

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КАФЕДРА ШТУЧНОГО ІНТЕЛЕКТУ**

КВАЛІФІКАЦІЙНА РОБОТА

на тему: «Розробка алгоритму розпізнавання типу автомобіля на зображеннях з відеокамери БПЛА»

на здобуття освітнього ступеня бакалавра
зі спеціальності 122 Комп'ютерні науки

освітньо-професійної програми Штучний інтелект

*Кваліфікаційна робота містить результати власних досліджень.
Використання ідей, результатів і текстів інших авторів мають посилання
на відповідне джерело*

_____ Богдан ЛЕВУСЬ

Виконав: здобувач вищої освіти
гр. ШІД-41

Богдан ЛЕВУСЬ

Керівник: Олександр _____

К.т.н. доцент

Рецензент:

*науковий ступінь,
вчене звання* _____

Київ 2024

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ**
Навчально-науковий інститут інформаційних технологій

Кафедра Штучного інтелекту

Ступінь вищої освіти Бакалавр

Спеціальність 122 Комп'ютерні науки

Освітньо-професійна програма Штучний інтелект

ЗАТВЕРДЖУЮ

Завідувач кафедру Штучного інтелекту

_____ Ольга ЗІНЧЕНКО

«_____» _____ 2024 р.

**ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

Левусь Богдан Дмитрович

1. Тема кваліфікаційної роботи: Розробка алгоритму розпізнавання типу автомобіля на зображеннях з відеокамери БПЛА.

керівник кваліфікаційної роботи Олександр ЗВЕНИГОРОДСЬКИЙ к.т.н., доцент,

затверджені наказом Державного університету інформаційно-комунікаційних технологій від «27» 02.2024р. № 36

2. Строк подання кваліфікаційної роботи «31» травня 2024р.

3. Вихідні дані до кваліфікаційної роботи: науково-технічна література, алгоритми розпізнавання об'єктів, вимоги до алгоритму розпізнавання

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)

Дослідження нейронних мереж для використання з БПЛА

Аналіз наявних засобів та технологій

Розробка алгоритму

5. Перелік графічного матеріалу: *презентація*

1. Вступ.

2. Безпілотні літальні апарати.

3. База даних.
4. Нейронна мережа
5. Висновки.

6. Дата видачі завдання «27» лютого 2024 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Збір та аналіз даних	27.02-12.03.24	Виконано
2	Вивчення безпілотних літальних апаратів	23.03-27.03.24	Виконано
3	Дослідження баз даних	27.03-1.04.24	Виконано
4	Дослідження штучних нейронних мереж	01.04-20.04.24	Виконано
5	Виконання практичної частини	22.04.-02.05.24	Виконано
6	Розробка демонстраційних матеріалів	04.05-8.05.24	Виконано
7	Передзахист	14.05-17.05.24	Виконано
8	Здача в декана	25.05-1.06.24	Виконано

Здобувач вищої освіти _____

Богдан ЛЕБУСЬ

Керівник
кваліфікаційної роботи _____

Олександр ЗВЕНИГОРОДСЬКИЙ

РЕФЕРАТ

Текстова частина кваліфікаційної роботи на здобуття освітнього ступеня бакалавра: 45 стор., 23 рис., 9 джерел.

Мета роботи – розробити алгоритм розпізнавання типу автомобіля на зображеннях з відеокамери БПЛА для підвищення ефективності їх використання.

Об'єкт дослідження – процес впровадження та оптимізації автоматизації обробки даних з зображень відеокамер БПЛА.

Предмет дослідження – алгоритм розпізнавання автомобільної техніки.

Короткий зміст роботи: розробка алгоритму розпізнавання автомобільного транспорту з метою підвищення ефективності використання БПЛА та автоматизації процесів органів управління. Використання штучного інтелекту з метою унеможливлення людських помилок.

Проведено аналіз найпопулярніших безпілотних літальних апаратів, використовуваних у оборонній сфері. Для програмної реалізації системи використовуватимуться мови програмування Python.

КЛЮЧОВІ СЛОВА: БЕЗПЛОТНИЙ ЛІТАЛЬНИЙ АПАРАТ, ШТУЧНИЙ-ІНТЕЛЕКТ, ПУНТОН.

ЗМІСТ

ВСТУП.....	9
РОЗДІЛ 1 БЕЗПІЛОТНІ ЛІТАЛЬНІ АПАРАТИ ТА НЕЙРОННІ МЕРЕЖІ.....	11
1.1 Безпілотні літальні апарати, класифікація, зразки	11
1.2 Українські БПЛА	15
1.3 Безпілотні літальні пристрої та штучний інтелект	23
1.4 Штучні нейронні мережі, види й задачі машинного навчання	23
Висновки до першого розділу.....	25
РОЗДІЛ 2 ОПИС БАЗИ ДАНИХ	26
2.1 загальна характеристика баз даних	26
2.2 База даних COCO Dataset.....	28
Висновки до другого розділу.....	31
РОЗДІЛ 3 НЕЙРОННА МЕРЕЖА РОЗПІЗНАВАННЯ АВТОМОБІЛЬНОГО ТРАНСПОРТУ.....	32
3.1 Мова програмування та середовище	32
3.2 Алгоритми обробки зображень	35
3.3 Практична робота	36
ВИСНОВКИ.....	43
ВИСНОВКИ.....	43
ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ.....	45

ВСТУП

Актуальність теми. У сучасному світі безпілотні літальні апарати стають все більш масовими і розповсюдженими. Практично в усіх сферах можна зустріти БПЛА. У пожежогасінні допомагають великі октокоптери, також їх можна побачити в господарстві, де вони ефективно справляються з обприскуванням сільськогосподарських культур. Для боротьби з шкідниками використовують “мавіки”, невеликі дрони прокладають електромережі, а їх більші брати вже навіть навчилися підзаряджатися від кабелів електроживлення. Важко уявити рекламу без ефектного прольоту дорогої кінокамери прилаштованої на безпілотник, автомобільні перегони без надшвидких FPV-дронів які переганяють боліди Формули один, відео бойових дій не з висоти польоту “мавіка”. Залученість БПЛА в життя людини зараз важко переоцінити.

Мета роботи. Метою цієї дипломної роботи є розробка алгоритму розпізнавання типу автомобіля на зображеннях з відеокамери БПЛА. Алгоритм буде призначений для покращення ефективності безпілотних літальних апаратів та автоматизацію роботи з даними від них.

Аналіз останніх досліджень. У даному контексті, аналіз безпілотних літальних апаратів був потрібен для розуміння розвитку спостережних засобів на них. Наразі розробки БПЛА з використанням штучного інтелекту засекречені і є військовими програмами що означає зацікавленість військового сектору й подальший вихід подібних алгоритмів на цивільний ринок. Враховуючи ці фактори, вивчення алгоритмів штучного інтелекту є дуже важливим остільки це є технології майбутнього.

Об’єкт дослідження. Дослідження включає в себе аналіз алгоритмів машинного навчання, спрямований на створення алгоритму розпізнавання автомобільної техніки з відеокамер БПЛА

Предметом дослідження алгоритми розпізнавання зображень різної автомобільної техніки.

Дослідження та розробка цього алгоритму є захоплюючою та важливою темою. Ця робота дозволить не лише поглибити свої знання у сфері програмування та інформаційних технологій, але й ввести певний вклад у покращення обороноздатності та економічної ситуації у країні. Результатом цього дослідження стане створення ефективного алгоритму, який дозволить використовувати БПЛА більш ефективно. Це дослідження надихне інших фахівців на подальші розвідки у цій області та допоможе зробити тему БПЛА більш важливою у суспільстві.

1 БЕЗПЛОТНІ ЛІТАЛЬНІ АПАРАТИ ТА НЕЙРОННІ МЕРЕЖІ

1.1 Безпілотні літальні апарати, класифікація, зразки

Безпілотний літальний апарат (БпЛА) — повітряне судно, призначене для виконання польоту без пілота на борту, керування польотом здійснюються закладеною програмою або за допомогою спеціальної станції керування, що знаходиться поза повітряним судном (рис 1.1).[1]



Рис. 1.1 Різні безпілотні літальні апарати.

Безпілотні літальні комплекси - це комплекси, що складаються з одного або декількох безпілотних літальних апаратів, спеціального обладнання для перенесення, запуску, контролю польотом, передавання та опрацювання інформації (рис1.2) [1].



Рис.1.2

Безпілотний літальний комплекс.

Безпілотні літальні апарати можна класифікувати за різними критеріями, основні це є їх характеристики, розмір та призначення. За призначенням можна умовно поділити на дві групи, цивільні та військові. Зараз досить важко провести чітку грань між цими групами. В багатьох країнах таких як Німеччина, КНР та інші необхідно мати дозвіл на польоти на будь яких дронах, а також мати спеціальну ліцензію пілота. В основному цивільні безпілотні літальні апарати виконують для комерційних цілей таких як відеозйомка, доставка, вирощування сільськогосподарської продукції, а також пошук та порятунок людей, пожежогасіння, наукові дослідження. Військові БПЛА використовуються для таких цілей як розвідка, цілевказання, удари по противнику та для інших завдань.

Також можна поділити безпілотні літальні апарати за розмірами:

1. Мікро-БПЛА: Дуже маленькі апарати, зазвичай вагою кілька кілограмів або менше. Використовуються для маломасштабних місій або в обмежених просторах. Найкращий приклад PD-100 Black Hornet Nano.

Використовується силами спеціальних операцій багатьох країн світу. Неодноразово використовувався Силами оборони України, переданий від партнерів.

1. Середні БПЛА (MA): Більші за розміром, здатні до довшого польоту та більшої вантажопідйомності.
2. Великі БПЛА (LA): Великі апарати, які можуть мати значну вантажопідйомність та радіус дії.

За способом польоту:

1. Фіксований крило: Апарати з фіксованим крилом, схожі на літаки. Вони використовуються для довгих польотів та високої швидкості.
2. Мультироторні: БПЛА з кількома моторами, які забезпечують їм вертикальний зліт і посадку. Зазвичай використовуються для коротких місій або при потребі точної стабільності у повітрі.

Класифікація військових БпЛА НАТО

Класифікуються безпілотні літальні апарати, відповідно до стандартів НАТО, так само, як і літаки із пілотом на борту, керуючись значенням повної злітної маси розділено на 3 класи(рис 1.3):

1. I — повна злітна маса до 150 кг.
2. II — повна злітна маса до 600 кг.
3. III — повна злітна маса більше 600 кг.

Клас I поділяється на категорії:

1. мікро — до 2 кг
2. міні — до 15 кг
3. малі — від 15 кг

UAS CLASSIFICATION						
Class	Category	Normal Employment	Normal Operating Altitude	Normal Mission Radius	Primary Supported Commander	Example Platform
Class III (> 600 kg)	Strike/ Combat *	Strategic/National	Up to 65,000 ft	Unlimited (BLOS)	Theatre	Reaper
	HALE	Strategic/National	Up to 65,000 ft	Unlimited (BLOS)	Theatre	Global Hawk
	MALE	Operational/Theatre	Up to 45,000 ft MSL	Unlimited (BLOS)	JTF	Heron
Class II (150 kg - 600 kg)	Tactical	Tactical Formation	Up to 18,000 ft AGL	200 km (LOS)	Brigade	Hermes 450
Class I (< 150 kg)	Small (>15 kg)	Tactical Unit	Up to 5,000 ft AGL	50 km (LOS)	Battalion, Regiment	Scan Eagle
	Mini (<15 kg)	Tactical Sub -unit (manual or hand launch)	Up to 3,000 ft AGL	Up to 25 km (LOS)	Company, Platoon, Squad	Skylark
	Micro ** (<66 J)	Tactical Sub -unit (manual or hand launch)	Up to 200 ft AGL	Up to 5 km (LOS)	Platoon, Squad	Black Widow

Рис. 1.3 Класифікація країн НАТО.

Використання безпілотних літальних апаратів (БПЛА) має різноманітні напрямки, що охоплюють різні сфери життя та діяльності. Ці апарати відкривають широкі можливості як у комерційній, так і у цивільній сферах, а також у військовому застосуванні. БПЛА використовуються для моніторингу та розвідки великих територій, які мають різні природні та географічні особливості. Вони стали незамінними в агротехнологіях, де вони використовуються для моніторингу стану сільськогосподарських культур, виявлення хвороб та шкідників, а також для оптимізації використання ресурсів. Крім того, БПЛА використовуються в геодезії та картографії для створення високоточних карт територій та тривимірних моделей ландшафту. У сфері доставки, вони допомагають зменшити час доставки та витрати на логістику, а у сфері пошуку та рятування - виявляти людей в умовах стихійних лих або при підтопленнях. БПЛА також широко використовуються для фото- та відеозйомки, як у комерційних, так і у художніх проектах. У військовому використанні, вони застосовуються для розвідки, наведення вогню, патрулювання кордонів та атак. Також вони знайшли своє застосування в наукових

дослідженнях, де вони використовуються для вивчення кліматичних змін, моніторингу дикої природи.

1.2 Українські БПЛА

Українські БПЛА представлені в майже усіх класах за класифікацією НАТО і є вдалими та ефективними представниками безпілотних літальних пристроїв світового рівня.

БПЛА «ФУРІЯ»



Рис. 1.4. БПЛА «ФУРІЯ»

БПАК «Фурія» розробило ТОВ «НВП "Атлон Авіа"» 2014 року. Уперше було представлено в січні 2015 року. На озброєння Національної гвардії України взято 2015 року. У 2015 році Збройні сили України закупили п'ять комплексів БПАК «Фурія». Офіційно взято на озброєння у квітні 2020 року за результатами успішних державних випробувань. Усього станом на лютий 2022 року в ЗСУ було

близько 300 апаратів цього типу, здебільшого в артилерійських підрозділах бригад.

Тактико-технічні характеристики [8]

Загальні:

Тип: літаюче крило

Матеріали: склотканина, карбон, кевлар

Максимальна злітна вага: 5,5 кг

Габаритні розміри:

довжина: 900 мм

розмах крил: 2050 мм

Силова установка:

Двигун: електричний

Живлення: Li-Po / Li-ion батареї до 42000 mAh

Експлуатаційні показники:

Оперативний радіус: до 50 км

Час польоту: близько 3 год (нормальні метеоумови)

Крейсерська швидкість: 65 км/год

Максимальна швидкість: 130 км/год

Дальність польоту: до 200 км

Режим польоту: ручний, напівавтоматичний, автономний

Практична стеля: 2500 м

Максимальна допустима швидкість вітру: до 15 м/с

Спосіб зльоту: еластична або механічна катапульта

Спосіб посадки: парашутний (штатно) або по-літаковому

Корисне навантаження БПЛА

Денне:

Ефективна робоча висота до 1200 м в залежності від типу місії*

Оптичний модуль на базі Sony FSB-H11

- 2-х осьова механічна стабілізація, цифрова стабілізація
- 10-х оптичний та 12-х цифровий зум
- горизонтальний кут візування: від 50° до 5,4°
- роздільна здатність: 0,1 м
- режим підвищеної контрастності
- запис Full HD (1080p) на борту БПЛА

Нічне:

Ефективна робоча висота до 450-550 м в залежності від типу місії*

Оптичний модуль на базі Flir Tau 640x480 30Hz або аналога

- 2-х осьова механічна стабілізація, цифрова стабілізація
- 2,4,8-х цифровий зум
- горизонтальний кут візування: 18° x 14°
- запис відео на борту БПЛА з роздільною здатністю 640x480



Рис.1.5. Вид екрану при польоті БПЛА «ФУРІЯ»

БПЛА ЛЕЛЕКА-100



Рис.

1.6. БПЛА Лелека-100

БПАК «Лелека-100» створила українська компанія DeViRo на базі китайського апарату X-UAV Talon. У лютому 2017 року комплекс пройшов полігонні випробування, станом на 2018 рік перебував на дослідній експлуатації в ЗСУ. Узято на озброєння ЗСУ у травні 2021 року. У квітні 2020 року Головний центр підготовки особового складу Держприкордонслужби України ім. генерал-майора Ігоря Момота безплатно отримав БПАК «Лелека-100». Станом на весну 2020 року до ЗСУ було поставлено понад 300 БПЛА цього типу. Наприкінці квітня 2022 року Фонд допомоги армії «Повернись живим» підписав угоду на виробництво 25 комплексів «Лелека-100». Нині апарат активно використовують на фронті.

Тактико-технічні характеристики [9]

Загальні:

Тип: літаюче крило

Матеріали: склотканина, карбон, кевлар

Максимальна злітна вага: 5 кг

Габаритні розміри:

довжина: 1135мм

розмах крил: 1980 мм

Силова установка:

Двигун: електричний

Живлення: Li-Po / Li-ion батареї

Експлуатаційні показники:

Оперативний радіус: до 50 км

Час польоту: близько 1.5-4 год

Крейсерська швидкість: 60-70 км/год

Максимальна швидкість: 120 км/год

Дальність польоту: до 100 км

Режим польоту: ручний, напівавтоматичний, автономний

Практична стеля: 1500 м

Максимальна допустима швидкість вітру: до 20 м/с

Спосіб зльоту: еластична або механічна катапульта

Спосіб посадки: парашутний (штатно) або по-літаковому

2. Корисне навантаження БПЛА

Цільове спорядження

модульне змінне:

1. модуль 'F16x9' — дистанційно керований, гіроскопічно стабілізований у двох площинах, фіксоване 4-х кратне оптичне збільшення.
2. модуль 'Z 10x32' — дистанційно керований, гіроскопічно стабілізований у двох площинах, змінне 16-ти кратне оптичне збільшення.
3. модуль тепловізійний 'T25x25' — дистанційно керований за віссю тангажу, формат 640×480, 50Гц.

Канал відеоданих

дистанційне керований:

1. аналоговий канал (базова комплектація) 720x576, стабільна передача на відстань до 40 км;
2. цифровий канал (дод. опція) 1080p full HD(рис 1.3), стабільна передача на відстань до 15 км.



Рис.

1.7. Вид екрану при польоті БПЛА Лелека-100

БПЛА SHARK



Рис. 1.8. БПЛА SHARK

БПЛА SHARK був розроблений українською компанією Ukrspecsystems у 2022 році. «SHARK» об'єднує у собі усі найкращі технології що вже були опрацьовані на попередніх виробках компанії. БПЛА запускається з катапульти та має оперативну дальність до 80 км, стеля до 3000 метрів. Максимальна швидкість — 130 км/год, крейсерська швидкість — 75 км/год, швидкість звалювання - 60 км/год. Посадка здійснюється за допомогою парашуту. Для спостереженнями за об'єктами використовується 30 кратний зум-модуль та тепловізійні канали спостереження. Решта технічних характеристик є засекреченими.



Рис. 1.9. Вид екрану при польоті БПЛА SHARK

1.3 Безпілотні літальні пристрої та штучний інтелект

Використання штучного інтелекту може суттєво підвищити ефективність використання безпілотних літальних пристроїв за рахунок таких чинників:

1. Стійкість до втрати керування. Втрата керування є серйозною проблемою в застосуванні безпілотних літальних апаратів, оскільки втрата багатовартісного борту спричиняє високі економічні і не тільки втрати. Коли БПЛА почнуть самі визначати вірність маршруту, безпечність польоту над певними територіями та на певних висотах, автоматичне приземлення у визначених районах, то питання втрати борту не буде настільки страшною проблемою
2. Визначення цілі. Визначення цілі з використанням штучних нейронних мереж допоможе пілотам витратити менше часу на пошук та класифікацію цілі та збільшить ефективність та швидкодію усієї системи.
3. Автоматизовані польоти. Повністю автоматизовані польоти без участі людини є дуже необхідними на даний час для збереження життів екіпажів безпілотних систем та прискорення роботи по різних напрямках використання дронів.

1.4 Штучні нейронні мережі

Для автоматизації аналізу різноманітних зображень широко використовуються штучні нейронні мережі (ШНМ). Вони являють собою послідовність алгоритмів, що частково імітують будову та функціонування нервової системи живих організмів. Елементи ШНМ названі на честь біологічних структур – нейронів і дендритів. Нейрон – це змінна в оперативній пам'яті, яка обробляє інформацію залежно від архітектури ШНМ, а дендрит – число, що зберігається в постійній пам'яті.

Нейрони в мережі зазвичай організовані у вигляді шарів або рівнів. Перший рівень, відомий як вхідний, отримує дані з зовнішнього середовища і передає їх на

наступний рівень після обробки. Другий рівень, або прихований, обробляє отримані сигнали і передає їх далі. Вихідний рівень отримує оброблені дані від прихованих шарів і генерує результат. Така структура дозволяє ШНМ навчатися та виявляти залежності в даних.

Проектування архітектури нейронної мережі, що включає визначення кількості шарів і нейронів у кожному, є необхідним кроком перед початком навчання. Існують різні методи навчання нейронних мереж, серед яких найбільш поширені навчання з учителем (supervised learning) та навчання без учителя (unsupervised learning).

Навчання з учителем передбачає, що машина працює з заздалегідь міченими даними, що дозволяє швидше та ефективніше засвоювати інформацію. Навчання без учителя полягає в роботі з немаркованими даними, де машина самостійно виявляє закономірності та зв'язки, класифікує і описує структуру даних.

Після успішного навчання нейронні мережі можуть виконувати завдання регресії, класифікації, кластеризації, прогнозування, зменшення розмірності, виявлення аномалій та пошуку правил. Класифікація є особливо важливою для аналізу зображень, оскільки вона дозволяє моделі розподіляти дані по певним категоріям на основі тренувальних даних з мітками, попередньо заданими вчителем. Після навчання нейронна мережа може класифікувати нові, непозначені дані.

1.5 Висновки до розділу 1

1. Світова тенденція популяризації безпілотних літальних апаратів не оминає Україну і ми можемо побачити широке і масове виробництво прогресивних та технологічних засобів ведення спостереження з повітря. Використання безпілотних літальних пристроїв розповсюджується на усі сфери діяльності людини.

2. Штучні нейронні мережі можуть бути використані для підвищення ефективності безпілотних літальних апаратів та є досить прогресивною та амбіційною складовою майбутнього розвитку БПЛА.

3. Є різні методи машинного навчання які можуть бути використані для навчання нейронних мереж які можуть бути застосовані для використання зображень з безпілотних літальних пристроїв

2 ОПИС БАЗИ ДАНИХ ТА АРХІТЕКТУРИ НЕЙРОННОЇ МЕРЕЖІ

2.1 Загальна характеристика баз даних

Бази даних, які використовуються при навчанні нейронних мереж, мають специфічні характеристики, що роблять їх особливо корисними для цього завдання:

1. Об'єм і Розмір Даних
2. Великі обсяги даних: Для ефективного навчання нейронних мереж потрібні великі набори даних, оскільки нейронні мережі мають багато параметрів, які потрібно налаштувати.
3. Висока роздільна здатність: Для зображень, аудіо та відео дані часто мають високу роздільну здатність, щоб зберегти максимум інформації.
4. Різні типи даних: Використовуються різноманітні типи даних, такі як зображення (для комп'ютерного зору), текст (для обробки природної мови), аудіо (для розпізнавання мови) і т.д.
5. Багатокласові та багатоміткові набори: Часто дані містять інформацію про кілька класів або міток, що дозволяє навчати моделі для розв'язання складніших завдань.
6. Ретельно анотовані дані: Набори даних часто містять анотації або мітки, що дозволяє нейронним мережам вчитися на прикладах з правильними відповідями.
7. Чіткі інструкції для анотації: Існують стандартизовані інструкції для створення анотацій, що підвищує якість даних.
8. Структуровані та неструктуровані дані: Дані можуть бути як структуровані (таблиці, бази даних), так і неструктуровані (тексти, зображення).

9. Популярні формати: Використовуються стандартні формати даних, такі як CSV, JSON, XML, HDF5, TFRecord (для TensorFlow) і інші.
10. Публічно доступні набори даних: Багато наборів даних є у вільному доступі для наукового використання, що сприяє розвитку досліджень у галузі штучного інтелекту.
11. Спільноти і конкурси: Існують платформи, такі як Kaggle, де дослідники можуть ділитися даними і брати участь у змаганнях.
12. Очистка даних: Процес видалення шуму, виправлення помилок і заповнення пропущених значень.
13. Нормалізація і стандартизація: Перетворення даних в єдиний формат, що полегшує навчання моделей.
14. Балансування класів: Важливо, щоб дані були збалансованими між різними класами для уникнення упередженості моделі.
15. Розподіл на тренувальні, валідаційні та тестові набори: Для оцінки якості навчання і запобігання перенавчанню.

Приклади популярних баз даних для навчання нейронних мереж:

1. MNIST: База даних зображень рукописних цифр.
2. CIFAR-10/100: Набори зображень для задач класифікації.
3. ImageNet: Великий набір зображень з мітками для різних об'єктів.
4. COCO (Common Objects in Context): Зображення з анотаціями для задач сегментації і детекції.
5. OpenAI Gym: Набір середовищ для навчання агентів в підкріплювальному навчанні.
6. IMDB: Набір текстів для задач аналізу тональності.

7. LibriSpeech: Аудіо дані для розпізнавання мовлення.

Ці характеристики дозволяють створювати високоякісні нейронні мережі, здатні розв'язувати складні завдання з використанням машинного навчання та штучного інтелекту.

2.2 База даних COCO Dataset

Під час виконання практичної частини дипломної роботи було здійснено пошук моделей що вже були навчені на основі баз даних та підтримують можливість навчання. Автомобільна техніка є досить популярною серед розробників програмного забезпечення на основі штучного інтелекту що забезпечує якісний і швидкий результат. Існує безліч баз даних та ще більше моделей створених на їх основі.

Для використання у моїй роботі був вибраний COCO Dataset як один з найбільших та найякісніших наборів даних для об'єктного виявлення та



сегментації зображень. Він був створений для стимулювання розвитку

комп'ютерного зору, зокрема сегментації об'єктів у сценах з реальними умовами. COCO Dataset має величезну кількість зображень на яких представлені різні типи об'єктів, включаючи тварин, транспортні засоби, меблі, фрукти, овочі, інструменти та багато іншого. Кожне зображення має анотації, які вказують на місцезнаходження та класи об'єктів на зображенні. Це дозволяє наїр бути максимально ефективним у навчанні нейронних мереж та ефективній сегментації. [4][5]

Рис. 2.1 Приклад COCO Dataset



Рис. 2.2 Приклад COCO Dataset

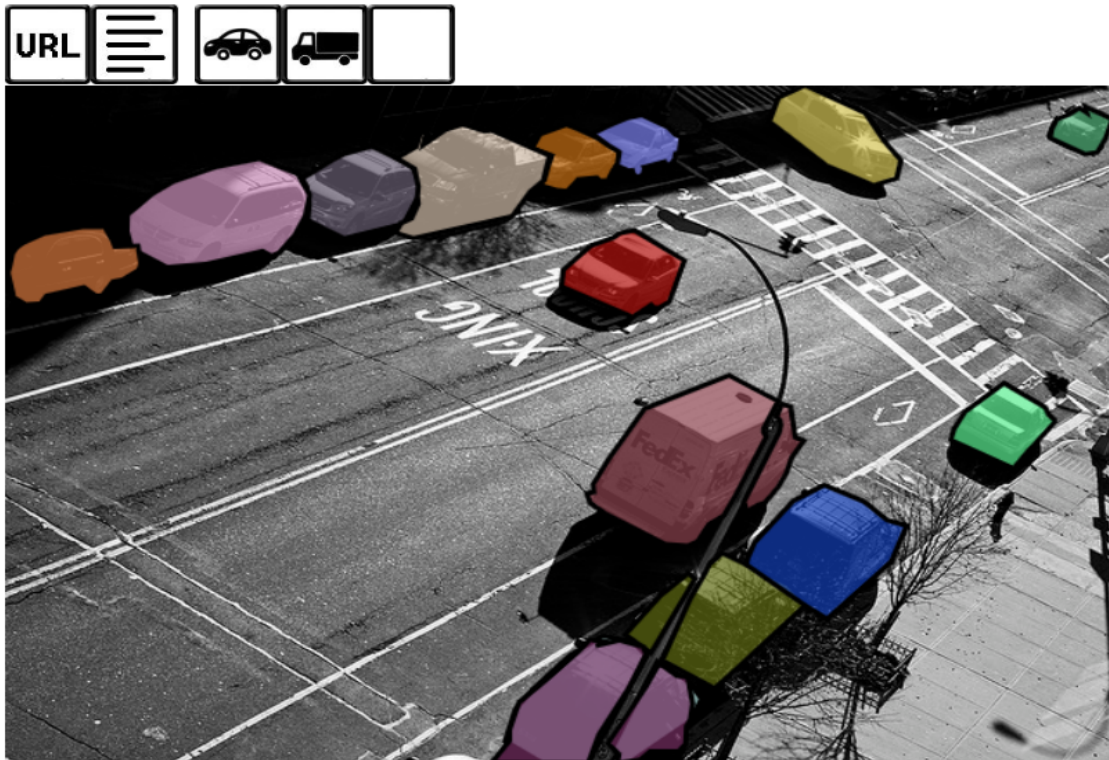


Рис. 2.3 Приклад COCO Dataset

На усіх зображеннях ми бачимо різним кольором виділені різні об'єкти які по різному класифіковані: транспорт різних типів, людей, дорожні знаки тощо. Це все дозволяє використовувати COCO Dataset для роботи зі штучними нейронними мережами для найефективнішого навчання на реальних прикладах.

Висновки до розділу 2

Аналіз сучасних баз даних показав що COCO Dataset є найкращою та найзручнішою для виконання поставленого завдання. Кількість предметів, варіативність, зручність та легкість використання є важливими перевагами цієї бази даних.

3 НЕЙРОННА МЕРЕЖА РОЗПІЗНАВАННЯ АВТОМОБІЛЬНОГО ТРАНСПОРТУ

3.1 Мова програмування та середовище

Для роботи з штучними нейронними мережами потрібно правильно вибрати оптимальну мову програмування та середовище що забезпечить найвищу продуктивність та швидкодію усіх компонентів майбутнього проєкту.

Мовою програмування мною був обраний Python. Python є одним з найкращих можливих варіантів для застосування з нейронними мережами по таким причинам:

1. Python має велику кількість бібліотек та фреймворків, які значно полегшують розробку та тренування нейронних мереж. Найпопулярніші з них включають TensorFlow, Keras, PyTorch, і Theano. Ці бібліотеки забезпечують високорівневі API для створення моделей, що дозволяє зосередитись на архітектурі мережі та експериментах, а не на низькорівневій реалізації алгоритмів.
2. Легкість читання та синтаксис. Python має простий синтаксис що дозволяє легко читати і писати код, підходить як для новачків, так і для досвідчених дослідників і розробників штучних нейронних мереж. Він не відштовхує від себе своєю складністю як інші мови програмування та часто вивчається навіть у школах.
3. Python має одну з найбільших спільнот і найактивніших спільнот у світі. Існує безліч посібників, ресурсів, форумів та матеріалів по використанню цієї мови програмування та завжди точно є у кого запитати про той, чи інший аспект застосування.
4. Python є досить гнучким та варіативним. Він легко компонується з іншими мовами що можуть використовуватися у проєктах, таких як C++ та в цілому усі C-

```
import tensorflow as tf
from tensorflow.keras import layers, models
from tensorflow.keras.datasets import mnist
from tensorflow.keras.utils import to_categorical

# Завантаження та підготовка даних
(train_images, train_labels), (test_images, test_labels) = mnist.load_data()

train_images = train_images.reshape((60000, 28, 28, 1)).astype('float32') / 255
test_images = test_images.reshape((10000, 28, 28, 1)).astype('float32') / 255

train_labels = to_categorical(train_labels)
test_labels = to_categorical(test_labels)

# Побудова моделі
model = models.Sequential()
model.add(layers.Conv2D(32, (3, 3), activation='relu', input_shape=(28, 28, 1)))
model.add(layers.MaxPooling2D((2, 2)))
model.add(layers.Conv2D(64, (3, 3), activation='relu'))
model.add(layers.MaxPooling2D((2, 2)))
model.add(layers.Conv2D(64, (3, 3), activation='relu'))
model.add(layers.Flatten())
model.add(layers.Dense(64, activation='relu'))
model.add(layers.Dense(10, activation='softmax'))

# Компіляція моделі
model.compile(optimizer='adam',
              loss='categorical_crossentropy',
              metrics=['accuracy'])

# Тренування моделі
model.fit(train_images, train_labels, epochs=5, batch_size=64, validation_split=0.2)

# Оцінка моделі на тестовому наборі
test_loss, test_acc = model.evaluate(test_images, test_labels)
print(f"Тестова точність: {test_acc:.4f}")
```

Рис. 3.1 Приклад коду Python

Робота виконувалася в середовищі Google Colab. Це безкоштовне хмарне середовище, яке дозволяє писати та виконувати код на Python прямо в браузері. Його основне призначення – забезпечити легкий доступ до потужних обчислювальних ресурсів особливо для завдань машинного навчання та аналізу

даних. Основними перевагами Google Colab є базування на Jupyter Notebook який має такі переваги: 1. Попередній аналіз: Jupyter надає можливість переглядати результати виконання тих чи інших рядків коду, незалежно від інших його частин. У Jupyter кожна комірка коду може бути перевірена в будь-який момент, щоб оцінити результати, які отримуються у процесі написання програми.

2. Інтерактивна взаємодія з кодом: Jupyter Notebook використовує пакети, що дозволяють редагувати код і повторно запускати його безпосередньо у Jupyter, забезпечуючи живу взаємодію з кодом. Таким чином, користувачі можуть контролювати та повторно змінювати вхідні дані для коду, а також отримувати зворотний зв'язок прямо у веб-браузері.

3. Зручність у веденні документації: Jupyter містить окремі блоки, призначені для введення тексту, що надає зручні можливості для пояснення коду крок за кроком у повному та зручному обсязі.

Значною перевагою Google Colab є підтримка різних бібліотек таких як NumPy, Pandas, Matplotlib, TensorFlow, Keras, PyTorch і багато інших. Google Colab інтегрований з Google Drive що дуже зручно для роботи над проектами з різних девайсів, передача результатів і даних стає зручнішою і захищено.

Однією з найбільших переваг є потужність. Хмарні обчислювальні можливості Google Colab дозволяє виконувати складні обчислення, які були б неможливими на звичайних ПК. Автоматичне зберігання у хмарі напрацьованого коду і інформації є дуже актуальним, оскільки в наш з постійними відключеннями світла практично неможливо бути впевненим що буде збережено весь матеріал.

```

# clone darknet repo
git clone https://github.com/AlexeyAB/darknet

Cloning into 'darknet'...
remote: Enumerating objects: 15851, done.
remote: Counting objects: 100% (18/18), done.
remote: Compressing objects: 100% (14/14), done.
remote: Total 15851 (delta 5), reused 13 (delta 4), pack-reused 15833
Receiving objects: 100% (15851/15851), 14.42 MiB | 6.56 MiB/s, done.
Resolving deltas: 100% (10671/10671), done.

# change makefile to have GPU and OPENCV enabled
%cd darknet
lsed -i 's/OPENCV=0/OPENCV=1/' Makefile
lsed -i 's/GPU=0/GPU=1/' Makefile
lsed -i 's/CUDNN=0/CUDNN=1/' Makefile
lsed -i 's/CUDNN_HALF=0/CUDNN_HALF=1/' Makefile

/content/darknet

# verify CUDA
!usr/local/cuda/bin/nvcc --version

nvcc: NVIDIA (R) cuda compiler driver
Copyright (c) 2005-2023 NVIDIA Corporation
Built on Tue_Aug_15_22:02:13_PDT_2023
Cuda compilation tools, release 12.2, V12.2.140
Build cuda_12.2.r12.2/compiler.33191640_0

# make darknet (builds darknet so that you can then use the darknet executable file to run or train object detectors)

```

Рис. 3.2 Приклад коду Python

3.2 Алгоритми обробки зображення

Існує декілька найпопулярніших алгоритмів обробки зображення:

SSD (Single Shot MultiBox Detector): Це популярний алгоритм для об'єктного виявлення, який працює швидко та точно. SSD використовує ряд згорткових та повнозв'язних шарів для прогнозування класів об'єктів та їх bounding boxes.

Faster R-CNN (Region-based Convolutional Neural Network): Цей алгоритм є більш складним порівняно з YOLO і SSD, але він також здатний досягати високої точності виявлення. Faster R-CNN використовує дві мережі - одну для виявлення пропозицій областей (region proposals), а іншу для класифікації та поправки bounding boxes.

YOLO - YOLO (You Only Look Once) - це один із найпопулярніших і потужних алгоритмів для об'єктного виявлення в зображеннях та відео. YOLO відомий своєю швидкістю та точністю, що робить його відмінним вибором для реального застосування в різних задачах комп'ютерного зору.

3.3 Практична частина

Для виконання поставленого мною завдання та отримання бажаних результатів мною був обраний алгоритм обробки зображень YOLOv4. Це остання версія алгоритму YOLO, яка була випущена. Ключові риси YOLOv4:

1. **Швидкість:** Однією з основних переваг YOLO є швидкість обробки зображень. YOLOv4, як і його попередники, пропонує високу швидкість виявлення об'єктів, що робить його ідеальним вибором для застосування в реальному часі.
2. **Точність:** YOLOv4 поєднує в собі техніки, спрямовані на підвищення точності виявлення об'єктів. Він використовує як нові, так і вдосконалені алгоритми, такі як спеціалізовані шари, оптимізовані функції активації та інші покращення, що дозволяють досягти вражаючих результатів.
3. **Складність моделі:** YOLOv4 має досить складну архітектуру, що дозволяє йому виявляти і класифікувати об'єкти з високою точністю та робити це досить швидко. Це досягається завдяки використанню різних технік машинного навчання, таких як згорткові нейронні мережі (CNN), оптимізація ваг та інші методи.
4. **Застосування:** YOLOv4 може бути використаний у багатьох сферах, включаючи моніторинг безпеки, автоматизоване водіння, розпізнавання облич, медичне зображення та багато іншого. Він відомий своєю універсальністю та можливістю ефективно працювати в різноманітних умовах.

Узагальнюючи усе вище перелічене, YOLOv4 є потужним інструментом для об'єктного виявлення зображень та відео, який поєднує в собі швидкість та точність що нам так необхідно. Цей алгоритм відкриває можливості до виконання задачі з найвищою ефективністю та автоматизацією. Для початку був клонований репозиторій darknet, за допомогою якого і працює алгоритм YOLO.

```

▶ # clone darknet repo
!git clone https://github.com/AlexeyAB/darknet

```

Підключаємо можливість роботи через GPU та підключаємо OPENCV

```

[ ] # change makefile to have GPU and OPENCV enabled
!cd darknet
!sed -i 's/OPENCV=0/OPENCV=1/' Makefile
!sed -i 's/GPU=0/GPU=1/' Makefile
!sed -i 's/CUDNN=0/CUDNN=1/' Makefile
!sed -i 's/CUDNN_HALF=0/CUDNN_HALF=1/' Makefile

```

Прописуємо подальший код

```

▶ # define helper functions
def imShow(path):
    import cv2
    import matplotlib.pyplot as plt
    %matplotlib inline

    image = cv2.imread(path)
    height, width = image.shape[:2]
    resized_image = cv2.resize(image, (3*width, 3*height), interpolation = cv2.INTER_CUBIC)

    fig = plt.gcf()
    fig.set_size_inches(18, 10)
    plt.axis("off")
    plt.imshow(cv2.cvtColor(resized_image, cv2.COLOR_BGR2RGB))
    plt.show()

# use this to upload files
def upload():
    from google.colab import files
    uploaded = files.upload()
    for name, data in uploaded.items():
        with open(name, 'wb') as f:
            f.write(data)
            print ('saved file', name)

# use this to download a file
def download(path):
    from google.colab import files
    files.download(path)

```

Після чого отримуємо віконечко у яке можемо завантажити зображення об'єкт на якому хочемо розпізнати

```
# try out the upload helper fu
%cd ..
upload()
%cd darknet
```

... /

Файл не вибрано

Після завантаження файлу з автомобілями на магістралі ми отримуємо таке зображення:



Рис. 3.3 До обробки

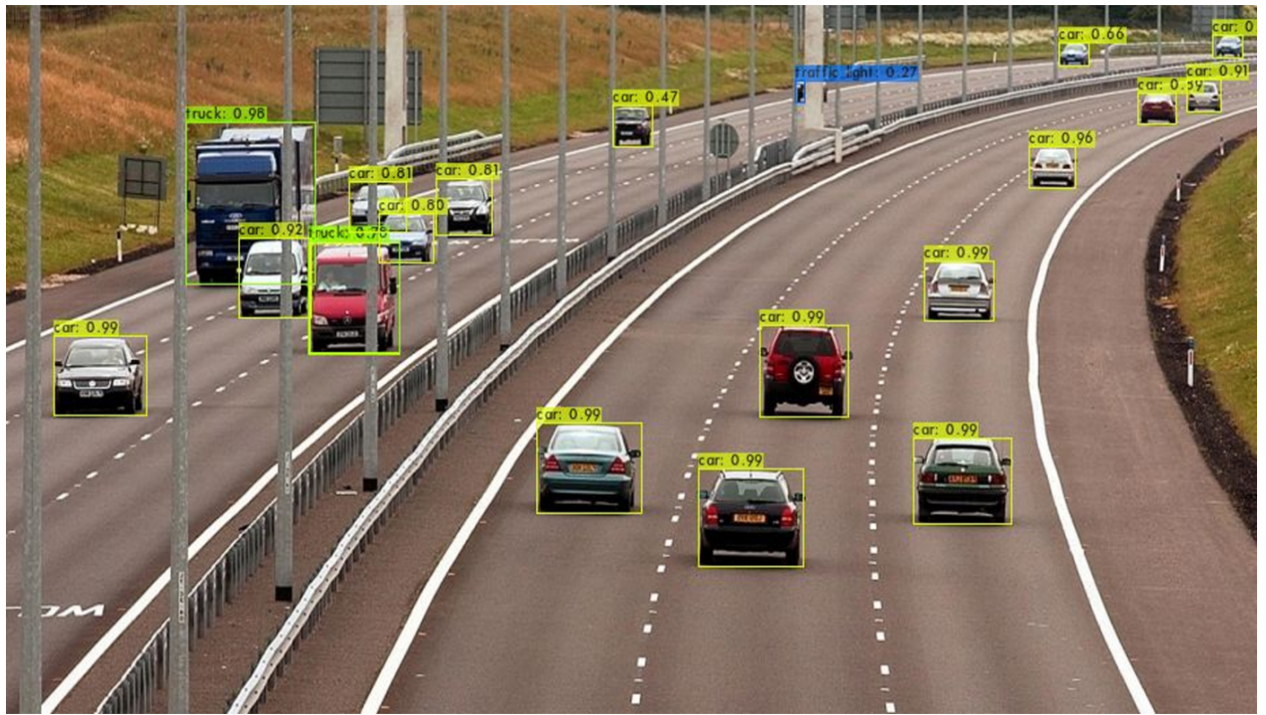


Рис. 3.4 Після обробки

Повторимо дію з іншим зображеннями



Рис. 3.5 До обработки



Рис.

3.6 Після обробки



Рис. 3.7 До обробки



Рис. 3.8 Після обробки

Як ми бачимо алгоритм працює справно, визначає різні типи автомобільної техніки під різними кутами верхньої проекції.

Висновок до розділу 3

У процесі розробки проекту була приділена значна увага аналізу та пошуку підходящих алгоритмів та середовищ розробки, а також надійності та точності результатів що отримуємо, що забезпечило високу відповідність вимогам та відповідності меті поставленого завдання. В рамках процесу були проведені різноманітні дослідження для підтвердження точності отриманих результатів.. Тестування на реальних зображеннях показало перспективність та важливість даних досліджень. Проект представляє собою покращення ефективності використання безпілотних літальних апаратів за допомогою штучного інтелекту. Ці технології дозволяють зняти навантаження з пілота та зменшити фактор людської похибки.

ВИСНОВКИ

1. Застосування штучного інтелекту разом з безпілотними літальними апаратами є технологією майбутнього впливати на розвиток якої можна ми можемо тут і зараз. Застосування безпілотних літальних апаратів з штучним інтелектом можна буде побачити вже найближчими роками.

2. Хоч результати отримані не вражають своєю швидкістю, проте використовуючи потужніші машини та довершеніші алгоритми і технології можна добити кращих результатів у реальному часі, як наприклад YOLOv8.

4. Отже, розроблений алгоритм є функціонуючою системою яку в подальшому можна використовувати в більш довершених системах розпізнавання автомобільного транспорту.

ПЕРЕЛІК ПОСИЛАНЬ

1. Веб-ресурс: https://www.ukrmilitary.com/2016/08/furya.html#google_vignette
2. Михайло Жирохов. Зброя перемоги
3. Веб-ресурс: <https://deviro.ua/ukr/ciconiavtol>
4. Веб-ресурс: <https://medium.com/@ilyesrezgui46/the-coco-dataset-all-you-need-to-know-ae465c780bb5>
5. Веб-ресурс: <https://cocodataset.org/#explore>
6. Веб-ресурс: https://defence-ua.com/video/bezpilotnij_kompleks_furija_povitrjana_rozvidka_dlja_armiji_ukrajini_vid_nvp_atlon_avia-16.html
7. Веб-ресурс: <https://old.defence-ua.com/index.php/home-page/2484-nvp-atlon-avia-vykonuye-postavky-bpak-a1-sm-furiiia-v-interesakh-zamovnyka>
8. Веб-ресурс: <https://www.ukrmilitary.com/2016/08/furya.html>
9. Веб-ресурс: <http://uav-ua.com/ukr/leleka-100>

ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ІНФОРМАЦІЙНО-
КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ
ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ КАФЕДРА
ШТУЧНОГО ІНТЕЛЕКТУ

КВАЛІФІКАЦІЙНА РОБОТА на тему: «Розробка алгоритму розпізнавання автомобіля на зображеннях з відеокамери БПЛА»

Виконала: здобувач вищої освіти
гр. ШІД-41 Богдан ЛЕВУСЬ
Керівник: кандидат технічних
наук, доцент Олександр
ЗВЕНИГОРОДСЬКИЙ

2024

Мета роботи – розробити алгоритм розпізнавання типу автомобіля на зображеннях з відеокамери БПЛА для підвищення ефективності їх використання.

Об'єкт дослідження – процес впровадження та оптимізації автоматизації обробки даних з зображень відеокамер БПЛА.

Предмет дослідження – алгоритм розпізнавання автомобільної техніки.

Короткий зміст роботи:

Розробка алгоритму розпізнавання автомобільного транспорту з метою підвищення ефективності використання БПЛА та автоматизації процесів органів управління. Використання штучного інтелекту з метою унеможливлення людських помилок.

Безпілотні літальні апарати

повітряне судно, призначене для виконання польоту без пілота на борту, керування польотом здійснюється закладеною програмою або за допомогою спеціальної станції керування, що знаходиться поза повітряним судном



Рис 1.1 Різні БПЛА



Рис 1.2 БПЛА ШАРК

Класифікація БПЛА

UAS CLASSIFICATION						
Class	Category	Normal Employment	Normal Operating Altitude	Normal Mission Radius	Primary Supported Commander	Example Platform
Class III (> 600 kg)	Strike/ Combat *	Strategic/National	Up to 65,000 ft	Unlimited (BLOS)	Theatre	Reaper
	HALE	Strategic/National	Up to 65,000 ft	Unlimited (BLOS)	Theatre	Global Hawk
	MALE	Operational/Theatre	Up to 45,000 ft MSL	Unlimited (BLOS)	JTF	Heron
Class II (150 kg - 600 kg)	Tactical	Tactical Formation	Up to 18,000 ft AGL	200 km (LOS)	Brigade	Hermes 450
Class I (< 150 kg)	Small (>15 kg)	Tactical Unit	Up to 5,000 ft AGL	50 km (LOS)	Battalion, Regiment	Scan Eagle
	Mini (<15 kg)	Tactical Sub -unit (manual or hand launch)	Up to 3,000 ft AGL	Up to 25 km (LOS)	Company, Platoon, Squad	Skylark
	Micro ** (<66 J)	Tactical Sub -unit (manual or hand launch)	Up to 200 ft AGL	Up to 5 km (LOS)	Platoon, Squad	Black Widow

Рис 1.3 Класифікація БПЛА НАТО

База даних

Була вибрана COCO Dataset як найбільша та найзручніша база даних



Рис 1.4 Приклад зображення у COCO Dataset

Нейронна мережа

На основі YOLOv4 та бази даних COCO Dataset була створена нейронна мережа для визначення різного автомобільного транспорту

```

# define helper functions
def imshow(path):
    import cv2
    import matplotlib.pyplot as plt
    %matplotlib inline

    image = cv2.imread(path)
    height, width = image.shape[:2]
    resized_image = cv2.resize(image, (3*width, 3*height), interpolation = cv2.INTER_CUBIC)

    fig = plt.gcf()
    fig.set_size_inches(18, 10)
    plt.axis("off")
    plt.imshow(cv2.cvtColor(resized_image, cv2.COLOR_BGR2RGB))
    plt.show()

# use this to upload files
def upload():
    from google.colab import files
    uploaded = files.upload()
    for name, data in uploaded.items():
        with open(name, 'wb') as f:
            f.write(data)
        print ('saved file', name)

# use this to download a file
def download(path):
    from google.colab import files
    files.download(path)
  
```

Рис 1.5 Частина коду



Рис 1.6 Зразок роботи

Висновки

1. Застосування штучного інтелекту разом з безпілотними літальними апаратами є технологією майбутнього впливати на розвиток якої можна ми можемо тут і зараз. Застосування безпілотних літальних апаратів з штучним інтелектом можна буде побачити вже найближчими роками.

2. Хоч результати отримані не вражають своєю швидкістю, проте використовуючи потужніші машини та довершеніші алгоритми і технології можна добити кращих результатів у реальному часі, як наприклад YOLOv8.

4. Отже, розроблений алгоритм є функціонуючою системою яку в подальшому можна використовувати в більш довершених системах розпізнавання автомобільного транспорту

Дякую за увагу!