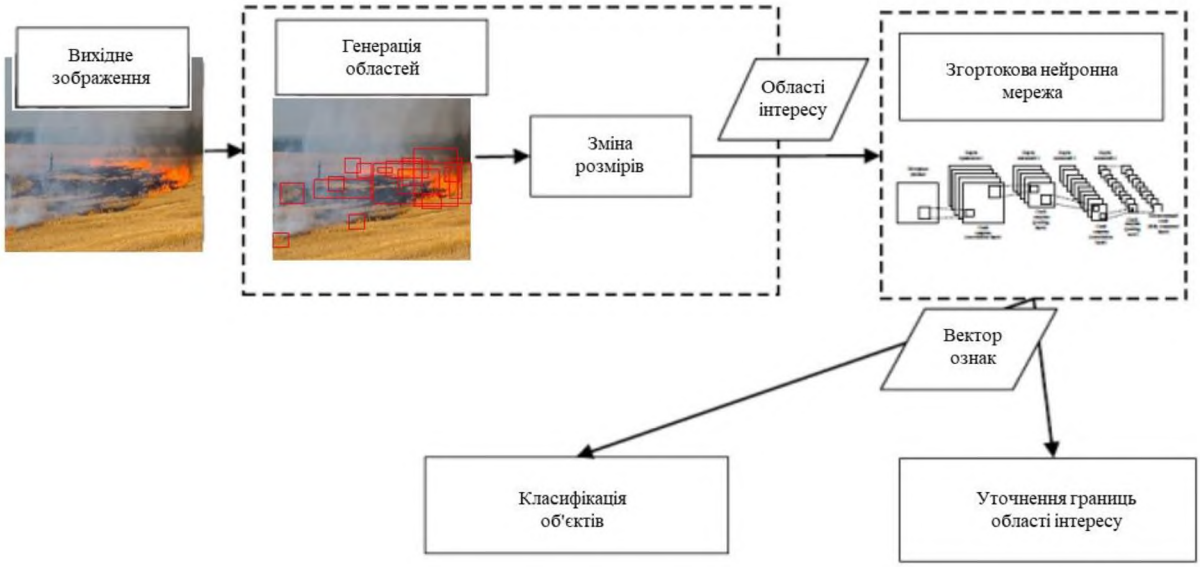

Рисунок 2.3 – Сутність Selective Search

Дворівневі нейронні мережі, перелічені вище, використовують звані регіони на зображенні, щоб визначити, чи у цьому регіоні певний об'єкт (рис.2.4).

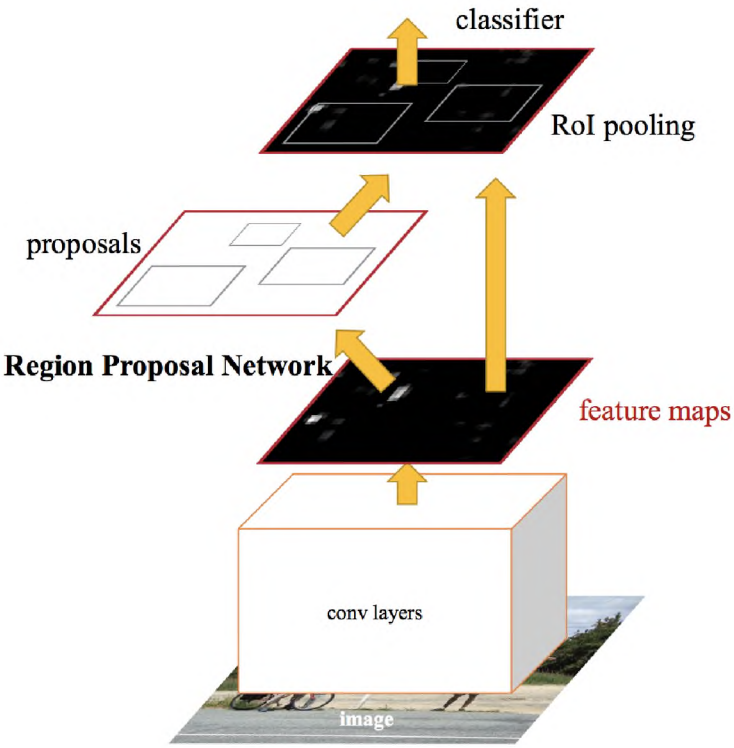

Рисунок 2.4 – Сутність роботи типової 2-рівневої нейронні мережі Object Detection

Зазвичай, це виглядає так (для Faster R-CNN, яка є найшвидшою із перерахованих дворівневих систем)

1. Подається картинка/кадр на вхід.

2. Кадр проходить через CNN для формування карти ознак.

3. Окремою нейронною мережею визначаються регіони з високою ймовірністю знаходження в них об'єктів.

4. Далі ці регіони за допомогою RoI Pooling стискаються та подаються до нейронної мережі, що визначає клас об'єкта в регіонах.

R-CNN – перша модель для вирішення цього завдання. Працює як стандартний класифікатор зображень. На вхід мережі подаються різні

регіони зображення та робиться передбачення. Дуже повільна, тому що проганяє одне зображення кілька тисяч разів.

Fast R-CNN. Покращена і швидша версія R-CNN працює за схожим принципом, але спочатку все зображення подається на вхід CNN, потім з отриманого внутрішнього уявлення генеруються регіони. Але, як і раніше, досить повільна для завдань реального часу.

Faster R-CNN. Головна відмінність від попередніх у тому, що замість алгоритму Selective Search для вибору регіонів використовує нейронну мережу для їхнього «заучування».

YOLO. Зовсім інший принцип роботи, порівняно з попередніми, не використовує регіони взагалі.

SSD (Single Shot Detector). За принципом схожа на YOLO, але як мережу для отримання ознак використовує VGG16. Теж задовільна швидкість і придатна для роботи в реальному часі.

Feature Pyramid Networks (FPN). Ще один різновид SSD, через особливості вилучення ознак краще, ніж SSD, розпізнає дрібні об'єкти.

RetinaNet. Використовує комбінацію FPN+ResNet і завдяки спеціальній функції помилки (Focal Loss) дає більш високу точність (Accuracy).

2.2 Архітектура R-CNN

Модель R-CNN (Regional Convolutional Neural Networks) орієнтована вирішення завдання пошуку об'єктів з одночасною їх класифікацією. При пошуку об'єктів за допомогою моделі R-CNN виконується наступна послідовність кроків:

– генерація областей інтересу для вихідного зображення;

– формування картки ознак;

– класифікація об'єктів.

Генерація областей інтересу. Виконується генерація областей інтересу (region proposals), що ймовірно містять у собі об'єкти, що шукаються з використанням різних алгоритмів (наприклад, Selective search або Regional-proposalfunction).

Формування картки ознак. На цьому кроці виконується формування карти ознак для вихідного зображення. Проводиться масштабування сформованих областей інтересу розмір, який можна порівняти з архітектурою нейронної мережі CNN. Сформовані дані надходять на вхід згорткової нейронної мережі (CNN). На виході нейронної мережі CNN формується 4096-мірний вектор ознак кожної області інтересу.

Класифікація об'єктів. На цьому кроці виконується класифікація об'єктів для кожної області інтересу з використанням сформованого вектору ознак на основі методу опорних векторів (SVM) – рис. 2.5.

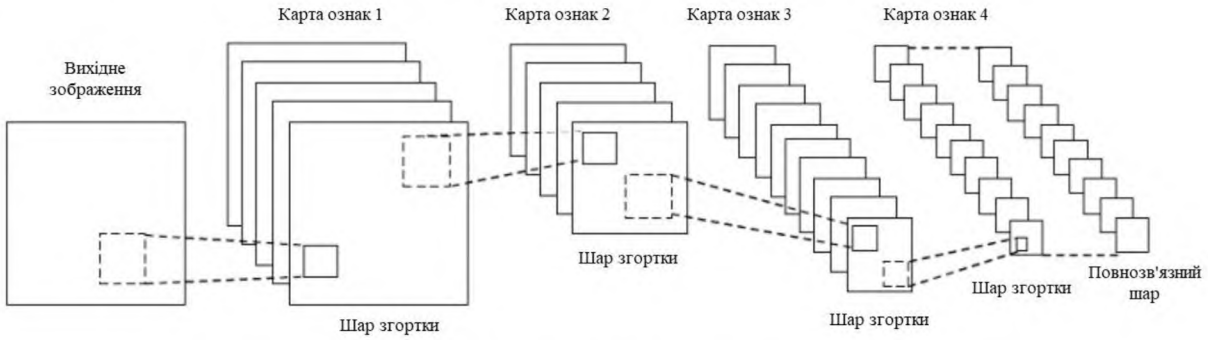

Рисунок 2.5 – Формування вектору ознак на основі SVM

Для цього перед виконанням класифікації для кожної згенерованої області застосовується пригнічення немаксимумів (алгоритм Non-Maximum Suppression), на основі якого локальні максимуми відзначаються як контур об'єкта (рис. 2.6). Пригнічення немаксимумов необхідне виключення згенерованих дублюючих областей інтересу кожного об'єкта на вихідному зображенні.

Для оцінки якості класифікації використовується показник, що дорівнює відношенню площі перетину прямокутника (області інтересу), отриманого в результаті детектування, та прямокутника з розмітки до площі їх об'єднання (Intersection over Union, IoU). Принцип роботи метрики IoU працює наведений на рис. 2.7.

Рисунок 2.6 – Алгоритм NMS

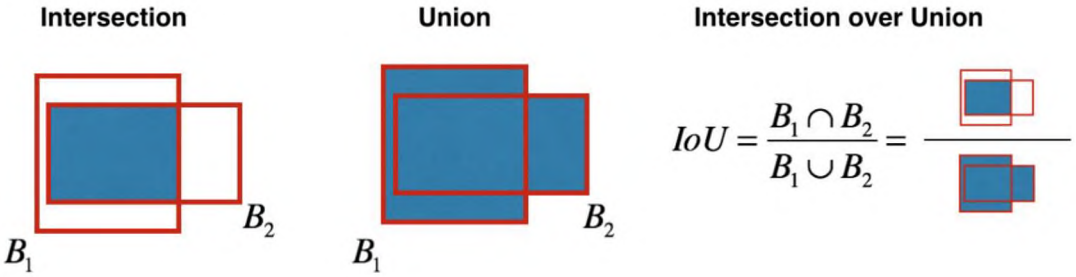

Рис. 2.7 – Метрика IoU

На початку можна виставити певний поріг для цієї метрики, і якщо передбачений bounding box буде вище цього порога, то у нього буде objectness рівний одиниці, а решта bounding boxes, у яких objectness нижче, будуть виключені. Ця величина objectness знадобиться, коли вважатимемо загальний confidence score (наскільки впевнені, що це саме потрібний об'єкт розташований всередині передбачуваного прямокутника) кожного конкретного об'єкта. Таким чином, вважається, що об'єкт виявлено правильно,

якщо цей показник перевищує деякий поріг, інакше вважається, що об'єкт не виявлено.

До недоліків мереж R-CNN, Fast R-CNN та Faster R-CNN слід віднести такі положення: не дивляться на картинку «повністю», а лише на окремі регіони; відносно повільні.

2.3 Архітектура YOLO

Модель YOLO (You Only Look Once) – це алгоритм виявлення об'єкта, який сильно відрізняється від алгоритмів, заснованих на регіоні, які ми розглянули вище. У YOLO одиночна згорткова мережа передбачає обмежуючі блоки та ймовірності класу для цих блоків. YOLO несе в собі філософію дивитися на картинку один раз, і за цей один перегляд (тобто один прогін картинки через одну нейронну мережу) робити всі необхідні визначення об'єктів. Як це відбувається?

Взагалі архітектура YOLO у перших блоках не сильно відрізняється за «логікою блоків» від інших детекторів, тобто на вхід подається картинка, далі створюються карти ознак за допомогою CNN (насправді, в YOLO використовується власна CNN під назвою Darknet-53), потім ці карти ознак певним чином аналізуються, видаючи на виході позиції та розміри bounding boxes та класи, яким вони належать. При цьому, Sparse Prediction – це повторення того, як 2-рівневі алгоритми працюють: визначають окремо регіони і потім класифікують ці регіони (рис. 2.8).

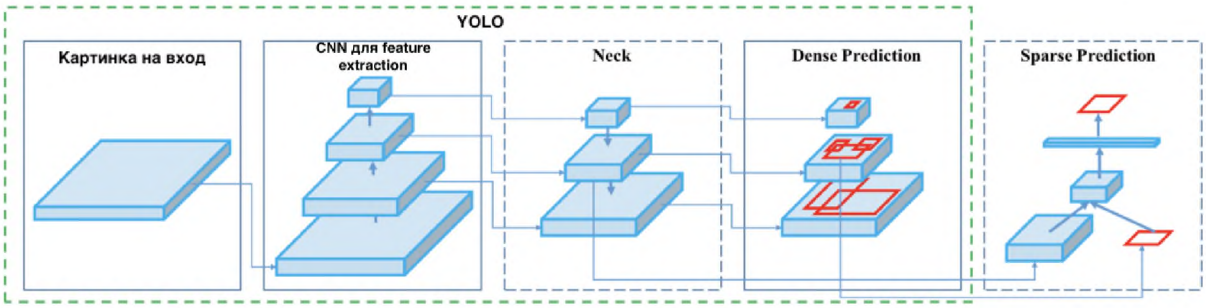

Рисунок 2.8 – Складові YOLO

Neck (або «шия») – це окремий блок, який створений для того, щоб агрегувати інформацію від окремих шарів із попередніх блоків (як показано на малюнку вище) для збільшення акуратності передбачення. Нарешті, те, що відрізняє YOLO від інших архітектур – блок під назвою Dense Prediction, який дозволив YOLO вирватися в лідери ефективності визначення об'єктів (рис. 2.9).

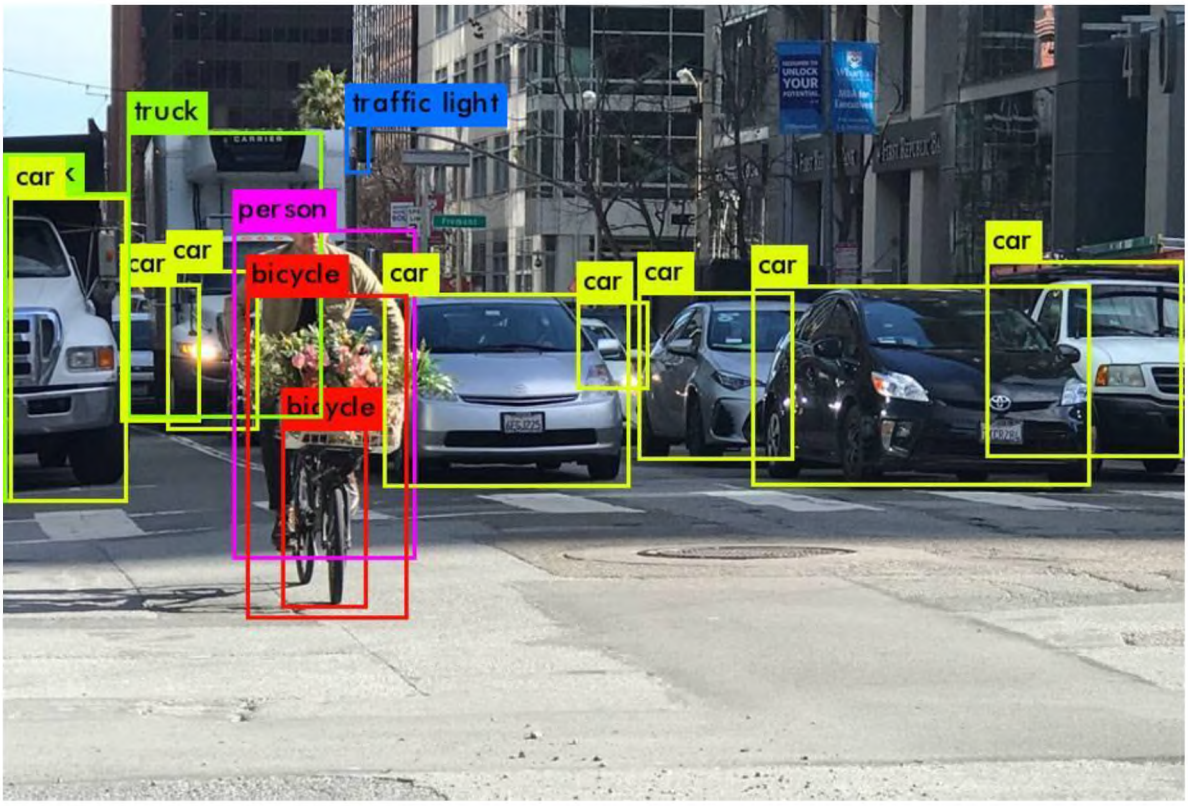

Рисунок 2.9 – Робота YOLO

При цьому, YOLO робить операції, коли навчається на даних.

Крок 1: Зазвичай, у зображення змінюють під розмір 416×416 перед початком навчання нейронної мережі, щоб їх подавати пакетами batch (для прискорення навчання).

Крок 2: Зображення поділяється на клітини розміром $a \times a$. Наприклад, у YOLOv3-4 прийнято ділити на клітини розміром 13×13 (рис. 2.10).

Далі виконується фокусування на цих клітинах, на які розділено зображення/кадр. Такі клітини, які називаються grid cells, є основою ідеї YOLO. Кожна клітка є «якорем», до якого прикріплюються bounding boxes.

Тобто навколо клітини малюються кілька прямокутників визначення об'єкта (оскільки незрозуміло, який форми прямокутник буде найбільш підходящим, їх малюють відразу кілька різних форм), та його позиції, ширина і висота обчислюються щодо центру цієї клітини – рис. 2.11. Як формуються bounding boxes навколо клітини? Як визначається їх розмір та позиція? В даному випадку, починає працювати anchor boxes (у перекладі – якірні коробки, або «якорні прямокутники»).

Рисунок 2.10 – Ділення зображення на клітини розміром 13×13

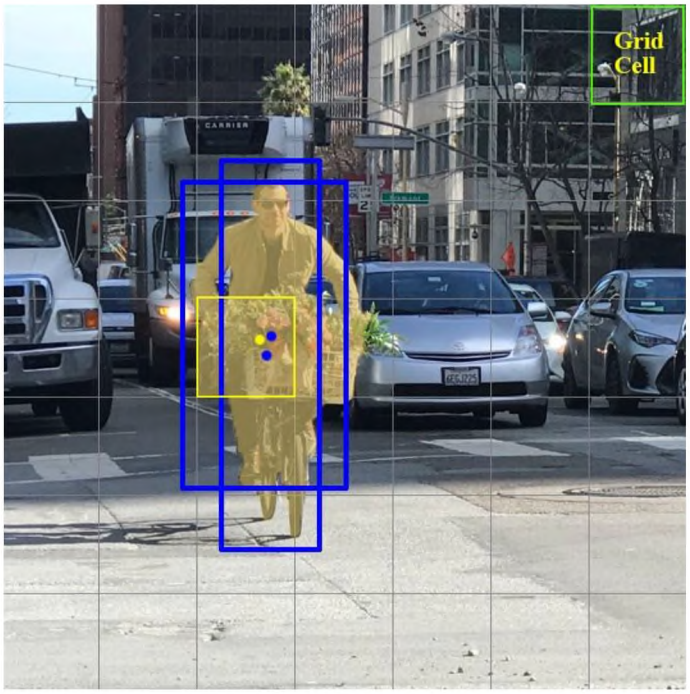

Рисунок 2.11 – Grid cells і центри bounding boxes

Вони задаються на самому початку або самим користувачем, або їх розміри визначаються виходячи з розмірів bounding boxes, які є в датасеті, на якому тренуватиметься YOLO (використовується *k*-means clustering та IoU для визначення відповідних розмірів). Зазвичай задають близько 3 різних anchor boxes, які будуть намальовані навколо (або всередині) однієї клітини (рис. 2.12).

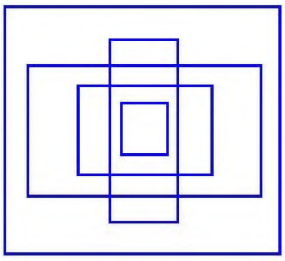

Рис. 2.12 – Anchor boxes

Крок 3. Зображення з датасету проходить через нейронну мережу (зауважимо, що окрім картинки в тренувальному датасеті повинні бути визначені позиції та розміри справжніх bounding boxes для об'єктів, які є на ній, тобто, проведення процедури «анотації», що робиться в основному вручну).

Для кожної клітини потрібно зрозуміти дві принципові речі.

1. Який з Anchor boxes, з 3 намальованих навколо клітки, підходить найбільше і як його можна трохи підправити для того, щоб він добре вписував об'єкт?

2. Який об'єкт знаходиться всередині цього Anchor box і чи є він взагалі?

Надалі визначимо, який повинен бути тоді Output у YOLO?

1. На виході кожної клітини прагнемо отримати результат, що наведений на рис. 2.13.

2. Output повинен включати кілька параметрів, що наведені на рис. 2.14. YOLO потрібно натренувати на те, щоб розпізнавати об’єкти на кадрі/зображенні. Подаємо зображення з датасету в YOLO, там відбувається

feature extraction на початку, а в кінці у виходить CNN шар, який розповідає про всі клітини, на які розділено зображення.

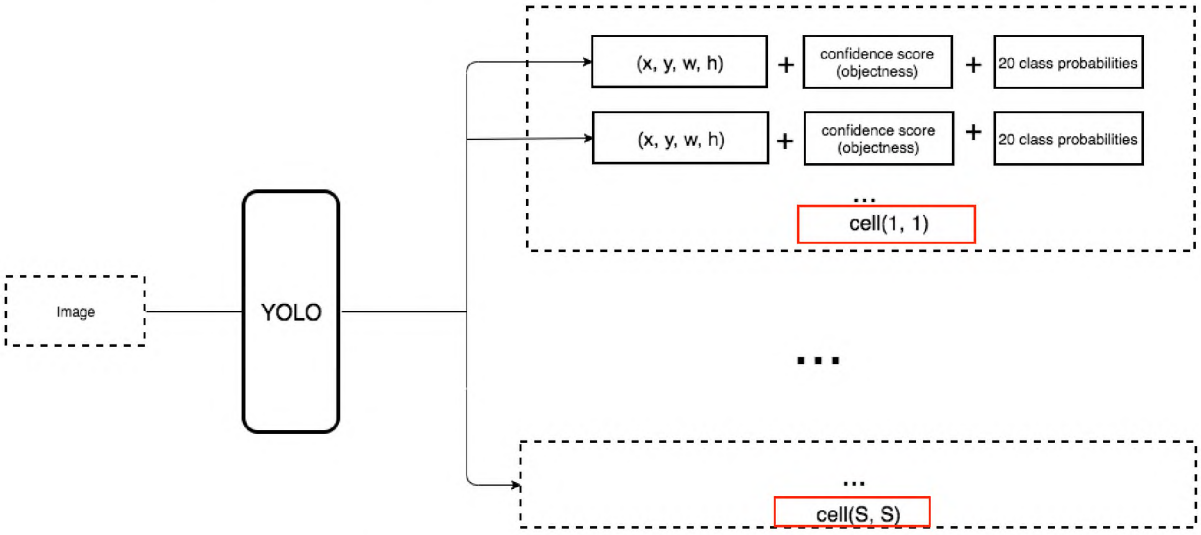

Рисунок 2.13 – Output YOLO

Якщо цей шар «розповідає неправду» про клітини на зображенні, то має бути велике значення Loss, щоб потім його зменшувати при подачі в нейронну мережу наступних зображень. Щоб було зрозуміло, є дуже проста схема з тим, як YOLO створює цей останній шар (рис. 2.15).

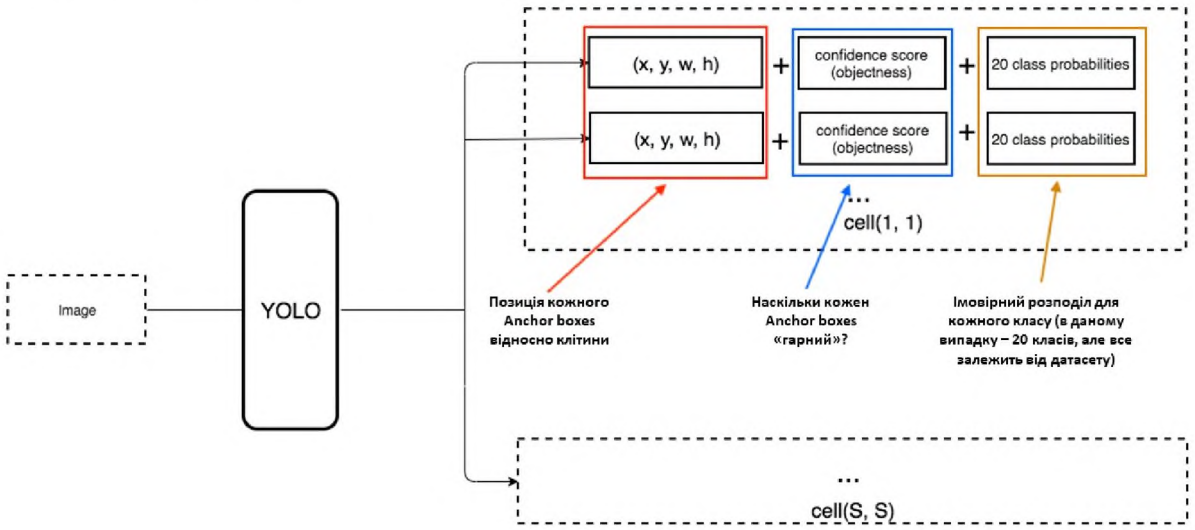

Рисунок 2.14 – Параметри Output

Як видно, цей шар, розміром 13×13 (для зображень початкового розміру 416×416) «розповідає» про кожну клітинку на зображенні. З цього останнього

шару і дістається інформація, яка потрібна. YOLO передбачає 5 параметрів (для кожного anchor box для певної клітини) – рис. 2.16:

$$\Pr(object) \cdot IOU(b, object) = \sigma(t_0), \tag{2.1}$$

де $b \in \{b_x \quad b_y \quad b_w \quad b_h\}$, $b_x = \sigma(t_x) + c_x$, $b_y = \sigma(t_y) + c_y$, $b_w = p_w e^{t_w}$, $b_h = p_h e^{t_h}$,

t_x, t_y, t_h, t_w – те, що пророкувала YOLO;

c_x, c_y – координата верхньої лівої точки потрібної grid cells;

p_w, p_h – ширина та висота конкретного anchor box;

b_x, b_y, b_h, b_w – параметри для пророкованого bounding boxes;

$\sigma(t_0)$ – параметр confidence score.

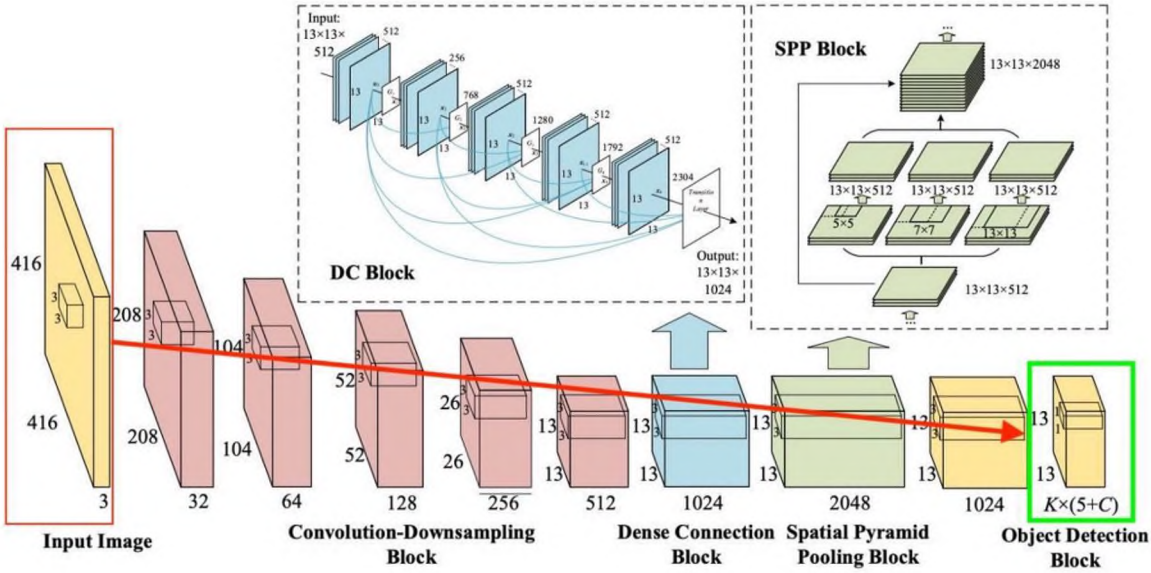

Рисунок 2.15 – Формування останнього шару в YOLO

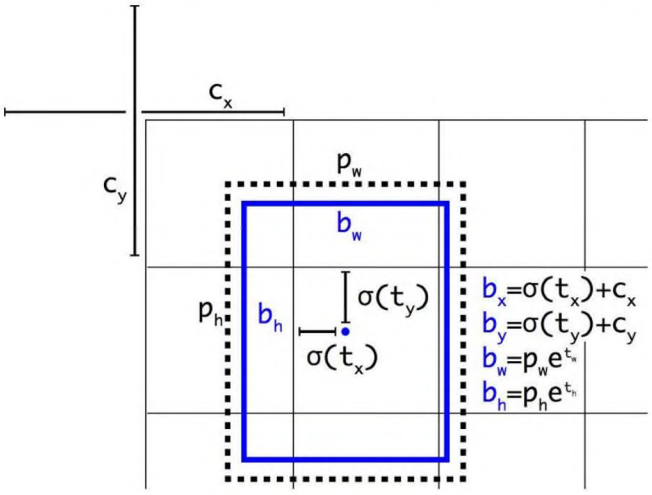

Рисунок 2.16 – Параметри, що YOLO передбачає

Як можна зрозуміти, завдання YOLO – максимально точно передбачити параметри, щоб максимально точно визначати об'єкт на зображенні. А confidence score, що визначається для кожного передбачуваного bounding box, є фільтром для того, щоб відсіяти зовсім неточні прогнози. Для кожного передбачуваного bounding box множимо його IoU на ймовірність того, що це певний об'єкт (імовірнісний розподіл розраховується під час навчання нейронної мережі), беремо найкращу ймовірність з усіх можливих, і якщо число після множення перевищує певний поріг, то можемо залишити цей передбачений bounding box на зображені. Далі, коли залишилися тільки передбачені bounding boxes з високим confidence score, передбачення (якщо їх візуалізувати) можуть виглядати приблизно так, як наведено на рис. 2.17.

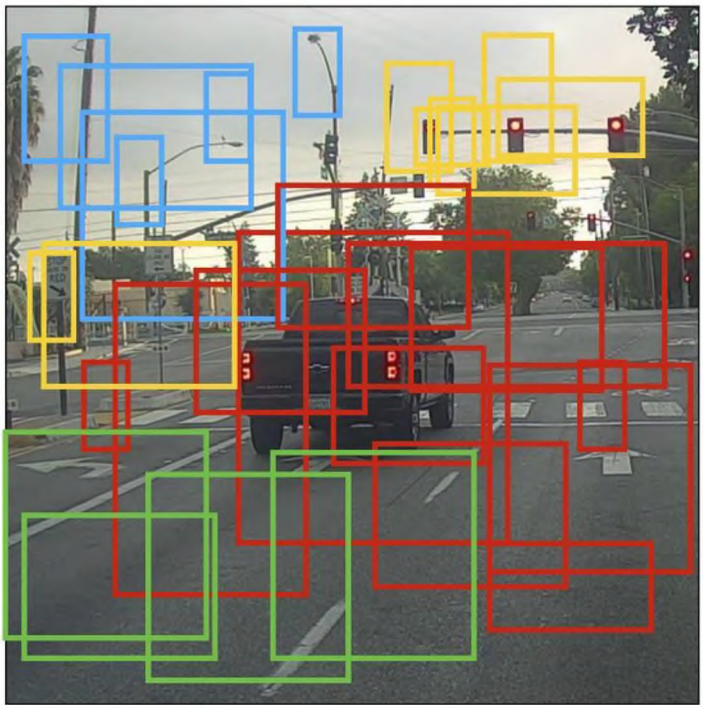

Рисунок 2.17 – Передбачені bounding boxes

Далі можемо використовувати техніку NMS (Non-Max Suppression), щоб відфільтрувати bounding boxes таким чином, щоб для одного об'єкта був тільки один передбачений bounding box (див. рис. 2.6).

Потрібно також знати, що YOLOV3-4 пророкують на 3-х різних скейлах. Тобто картинка ділиться на 64 grid cells, на 256 клітин та на 1024 клітини, щоб

також бачити маленькі об'єкти. Для кожної групи клітин алгоритм повторює необхідні дії під час передбачення/навчання, описаних зверху. У YOLOV4 було використано багато технік для збільшення точності моделі без сильної втрати швидкості. Але для самого передбачення залишили Dense Prediction таким самим, як і у YOLOV3.

Таким чином, YOLO на кілька порядків швидше (45 кадрів за секунду), ніж інші алгоритми виявлення об'єктів. Обмеження алгоритму YOLO полягає в тому, що він бореться з невеликими об'єктами на зображенні. Наприклад він може відчувати труднощі при виявленні зграї птахів. Це є просторовими обмеженнями алгоритму.

2.4 Варіант використання YOLO для Object Detection пожеж

В [54] була запропонована вдосконалена модель YOLOv4 (MoAm-YOLOv4), яка поєднує в собі легку мережу CNN та механізм уваги. Це дозволило покращити продуктивність моделі виявлення диму пожежі та досягти кращого балансу між точністю та швидкістю виявлення.

YOLOv4 в основному покращила YOLOv3 [59]. Структуру мережі можна розділити на чотири модулі (рис .2.18): вхід, хребет, шия та голова. Сторона вводу – це вхідне зображення, цей етап містить етап попередньої обробки зображення; Backbone – це магістральна мережа для вилучення ознак, тут магістральна мережа – це CSPDarknet53; шия розташована між хребтом і головою, яка містить дві частини: додатковий модуль SPP і PANet. Після того, як SPP досягає останнього вихідного рівня магістральної мережі, максимальна операція об’єднання чотирьох різних розмірів згорткового ядра становить 1×1 (тобто без обробки), 5×5, 9×9 і 13×13. Вхід цього додаткового модуля може допомогти відокремити найважливіші функції контексту без зниження швидкості роботи моделі. PANet використовується для об’єднання функцій, поєднуючи елементи дрібних і глибоких мереж за допомогою вибірки вгору та

вниз. Головний вихід завершує кінцеві результати виявлення, що є унікальним модулем алгоритму YOLO порівняно з іншими алгоритмами виявлення цілей.

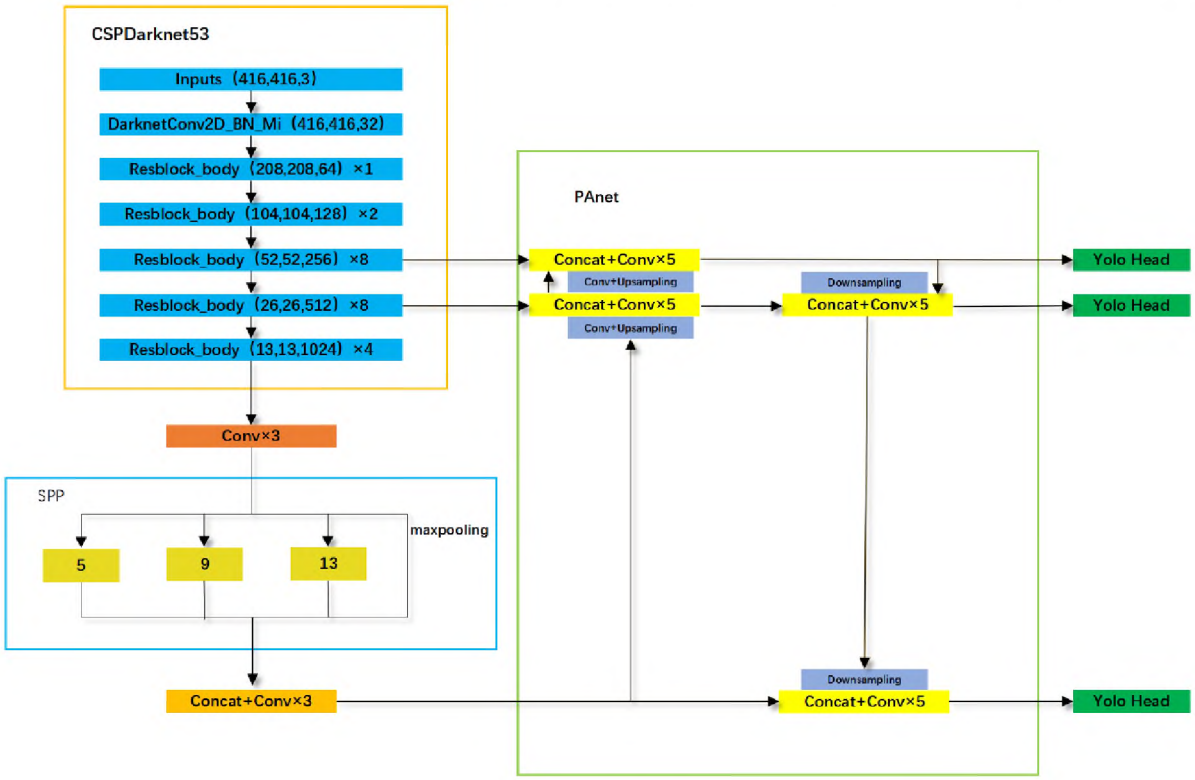

Рисунок 2.18 – Структурна схема мережі YOLOv4

Розмір пакету-кандидата є корисним для регресії коробки прогнозування диму пожежі. В експерименті використовується метод кластеризації *k*-середніх для генерації блоків-кандидатів, завдяки якому розмір блоку-кандидата можна наблизити до розміру зображення набору даних. Розмір зображення мережного входу становить 416×416, а 9 блоків-кандидатів, згенерованих *k*-середніми, мають такі розміри: 33×141, 45×164, 58×104, 66×165, 97×168, 129×37, 165×196, 177×76, 250×310.

Покращення спрощеної структури мережі. Магістраль алгоритму виявлення більше не використовує CPSDarknet53 і замінена мережею MobilenetV1. Google запропонував mobilenetV1 у 2017 р., та її модель має невеликий розмір і невелику кількість параметрів, що дуже підходить для застосування на мобільних пристроях. Основна ідея мережі MobilenetV 1 є глибоке розділення для виконання згортки. Вона також використовує множник

ширини та роздільної здатності для зменшення кількості параметрів, що забезпечує кращу пропускну здатність даних із мінімальною точністю.

Згортка, що розділяється по глибині. Глибока роздільна згортка є варіантом звичайних операцій згортки, які можуть замінювати різні згортки для формування згорткових нейронних мереж [60]. Вона включає поглиблену та поточкову згортку, які поєднують ці два методи згортки для завершення виділення ознак зображення. Згортка з роздільною глибиною значно зменшує кількість параметрів, так що мережа також може відповідати вимогам виявлення в реальному часі на CPU.

Припускаючи, що розмір вхідного зображення дорівнює $D_F \times D_F \times M$, розмір ядра згортки дорівнює $D_K \times D_K$. Розмір вихідної карти ознак позначається як $D_F \times D_F \times N$, де M є кількістю вхідних даних. передбачається кількість каналів шару, а N – це кількість каналів вихідного шару, тут ширина та висота вхідних і вихідних карт ознак узгоджуються. Для традиційних операцій згортки сума обчислення параметра:

$$D_K \times D_K \times M \times N \times D_F \times D_F . \tag{2.2}$$

Для операцій згортки по глибині сума обчислення параметра дорівнює:

$$D_K \times D_K \times M \times D_F \times D_F . \tag{2.3}$$

Для операції поточкової згортки величина обчислення параметра дорівнює:

$$M \times N \times D_F \times D_F \tag{2.4}$$

Для операцій згортання з роздільною глибиною обчислення параметра становить:

$$D_K \times D_K \times M \times D_F \times D_F + M \times N \times D_F \times D_F \tag{2.5}$$

Ділення (2.5) на рівняння (2.2) дає результат:

$$\frac{D_K \times D_K \times M \times D_F \times D_F + M \times N \times D_F \times D_F}{D_K \times D_K \times M \times N \times D_F \times D_F} = \frac{1}{N} + \frac{1}{D_K^2} . \tag{2.6}$$

Як видно з рівняння (2.6), коли кількість вихідних каналів є великою (якщо розмір ядра згортки використовується 3×3), то операція глибокої роздільної згортки може зменшити обчислення параметра майже у 8-9 разів порівняно з традиційною операцією згортки, що значно зменшує кількість обчислень.

Множення ширини. Для побудови моделей, які можуть задовольнити менші обчислювальні витрати, у MobilenetV1 вводиться параметр α, який називається Width Multiplier. Функція α полягає в уточненні мережі на кожному рівні, і для заданого рівня мережі кількість вхідних каналів M стає αM, а кількість вихідних каналів A стає αN. Після додавання гіперпараметру α обчислення згортки, що розділяється на глибину, має вигляд:

$$D_K \times D_K \times \alpha M \times D_F \times D_F + \alpha N \times \alpha M \times D_F \times D_F . \quad (2.7)$$

Значення а в рівнянні (2.7) коливаються від (0, 1] і зазвичай від 1, 0,75, 0,5 або 0,25. Якщо всі рівні в мережі помножити на α , розмір моделі мережі падає близько до α^2 у рази більше початкової мережі, а обчислювальний обсяг зменшується до α^2 разів від початкового розміру.

Множник роздільної здатності. Ще один параметр, представлений MobilenetV1, називається множником роздільної здатності, який використовується для керування роздільною здатністю вхідного зображення. Якщо розмір вхідного зображення 416×416, а гіперпараметр $\beta = 0,5$, розмір вхідного зображення подвоїться до 208×208. Обидва гіперпараметри α і β додаються до мережі MobilenetV1, і кількість параметрів операції згортки з глибоким розділенням стає:

$$D_K \times D_K \times \alpha M \times \alpha D_F \times \beta D_F + \alpha M \times \alpha N \times \beta D_F \times \beta D_F . \quad (2.8)$$

Діапазон β в формулі дорівнює (0, 1). Припустімо, що роздільна здатність зображення вхідного шару помножена на гіперпараметр β. У такому випадку, розмір моделі мережі не зміниться, він залишається таким самим розміром, як і у початкової мережної моделі, але кількість обчислень моделі

зменшиться в 2 рази, ніж у початковій моделі. Неважко побачити, що один із двох гіперпараметрів стискає мережну модель, а інший зменшує обчислення. Після додавання двох параметрів одночасно можна досягти мети скорочення моделі та малої кількості.

Покращення продуктивності голови. Під час згортки інформація про перешкоди в каналі заважає мережі зосередитися на інформації, я ка є важливою для завдання. Це призводить до зниження продуктивності. На даний час, механізм уваги широко використовується в згорткових нейронних мережах. Різні типи механізму уваги каналу можуть зменшити вплив інформації про перешкоди різним ступенем після коригування відповідних ваг. SENet (Squeeze and Excitation Networks) – це механізм звернення уваги каналу, який зазвичай використовується (рис. 2.19).

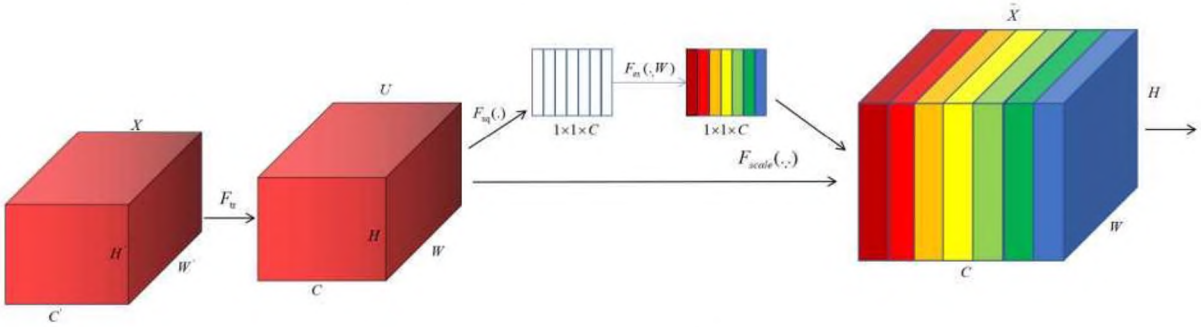

Рисунок 2.19 – Схема SENet

F_{tr} є традиційною згортковою структурою, X і U є входом і виходом відповідно для F_{tr}, розмір $C' \times W' \times H'$ і $C \times W \times H$, тобто U має канал C', розмір кожного каналу $H' \times W'$, U має канал C, розмір кожного каналу $H \times W$. Весь процес можна розділити на три етапи. Перший крок називається процесом стиснення, глобальним середнім об'єднанням U, який є процесом $F_{sq}(.)$. Вагові коефіцієнти вихідного каналу такі:

$$Z = F_{sq}(u_c) = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} u_c(i, j) \tag{2.9}$$

де Z – число розміру $1\times1\times C$;

(i, j) – точка відповідних координат на графіку ознак розміру.

Другий крок називається процесом збудження, який складається з двох повністю пов'язаних операцій, тобто $F_{ex}(., w)$. Формула для ваги вихідного каналу має вигляд:

$$S_c = F_{ex}(Z, W) = Sigmoid(W_2 \times ReLU(W_1, Z)), \tag{2.10}$$

де S_c – згенерована вага каналу, розмірність $1\times1\times C$;

W_1 – має розмірність $C / r \times C$;

W_2 – має розмірність $C \times C / r$;

r – параметр масштабування.

Останній крок називається операцією перезважування, що відповідає $F_{scale}(.,.)$. Відповідні вхідні характеристики виходу каналу зважуються шляхом множення, щоб скинути вхідні характеристики, роблячи витягнуті характеристики більш спрямованими. При цьому вираз має вигляд:

$$\hat{X} = F_{scale}(X_c, S_c) \otimes S_c, \tag{2.11}$$

де $\hat{X}$ – вихідний результат після обробки SENet;

$\otimes$ – символ вказує на добуток по елементах.

SENet можна додати до будь-якого згорткового рівня. В цілому, чим

більше разів буде вставлено SENet, тим більше параметрів він принесе в мережу.

Додаток до механізму уваги. Вбудований механізм уваги SENet перед виходом YOLO Head може зменшити вплив інформації про перешкоди на продуктивність мережі та не принесе в мережу занадто багато параметрів. Таким чином, модель збільшується лише на 0,33 МБ порівняно з оригінальною мережею. Схематична діаграма показана на рис. 2.20. Модуль механізму уваги SENet додається до трьох вихідних каналів PANet, а параметр масштабування

r, зазвичай, приймається рівним 2, 4, 8, 16. В [54] показано, що найкращий

ефект досягається при $r = 8$.

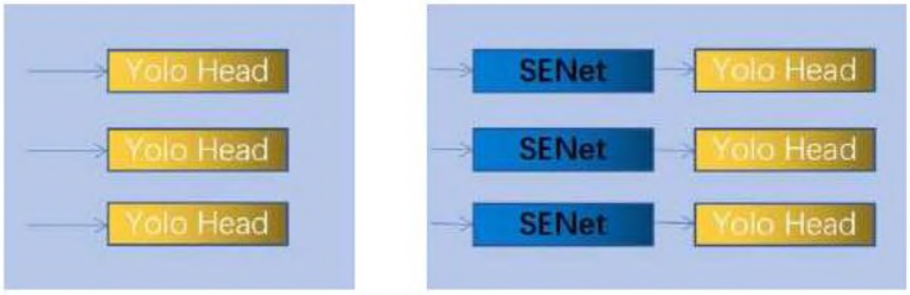

Рисунок 2.20 – Порівняння алгоритму до та після додавання механізму уваги

Інтегруючи вищевказані інноваційні моменти, остаточна вдосконалена архітектура мережі MoAm-YOLOv4 відповідає рис. 2.21.

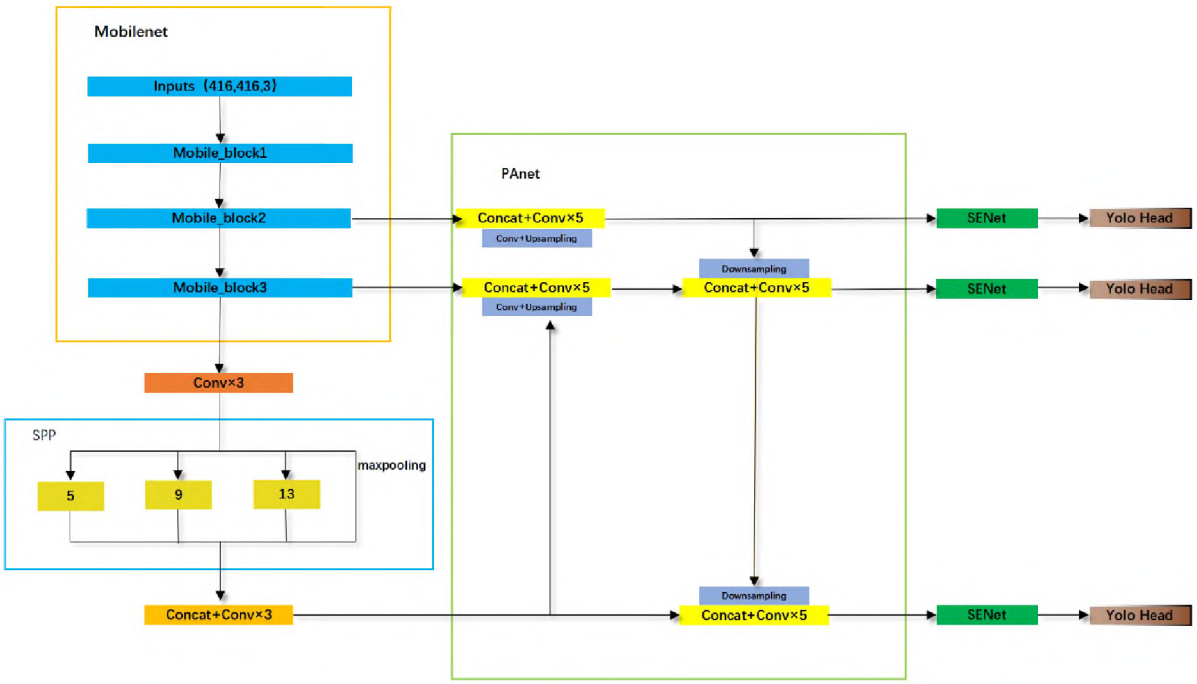

Рисунок 2.21 – Схема структури мережі MoAm-YOLOv4

Результати досліджень [54] показують, що вдосконалена YOLO може точно виявляти дим від пожежі в реальному часі. Для оцінки вибирались показники, що включають точність (Precision), повноту (Recall), середню точність (mAP), кількість виявлених кадрів за секунду (FPS) і розмір моделі. Формули розрахунку Precision і Recall відповідають виразам:

$$Precision = \frac{TP}{TP + FP}, \tag{2.12}$$

$$Recall = \frac{TP}{TP + FN}, \tag{2.13}$$

де TP – позитивний зразок, прогнозований моделлю позитивний клас;

FP – негативний зразок, прогнозований позитивний клас;

FN – позитивний зразок, попереднього детектований як негативний клас.

Але як Precision, так і Recall враховують лише один фактор і не можуть повністю виміряти продуктивність моделі. Параметр mAP – це середнє значення точності кожної категорії, яке є більш репрезентативним, ніж два вищезазначені параметри.

Щоб перевірити ефективність запропнованого підходу, авторами було проведено загалом п’ять наборів порівняльних експериментів із зібраним набором даних про дим від пожеж, включаючи YOLOv4, YOLOv4-tiny, замінивши лише мережу на MobilenetV1, вставивши механізм уваги SENet лише перед головним виходом і їх комбінація. У табл. 2.1 наведено порівняння результатів експерименту.

Таблиця 2.1 – Порівняння результатів п'яти моделей

Model	Precision, %	Recall, %	mAP, %	FPS	Size, MB
YOLOv4	92,08	89,84	93,72	53,19	245,52
YOLOv4-tiny	89,54	86,99	92,96	100,74	23,10
Mobilenetv1-YOLOv4	91,02	90,65	92,94	85,30	48,42
YOLOv4+SENet	92,63	91,87	94,54	53,03	245,85
MoAm-YOLOv4	93,28	90,24	93,45	80,33	45,85

Як видно, після заміни мережі з функціями на спрощену мережу MobilenetV1. mAP моделі зменшився майже на один відсотковий пункт, але розмір моделі зменшився з 245,52 МБ до 48,42 МБ, а швидкість виявлення FPS також було значно покращено. Після вбудови механізму уваги SENet розмір моделі збільшився лише на 0,33 МБ, але mAP збільшився на 0,82 %.

Як відомо, YOLOv4-tiny – це полегшена архітектура YOLOv4. На додаток до оптимізації магістральної мережі, він також зменшує вихідні прогнози з 3 до 2, зменшує блоки кандидатів з 9 до 4 і використовує традиційний FPN замість PANet для об’єднання функцій. Робота цих стовпців

різко скорочує кількість параметрів мережі, а розмір моделі становить лише 23,10 МБ. У MoAm-YOLOv4, незважаючи на те, що отримано вдвічі більший розмір моделі, ніж YOLOv4-tiny, Precision, Recall і mAP переважають. Як видно, порівняно з оригінальною YOLOv4, MoAm-YOLOv4, не отримав значного зниження, але Precision, Recall і mAP зросли на 1,8, 1,6 і 51 %, відповідно, а розмір моделі зменшився на 80 %, це еквівалентно за рахунок лише не великої точки mAP. Інші показники значно покращені (рис. 2.22).

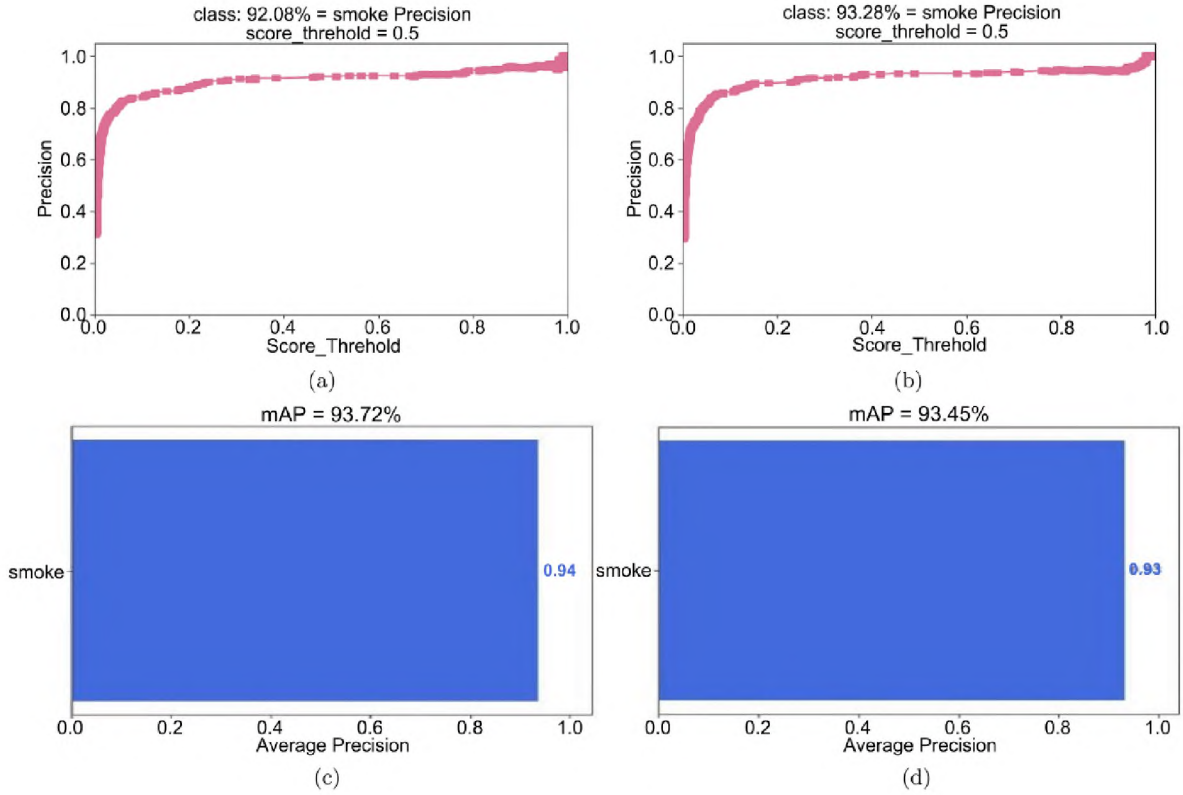

Рисунок 2.22 – Порівняння mAP і Precision на тестовому наборі

Експериментальні результати показують, що запропонований покращений алгоритм має певні переваги у підтримці високої точності виявлення диму від лісової пожежі та досягненні значною мірою балансу швидкості та точності, експериментального ефекту виявлення, як показано на рис. 2.23.

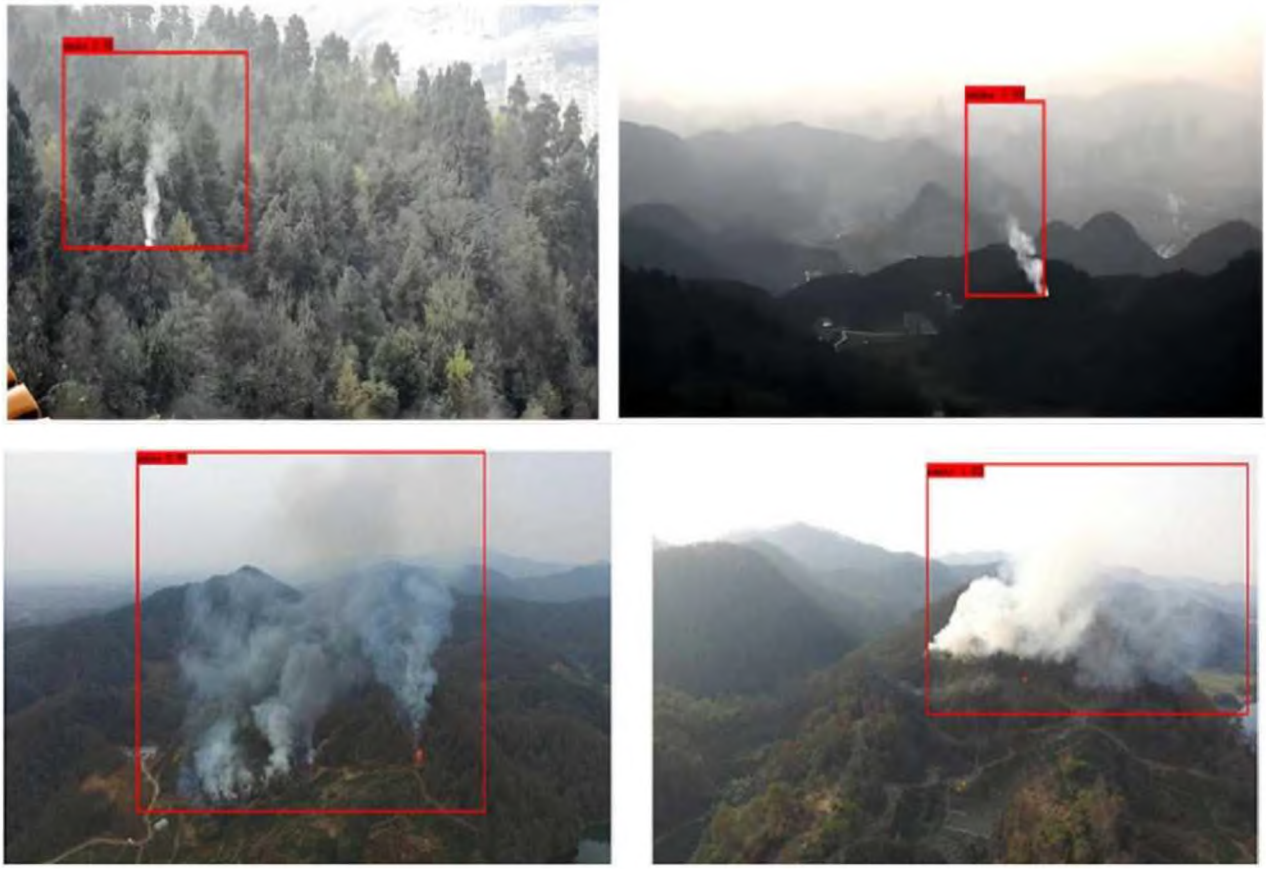

Рисунок 2.23 – Виявлення диму

Таким чином, розглянута модифікація YOLOv4 має розмір моделі становить лише 45,58 МБ, mAP на рівні 93,45 %, а точність може сягати 93,28 %. Хоча модель досягла гарних результатів, все ще є недоліки, які можна покращити, виходячи з наступних положень.

1. Під час навчання та тестування було відібрано лише понад 2000 зображень. Враховуючи різноманітність середовища пожеж, зібрані дані не повністю враховують фонове середовище. Набору даних бракує різноманітності. Частина набору даних походить із загальнодоступних онлайн-ресурсів. Існує два типи зображень: частина – це справжній дим пожежі, частина – синтетичні дані. Щоб розширити датасет, дані подвоюються: 1/3 зображення перевертається ліворуч, 1/3 зображення перевертається вниз і 1/3 зображення перевертається праворуч. Зображення були анотовані за допомогою інструменту LabelImg, що дало загалом 2000 зображень диму. Таким чином, щоб підвищити надійність, набори даних потрібно збільшити.

2. Дослідження зосереджено на середовищі, що розглядалось лише вдень і не є ситуацію вночі.

3. Після заміни спрощеної мережі швидкість виявлення була значно покращена, але є ще багато можливостей для підвищення точності. Точність можна підвищити, не змінюючи магістральну мережу за допомогою різних методів об'єднання функцій. Алгоритм відповідає методу об'єднання функцій PANet, і в майбутньому алгоритм може бути удосконалений у напрямку об'єднання функцій.

Висновки до розділу 2

Для реалізації Object Detection існує кілька поширених архітектур нейронних мереж. Вони в основному поділяються на 2-рівневі (R-CNN, Fast R-CNN та Faster R-CNN), та однорівневі (YOLO). До недоліків мереж R-CNN, Fast R-CNN та Faster R-CNN слід віднести такі положення: не дивляться на картинку «повністю», а лише на окремі регіони; відносно повільні.

У YOLO одиночна згорткова мережа передбачає обмежуючі блоки та ймовірності класу для цих блоків. YOLO несе в собі філософію дивитися на картинку один раз, і за цей один перегляд (тобто один прогін картинки через одну нейронну мережу) робити всі необхідні визначення об'єктів.

YOLO на кілька порядків швидше (45 кадрів за секунду), ніж інші алгоритми виявлення об'єктів. Обмеження алгоритму YOLO полягає в тому, що вона бореться з невеликими об'єктами на зображенні. Це є просторовими обмеженнями алгоритму.

Порівняно з оригінальною YOLOv4, MoAm-YOLOv4, не отримав значного зниження, але Precision, Recall і mAP зросли на 1,8, 1,6 і 51 %, відповідно, а розмір моделі зменшився на 80 %. Дана модифікація YOLOv4 має розмір моделі становить лише 45,58 МБ, mAP на рівні 93,45 %, а точність

може сягати 93,28 %. Основні зміни стосуються заміни магістральної мережі Darknet-53 на мережу MobilenetV1.

РОЗДІЛ 3
РЕКОМЕНДАЦІЙ ЩОДО ВИКОРИСТАННЯ ЗАПРОПОНОВАНОЇ МОДЕЛІ ОБРОБКИ ДАНИХ, ЩО ОТРИМАНІ З БПЛА

3.1 Анотація зображень

Для того, щоб нейронна мережа вміла розпізнавати об'єкти та образи на зображенні, потрібно їй спершу «показати», що саме там знаходиться. Для цього і призначено розмітку.

Розмітка зображень (Image Annotation, анотування зображень, анотація зображень) є невід'ємною частиною розробки ШІ, і це одне з основних завдань CV. Анотовані зображення потрібні як вступні дані для навчання нейронних мереж: розпізнавання об'єктів на зображеннях дозволяє комп'ютерам «бачити» навколишній світ подібно до людини.

Ручна розмітка об'єктів на зображеннях – трудомістка і досить затратна задача, особливо якщо потрібно розмітити великі набори даних.

Автоматична анотація зображень (Automatic Image Annotation, також відома як автоматичне маркування зображень) – це процес, у якому комп'ютер автоматично надає метадані цифрового зображення (підписи або мітки), використовуючи відповідні ключові слова для опису його візуального змісту. Існуючі алгоритми розмітки можна розділити на дві категорії:

- методи навчання на основі моделей – вони досліджують кореляцію між візуальними характеристиками та їх семантичним значенням для виявлення функції відображення за допомогою машинного навчання або моделей представлення знань для розмітки зображень;
- моделі на основі БД – вони одразу видають послідовність ймовірних міток відповідно до вже анотованих зображень у БД.

Інструменти для розмітки на основі нейронних мереж дозволяють виділяти об'єкти набагато швидше та ефективніше, обробляти значно більшу кількість зображень, автоматизують основну масу ручних завдань, їх можна додатково навчати більш точно розпізнавати нові зображення (рис. 3.1).

Рисунок 3.1 – Автоматична анотація зображень на основі нейронних мереж

На даний час, існують Open Source інструменти для розмітки зображень. Їх також можна використовувати як базу для розробки кастомних інструментів.

LabelImg – базовий інструмент анотування. Безкоштовний графічний інструмент для розмітки зображень написаний на Python, який використовується для виділення об'єктів на картинці. Анотації можна зберегти як файли XML у форматі PASCAL VOC/ YOLO. LabelImg дозволяє створювати прямокутники (bounding boxes) для розмітки об'єктів у графічному інтерфейсі Qt.

CVAT – універсальний редактор для розмітки даних. Це безкоштовний інструмент з відкритим вихідним кодом для розмітки цифрових зображень та відео, а також зручна підготовка наборів даних для алгоритмів CV. Він дозволяє розмічати дані для кількох завдань машинного навчання:

розпізнавання об'єктів, класифікації та сегментації зображень. CVAT підтримує низку додаткових компонентів: Deep Learning Deployment Toolkit (компонент у складі OpenVINO), NVIDIA CUDA Toolkit, TensorFlow Object Detection API та ін.

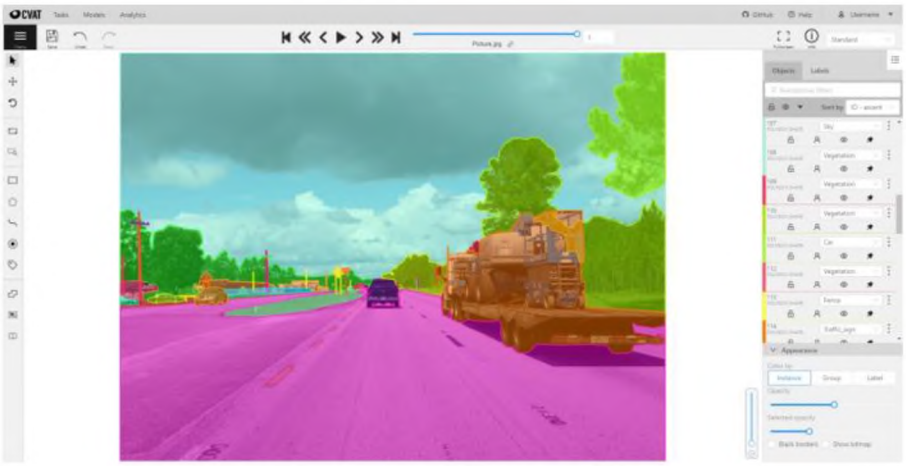

Рисунок 3.2 – CVAT

Auto_Annotate – 2D-розмітка за допомогою нейронної мережі (рис. 3.3). Це рішення з відкритим кодом для автоматизованої розмітки зображень. Клас Python під назвою «generate XML» розмічає зображення через вивід попередньо навченої моделі, щоб визначити позиції прямокутників bounding boxes. Скрипт також використовує репозиторій TensorFlow для навчання. Отримані в результаті зображення (з розміткою bounding boxes) і XML можна відкрити в LabelImg.

Рисунок 3.3 – Auto_Annotate

Комерційні утиліти для розмітки зображень.

Hasty.ai – автоматична розмітка даних. Платформа пропонує кілька інструментів анотування на основі ШІ (DEXTR, прогнозування класифікації, помічник з виявлення об'єктів та сегментації та ін.) поряд із інструментами ручної розмітки. Автоматично нанесені контури об'єктів можна коригувати вручну для підвищення точності та якості розмітки. Також можна використовувати власні набори даних для подальшого навчання алгоритмів виявлення об'єктів і сегментації.

V7 Darwin – піксельна розмітка зображень на базі нейронної мережі. Автоматизований інструмент розмітки на основі ШІ, який працює з усіма даними та автоматично генерує полігональні та попіксельні маски. Можна встановити область для розпізнавання – алгоритм глибокого навчання визначить найбільш помітний об'єкт або його видиму частину і нанесе розмітку. Завдяки використанню ШІ, Darwin може генерувати точні маски з першого разу.

Dataloop – розмітка великих масивів даних. Хмарна платформа для анотування, що складається з багатьох додатків для автоматизації процесу підготовки даних для рітейлу, робототехніки, автономних транспортних засобів, точного землеробства та ін. Інструменти розмітки Dataloop працюють із усіма видами зображень (зображення, відео). Можна інтегрувати моделі Deep Learning та автоматизувати процес розмітки, використовуючи попередньо навчені класи. Фахівець з розмітки даних потім лише перевіряє точність контурів та вносить необхідні зміни, що прискорює процес анотації.

Таким чином, розмітка даних на зображеннях або відео необхідна моделям глибинного навчання для надання інформації про те, що є на зображенні.

3.2 Порівняльна оцінка точності нейронних мереж Object Detection

З метою підтвердження висунутих в роботі теоретичних положень було проведено моделювання глибокого навчання на прикладі нейронної мережі YOLO4 без модифікацій. Для цього використовується попередньо навчена YOLO4, для якої задається лише один клас об'єктів – вогонь. Загальна схема застосування попередньо навченої YOLO4 для вирішення завдання Object Detection наведена на рис. 3.4. Основне завдання полягало в пошуку параметрів оптимізатору, виборі розміру пакету batch, та кількості епох навчання.

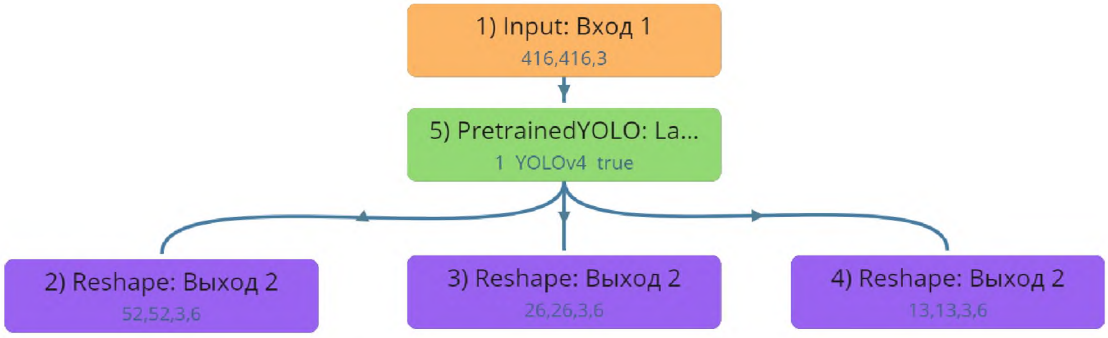

Рисунок 3.4 – YOLO4 для завдання Object Detection (об’єкт – вогонь)

Дана нейронна мережа має 61640962 параметри, з яких можна було проводити тренування за 61588354 параметрами. Розмірність вихідних тензорів шарів Reshape 52×52×3×6, 26×26×3×6, 13×13×3×6 зумовлена трьома розмірами боксів (52×52, 26×26, 13×13) та 3-кольоровим представленням RGB зображень. Розмір параметра 6 у розмірності вихідних тензорів пояснюється тим, що до кількості класів 1 мають бути додані 4 координати боксу і мітка класу, тобто 5 додаткових величин.

Датасет складається з 3755 розмічених фотогорафій, у формат фотознімків 416×416 пікселів. Баланс присутності вогню в тренувальній виборці (це особливість фотознімків з вогнем) – 2790, а в перевічній – 965. Відсоток простору, зайнятого вогнем на знімках у тренувальній вибірці – 11,7, а у перевірочній – 11,3.

Модель реалізовувалась в Google ColabPro+ з параметрами, що наведені на рис. 3.5. В якості оптимізатору використовувався Adam з початковим кроком навчання – 0,0001 і його кінцевим значенням – 0,000001. Витрати часу на навчання коливались в межах до 2,5 год. (залежить від розміру пакету batch, який приймав значення від 4 до 16). Найбільш вдалі результаті були отримані для $batch = 4$. При цьому було досягнуто $mAP50 = 76{,}9$ % (на 24-ій епосі) – рис. 3.6. Деякі приклади роботи запропонованої моделі наведено на рис. 3.7.

Рисунок 3.5 – Використовувані обчислювальні ресурси

Рисунок 3.6 – Результати моделювання

Рисунок 3.7 – Приклад роботи моделі на перевірочній виборці

Аналогічним чином проводилось дослідження роботи запропонованої моделі при вирішення завдання Object Detection (об’єкт – дим). В даному випадку, *mAP*50 не перевищував 68 % – (рис. 3.8).

Рисунок 3.8 – Результат роботи моделі

Такий результат пояснюється тим, що використовуваний датасет містить мало зображень посівів з димом (вогнем). Тобто, при якісній зборці навчальної бази можливо досягти потрібного рівня точності роботи нейронної мережі.

Результати якісної оцінки розглянутої моделі глибокого навчання YOLO4 для завдання Object Detection (об’єкт – вогонь) наведено на рис. 3.9.

Рисунок 3.9 – YOLO4 для завдання Object Detection (об’єкт – вогонь)

В цілому, отримані результати свідчать про можливість забезпечення інтелектуальної функції моніторингу пожежної безпеки посівів за допомогою БПЛА на базі запропонованих моделей глибокого навчання нейронних мереж Object Detection.

3.3 Техніко-економічне обґрунтування прийнятих рішень

На даний час, крім широко поширеної 5-ої версії YOLO (рис. 3.10) з'явилась 7-ма версія (рис. 3.11), яка позиціонується вище конкурентів за швидкістю та точністю.

Layers	YOLOv3	YOLOv4	YOLOv5
Neural Network Type	FCNN	FCNN	FCNN
Backbone	**Darknet53**	**CSPDarknet53** (CSPNet in Darknet)	**CSPDarkent53** Focus structure
Neck	**FPN** (Feature Pyramid Network)	**SPP** (Spatial Pyramid Pooling) and **PANet** (Path Aggregation Network)	**PANet**
Head	**B x (5 + C)** output layer **B**: No. of bounding boxes **C**: Class score	Same as Yolo v3	Same as Yolo v3
Loss Function	Binary Cross Entropy	Binary Cross Entropy	Binary Cross Entropy and Logit Loss Function

Рисунок 3.10 – Передні версії YOLO

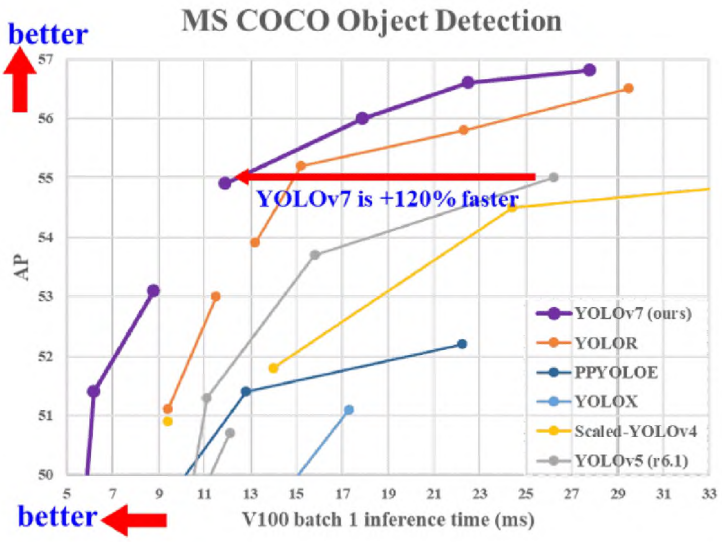

Рисунок 3.11 – YOLOv7

YOLOv7 покращує швидкість і точність, запроваджуючи кілька архітектурних реформ. Подібно до Scaled YOLOv4, магістралі YOLOv7 не використовують попередньо навчені магістралі ImageNet. Натомість моделі навчаються повністю з використанням набору даних COCO. Подібності можна очікувати, оскільки YOLOv7 написаний тими ж авторами, що і Scaled YOLOv4, який є розширенням YOLOv4. Архітектура походить від YOLOv4,

Scaled YOLOv4 і YOLO-R. Використовуючи ці моделі як основу, були проведені подальші експерименти для розробки нових і вдосконалених YOLOv7:

– E-ELAN (розширена ефективна мережа агрегації рівнів);

– масштабування моделі для моделей на основі конкатенації;

– планова перепараметризована згортка та ін.

На рис. 3.12 наведено результати експериментального порівняння YOLOv7 та інших базових моделей.

Model	#Param.	FLOPs	Size	AP^{val}	AP^{val}_{50}	AP^{val}_{75}	AP^{val}_{S}	AP^{val}_{M}	AP^{val}_{L}
YOLOv4 [3]	64.4M	142.8G	640	49.7%	68.2%	54.3%	32.9%	54.8%	63.7%
YOLOR-u5 (r6.1) [81]	46.5M	109.1G	640	50.2%	68.7%	54.6%	33.2%	55.5%	63.7%
YOLOv4-CSP [79]	52.9M	120.4G	640	50.3%	68.6%	54.9%	34.2%	55.6%	65.1%
YOLOR-CSP [81]	52.9M	120.4G	640	50.8%	69.5%	55.3%	33.7%	56.0%	65.4%
YOLOv7	36.9M	104.7G	640	**51.2%**	**69.7%**	**55.5%**	**35.2%**	**56.0%**	**66.7%**
improvement	-43%	-15%	-	+0.4	+0.2	+0.2	+1.5	=	+1.3
YOLOR-CSP-X [81]	96.9M	226.8G	640	52.7%	**71.3%**	57.4%	36.3%	57.5%	68.3%
YOLOv7-X	71.3M	189.9G	640	**52.9%**	71.1%	**57.5%**	**36.9%**	**57.7%**	**68.6%**
improvement	-36%	-19%	-	+0.2	-0.2	+0.1	+0.6	+0.2	+0.3
YOLOv4-tiny [79]	6.1	6.9	416	24.9%	42.1%	25.7%	8.7%	28.4%	39.2%
YOLOv7-tiny	6.2	5.8	416	**35.2%**	**52.8%**	**37.3%**	**15.7%**	**38.0%**	**53.4%**
improvement	+2%	-19%	-	+10.3	+10.7	+11.6	+7.0	+9.6	+14.2
YOLOv4-tiny-3l [79]	8.7	5.2	320	30.8%	47.3%	32.2%	**10.9%**	31.9%	51.5%
YOLOv7-tiny	6.2	3.5	320	**30.8%**	**47.3%**	**32.2%**	10.0%	**31.9%**	**52.2%**
improvement	-39%	-49%	-	=	=	=	-0.9	=	+0.7
YOLOR-E6 [81]	115.8M	683.2G	1280	55.7%	73.2%	60.7%	40.1%	**60.4%**	69.2%
YOLOv7-E6	97.2M	515.2G	1280	**55.9%**	**73.5%**	**61.1%**	**40.6%**	60.3%	**70.0%**
improvement	-19%	-33%	-	+0.2	+0.3	+0.4	+0.5	-0.1	+0.8
YOLOR-D6 [81]	151.7M	935.6G	1280	56.1%	73.9%	61.2%	**42.4%**	60.5%	69.9%
YOLOv7-D6	154.7M	806.8G	1280	56.3%	73.8%	61.4%	41.3%	60.6%	70.1%
YOLOv7-E6E	151.7M	843.2G	1280	**56.8%**	**74.4%**	**62.1%**	40.8%	**62.1%**	**70.6%**
improvement	=	-11%	-	+0.7	+0.5	+0.9	-1.6	+1.6	+0.7

Рисунок 3.12 – YOLOv7 проти інших базових моделей

Починаючи з моделі YOLOv7-Tiny, найменшої в сімействі з трохи більше 6 мільйонів параметрів. З параметром AP (35,2 %), він перевершує моделі YOLOv4-Tiny з подібними параметрами. Нормальна модель YOLOv7 з майже 37 мільйонами параметрів дає 51,2 % AP. Вона перевершує варіанти YOLOv4 і YOLOR, які легко мають більше параметрів. Більшими моделями сімейства YOLO7 є YOLOv7-X, YOLOv7-E6, YOLOv7-D6 і YOLOv7-E6E. Усі вони перевершують відповідні моделі YOLOR, маючи подібну або меншу кількість параметрів і дають AP рівним, відповідно: 52,9; 55,9; 56,3 та 56,8 %.

Тепер YOLOv7 перевершує не тільки моделі YOLOv4 і YOLOR. Порівнюючи AP моделей YOLOv5 і YOLOv7, які мають параметри в тому самому діапазоні, цілком зрозуміло, що YOLOv7 також перевершує всі моделі YOLOv5. На рис. 3.13 наведено результати порівняння YOLOv7 за FPS. YOLOv7 має найвищий FPS і mAP в діапазоні від 5 FPS до 160 FPS. Усі порівняння FPS проводилися на графічному процесорі Tesla V100.

Model	Parameters (million)	FPS	AP test (%)
YOLO7-Tiny	6.2	286	38.7
YOLOv7	36.9	161	51.4
YOLOv7-X	71.3	114	53.1
YOLOv7-W6	70.04	84	54.9
YOLOv7-E6	**97.2**	**56**	**56.0**
YOLOv7-D6	154.7	44	56.6
YOLOv7-E6E	151.7	36	56.8

Рисунок 3.13 – Порівняння FPS моделі YOLOv7

Варто зазначити, що жодна з моделей YOLOv7 не призначена для мобільних пристроїв/мобільних процесорів (як зазначено в документі YOLOv7). YOLOv7-Tiny, YOLOv7 і YOLOv7-W6 призначені для периферійного GPU, звичайного (споживчого) GPU та хмарного GPU відповідно. YOLOv7-E6 і YOLOv7-D6, і YOLOv7-E6E також призначені лише для хмарних графічних процесорів високого класу. Тим не менш, усі моделі YOLOv7 працюють зі швидкістю понад 30 FPS на графічному процесорі Tesla V100, що більше, ніж FPS у реальному часі. Наведені вище експерименти доводять, що моделі YOLOv7 перевершують існуючі детектори об'єктів. Як щодо швидкості, так і щодо точності.

Для реалізації запропонованого в роботі підходу можливе на базі хмарних або туманних обчислень [61-63]. Інший варіант полягає у використанні Edge AI. Наприклад, Intel, Google і Nvidia пропонують апаратні платформи (прискорювачі) в малих форм-факторах (рис. 3.14), наприклад, Nvidia Jetson Nano, Google Coral USB, Intel Movidius NCS2.

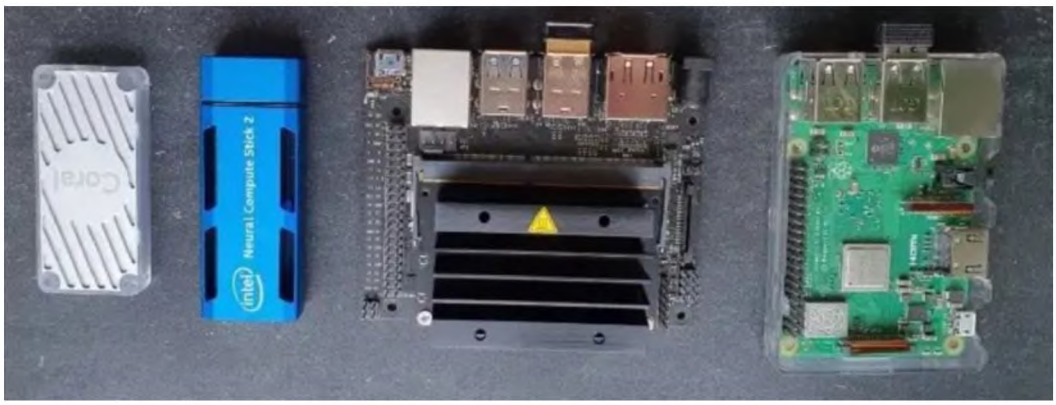

Рисунок 3.14 – Рішення Edge AI від Intel, Google і Nvidia

Вартість такого пристрою не перевищує 43000 грн (станом на листопад 2022 р.). Хоча всі три мають як слабкі, так і сильні сторони, все залежить від програми, бюджету та наявності наборів навичок.

Висновки до розділу 3

Щоб нейронна мережа вміла розпізнавати об'єкти на зображенні, проводиться анотація зображень. Automatic Image Annotation може виконуватись інструменти Open Source або комерційні утиліти.

З метою підтвердження висунутих в роботі теоретичних положень виконане моделювання глибокого навчання на прикладі нейронної мережі YOLOv4. Вона містить 61640962 параметри. Модель реалізовувалась в Google ColabPro+ на базі відеокарти Tesla V100-SXM2-16GB. Було досягнуто $mAP50 = 76{,}9$ %. Крім кількісної виконано якісну оцінки розглянутої моделі глибокого навчання YOLOv4 для завдання Object Detection. Отримані результати підтвердили теоретичні положення.

Перспективні напрями спільного використання Object Detection і БПЛА пов’язані з використанням YOLOv7, а також апаратними рішеннями Nvidia Jetson Nano, Google Coral USB, Intel Movidius NCS2.

ВИСНОВКИ

Концепція Smart Farm базується на інтеграції передових інформаційно-комунікаційних технологій, які спрямовані на використання автономних роботів, IoT та БПЛА.

На даний час, для БПЛА існують базові напрями використання, до яких слід віднести: виявлення та класифікація рослин; оцінка здоров'я рослин (моніторинг здоров'я посівів; виявлення хвороб і пошкоджень культур; сортування та аналіз форми плодів сільськогосподарської продукції); виявлення хвороб і пошкоджень культур; розумна боротьба зі шкідниками та травами; аналіз поля та оцінка врожайності.

Доступність візуальних даних, що отримані з БПЛА, і розвиток технологій глибокого навчання дозволяють розширити спектр потенційних можливостей дронів за рахунок інтелектуальних функцій Object Detection, наприклад, в інтересах автоматизованого моніторингу пожежної безпеки посівів.

Для впровадження Object Detection можна застосовувати кілька поширених архітектур нейронних мереж, серед яких слід виділити YOLO. Вона передбачає обмежуючі блоки та ймовірності класу для цих блоків. При здійснені один перегляд зображення. Недоліком такої архітектури вважаються просторові обмеження.

Розглянута в роботі модифікація MoAm-YOLOv4 у порівнянні з оригіналом не має значного зниження, але Precision, Recall і mAP зросли на 1,8, 1,6 і 51 %, відповідно, а розмір моделі зменшився на 80 %. Дана модифікація YOLOv4 має розмір моделі становить лише 45,58 МБ, mAP на рівні 93,45 %, а точність може сягати 93,28 %. Основні зміни стосуються заміни магістральної мережі Darknet-53 на мережу MobilenetV1.

Для того, щоб нейронна мережа вміла розпізнавати об'єкти на зображенні, проводиться анотація зображень. Automatic Image Annotation може виконуватись інструменти Open Source або комерційні утиліти.

З метою підтвердження висунутих в роботі теоретичних положень виконане моделювання глибокого навчання на прикладі нейронної мережі YOLOv4. Вона містить 61640962 параметри. Модель реалізовувалась в Google ColabPro+ на базі відеокарти Tesla V100-SXM2-16GB. Було досягнуто $mAP50 = 76{,}9$ %. Крім кількісної виконано якісну оцінки розглянутої моделі глибокого навчання YOLOv4 для завдання Object Detection. Отримані результати підтвердили теоретичні положення.

Перспективні напрями спільного використання Object Detection і БПЛА пов'язані з використанням YOLOv7, а також апаратних рішень edge AI.