

УДК 629.7.05:004.93
УКПП
№ держреєстрації 0117U003934
Інв. №

Міністерство освіти та науки України
Сумський державний університет
(СумДУ)

40007, м. Суми, вул. Римського-Корсакова, 2
тел. (0542) 33-54-79 факс (0542) 33-54-79

ЗАТВЕРДЖУЮ

Проректор з наукової роботи
д-р. фіз.-мат. наук, професор

_____ А.М. Черноус

ЗВІТ

ПРО НАУКОВО-ДОСЛІДНУ РОБОТУ

Інтелектуальна автономна бортова система безпілотної літального апарату
для ідентифікації об'єктів на місцевості

ВИГОТОВЛЕННЯ ДОСЛІДНОГО ЗРАЗКА БЕЗПІЛОТНОГО АПАРАТУ ТА ВПРОВАДЖЕННЯ
РЕЗУЛЬТАТІВ ДОСЛІДЖЕННЯ
(остаточний)

Керівник НДР
доцент, канд. техн. наук

В.В. Москаленко

2020

Рукопис завершено 05 жовтня 2020 р.

Результати цієї роботи розглянуто на засіданні наукової ради СумДУ, протокол від
24.09.2020 № 2

СПИСОК АВТОРІВ

Керівник НДР, канд. техн. наук, провідн. наук. співроб. (05.10.2020)	Москаленко В. В. (вступ, висновки, підрозділи 1.1, 1.2, 2.3)
Відповідальний виконавець канд. техн. наук, старш. наук. співроб. (05.10.2020)	Москаленко А. С. (підрозділи 1.1, 2.3)
Виконавці: канд. техн. наук, старш. наук. співроб. (05.10.2020)	Берест О. Б. (підрозділи 1.1, 1.3)
канд. техн. наук, старш. наук. співроб. (05.10.2020)	Мартиненко С. С. (підрозділ 1.3)
канд. техн. наук, старш. наук. співроб. (05.10.2020)	Нагорний В. В. (підрозділ 2.2)
канд. фіз.-мат. наук, старш. наук. співроб. (05.10.2020)	Журба В. О. (підрозділ 2.2)
інженер II кат. (05.10.2020)	Шаєхов А.Ф. (підрозділи 1.3, 2.2)

канд.техн.наук.,
мол. наук. співроб.

(05.10.2020)

Коробов А. Г. (підрозділи 2.1, 2.2)

аспірант,
мол. наук. співроб.

(05.10.2020)

Зарецький М. О. (підрозділ 2.3)

РЕФЕРАТ

Звіт про НДР: 79 с., 3 табл., 39 рис., 18 джерел.

БОРТОВА СИСТЕМА БЕЗПІЛОТНОГО ЛІТАЛЬНОГО АПАРАТУ, ЕКСТРАКТОР ОЗНАК, ВИРІШУВАЛЬНІ ПРАВИЛА, ІНФОРМАЦІЙНО-ЕКСТРЕМАЛЬНА ІНТЕЛЕКТУАЛЬНА ТЕХНОЛОГІЯ, МАШИННЕ НАВЧАННЯ, ОПТИМІЗАЦІЯ.

Об'єкт дослідження – процес машинного навчання бортової системи безпілотного апарату та автономної ідентифікації об'єктів на місцевості.

Мета роботи – підвищення функціональної ефективності бортової системи безпілотного літального апарату, що здійснює у автономному режимі локальну навігацію та класифікаційний аналіз спостережень за умов ресурсних та інформаційних обмежень.

Методи дослідження – аналіз літературних джерел, методи розрідженого кодування, методи нейромережевого та інформаційно-екстремального моделювання.

Розроблено апаратну частину безпілотного літального апарату. Встановлено та сконфігуровано операційну систему бортового комп'ютера та пакетів фреймворку Robot Operating System. Досліджено ефективність моделей аналізу візуальних образів на випадок малої кількості розмічених даних з метою реалізації інтерактивного навчання бортової системи. Розроблено новий метод синтезу моделі класифікаційного аналізу зображень з бортової системи безпілотного апарату шляхом поєднання критеріїв і принципів теорії інформації і кодування з ідеями регуляризації в рамках геометричного підходу, який на відміну від існуючих забезпечує робастність за найскладніших у статистичному відношенні умов. З прикладної точки зору це дозволяє зменшити кількість конфігураційних налаштувань бортової системи та зменшити кількість інтерактивних взаємодій з візуальним інтерфейсом для донавчання чи перенавчання.

Результати виконання роботи впроваджено в навчальний процес під час підготовки лекційних курсів з навчальних дисциплін «Мови

програмування інтелектуальних систем» і «Introduction to Data Science». Захищено одну кандидатську дисертацію на тему “Моделі і методи інформаційної технології автономного відеомоніторингу місцевості безпілотним літальним апаратом”. Деякі результати було використано для надання послуг за господарським договором №52.17-2020.СП/01 від 11 лютого 2020 р. : «Послуги з розробки складової частини програмного забезпечення інтелектуального модуля машинного зору для наземного транспорту» (замовник: ТОВ «РОЗУМНІ ТЕХНОЛОГІЇ «ТИТУЛ»»).

ЗМІСТ

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ	7
ВСТУП	8
1 МЕТОДИ ОПТИМІЗАЦІЇ МОДЕЛЕЙ КЛАСИФІКАЦІЙНОГО АНАЛІЗУ ДАНИХ БОРТОВОЇ КАМЕРИ БЕЗПІЛОТНОГО АПАРАТУ ЗА УМОВ РЕСУРСНИХ ТА ІНФОРМАЦІЙНИХ ОБМЕЖЕНЬ	10
1.1 Аналіз існуючих методів підвищення ефективності алгоритмів розпізнавання об'єктів на місцевості.....	10
1.2 Методи оптимізації моделей класифікаційного аналізу спостережень .	26
1.3 Ентропійний інформаційний критерій ефективності вирішувальних правил бортової системи класифікаційного аналізу зображень	34
2 РЕАЛІЗАЦІЯ БОРТОВОЇ СИСТЕМИ БЕЗПІЛОТНОГО АПАРАТУ ДЛЯ АНАЛІЗУ ВІДЕОСПОСТЕРЕЖЕНЬ	38
2.1 Апаратне забезпечення безпілотного літального апарату.....	38
2.2 Програмне забезпечення бортової системи безпілотного літального апарату для аеровідеомоніторингу	52
2.3 Результати машинного навчання бортової системи безпілотного апарату на прикладі задачі інспекції труб	63
ВИСНОВКИ.....	74
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	77

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

БПЛА – безпілотний літальний апарат

ВСТУП

Малогабаритні безпілотні літальні апарати (БПЛА) набувають широкого використання як в безпековому секторі, так і в секторі промислової інспекції. Безпілотний апарат може виконувати функції літаючого сенсора у важкодоступних та небезпечних для життя людини місцях. В обох випадках актуальною задачею є підвищення рівня автономності безпілотного апарату задля захисту інформації та забезпечення успішності місій за відсутності радіозв'язку. Крім того, розвиток інтелектуальних інформаційних технологій сприяє наближенню джерела інформації до алгоритмів її обробки і прийняття рішень в рамках концепції крайових обчислень. Це дозволяє розвантажити комунікаційні канали інфокомунікаційного середовища та знизити накладні витрати, що пов'язані з аналізом даних, за рахунок моделювання когнітивних процесів, притаманних людині під час прийняття рішень. Однак традиційні технології інтелектуального аналізу даних потребують знаних обсягів розмічених навчальних даних для забезпечення в режимі прийняття рішень з допустими для практичного використання рівнем достовірності. При цьому збір і розмітка навчальних даних залишається трудомісткою задачею, тому дослідження і впровадження нових інтелектуальних алгоритмів, невибагливих до обчислювальних ресурсів та обсягу розмічених даних, є актуальними задачами.

Розробка, що пропонується, є корисною з точки зору зменшення накладних витрат, пов'язаних з експлуатацією і конфігуруванням БПЛА до нових умов і задач. Це дозволить підвищити конкурентоздатність розроблених рішень на ринку безпілотної авіації.

Заключний звіт складається із вступу, двох розділів, висновків і переліку посилань.

Перший розділ присвячено розробці методів підвищення ефективності режиму навчання за умов обмеженого обсягу розмічених даних та режиму розпізнавання за умов обмеженого обсягу обчислювальних ресурсів бортової системи БПЛА.

Другий розділ присвячено реалізації інформаційної технології інтелектуального аналізу візуальних даних бортової системи БПЛА. Здійснюється аналіз результату машинного навчання бортової системи на прикладі задачі інспекції труб. Розглянуто складові та принципи конфігування апаратно-програмного забезпечення БПЛА.

Результати наукових досліджень, одержаних виконавцями проекту, опубліковано в працях [1]–[5].

1 МЕТОДИ ОПТИМІЗАЦІЇ МОДЕЛЕЙ КЛАСИФІКАЦІЙНОГО АНАЛІЗУ ДАНИХ БОРТОВОЇ КАМЕРИ БЕЗПІЛОТНОГО АПАРАТУ ЗА УМОВ РЕСУРСНИХ ТА ІНФОРМАЦІЙНИХ ОБМЕЖЕНЬ

1.1 Аналіз існуючих методів підвищення ефективності алгоритмів розпізнавання об'єктів на місцевості

Ефективність розпізнавання складних візуальних об'єктів залежить від способу ознакового опису зображень. При цьому питаннями формування ознакового опису займається наука про подання даних (Representation Learning) або наука про навчання ознак (Feature learning). Дослідниками цієї галузі було сформульовано ряд положень щодо ознакового опису спостережень, які є спільними для всіх задач інтелектуального аналізу даних. До таких положень належать [6]:

- ієрархічна організація пояснювальних факторів (першопричин);
- множинність пояснювальних факторів (першопричин);
- навчання з частковим залученням вчителя;
- спільність факторів під час розв'язання різних задач;
- гіпотеза багатовидів;
- гіпотеза про природну кластеризацію;
- просторово-часова зв'язаність;
- розрідженість ознакового подання;
- простота залежності високорівневого подання факторів.

Поняття, використовувані для опису середовища, можуть бути визначені в термінах інших більш абстрактних понять, тобто ієрархічно. Тобто високорівневі ознаки формуються методом композиції низькорівневих ознак (рис. 1.1). Функція f для екстракції ознак може бути описана у вигляді композиції n шарів трансформації простору ознак $f = f_1 \circ f_2 \circ \dots \circ f_n$. Ієрархічні моделі порівняно з однорівневими мають більшу ємність і потенційно можуть бути навчені значно більшим абстраціям ознак на верхніх рівнях.

За допомогою ієрархічного подання можна ефективно вирішувати два взаємопов'язані але різні завдання : навчання інваріантним ознакам (invariant features) і навчання розділенню пояснювальних факторів. Метою навчання інваріантних ознак є формування високорівневих ознак, що є нечутливими до таких змін у вхідних даних, що є неінформативними і нецікавими для подальшого оброблення. Метою навчання розділенню пояснювальних факторів є формування якомога більшої кількості незалежних ознак (факторів) різного рівня абстрактності, оскільки інформативність ознак для подальшого оброблення може бути невідомою. При цьому сформований ознаковий опис може повторно використовуватися для вирішення різних завдань.

Вхідні дані є результатом взаємодії багатьох пояснювальних факторів. Тому навчання моделі новому фактору приводить до його узагальнення в конфігураціях інших факторів. Саме ця ідея лежить в основі розподіленого подання даних. Кожен параметр може повторно бути задіяним для кодування різних вхідних спостережень чи частин вхідного спостереження. При цьому ці спостереження можуть навіть не бути близькими сусідами. У розподіленому поданні даних експоненційно більша кількість ознак або прихованих змінних можуть бути активовані вхідним сигналом, у той час як в алгоритмах із локальним узагальненням різні частини вхідного простору асоціюються лише зі своїм персональним набором параметрів [6, 7]. Отже, розподілене подання даних може кодувати більшу кількість різноманітних вхідних конфігурацій ніж подання з використанням локального узагальнення.

Під час синтезу моделей аналізу даних багато дослідників використовують гіпотезу природньої кластеризації. Вона полягає в тому, що люди виділяють категорію (клас) і дають їй назву на основі статистичної схожості спостережень, віднесених до відповідної категорії. Локальні зміни в багатовидах мають тенденцію до збереження категорії, а лінійна інтерполяція між зразками різних класів проходить у загальному випадку через області з низькою щільністю ймовірності. Тобто для ознакового подання даних X щільність імовірності

$P(X|Y = k)$ для різних категорій k прагне розділятися, а не накладатися. Іншими словами категорії асоціюється з різними багатовидами.

Важливе практичне значення має встановлення статистичного взаємозв'язку між навчанням без учителя та навчанням з учителем. Це дозволяє ефективно використати нерозмічені навчальні дані, які одержати простіше й дешевше, для формування інформативного ознакового опису. Було доведено, що ознакове подання даних X , зручне для обчислення ймовірнісного розподілу $P(X)$, є зручним і для навчання $P(Y|X)$, де Y – цільова змінна вирішувальних правил [6]. Саме в цьому й полягає ідея навчання з частковим залученням учителя. Використання навчання з частковим залученням учителя дозволяє уточнити межі категорій, беручи до уваги, що роздільна гіперповерхня повинна проходити через області простору з низькою щільністю ймовірності.

У багатозадачних моделях аналізу даних може досягатися вищий рівень узагальнення в навчанні внаслідок підсилення статистичного взаємозв'язку між задачами, у яких використовуються спільні пояснювальні фактори. Такі моделі є дуже цінними особливо з точки зору передачі знань з метою їх адаптації до нових задач.

Необхідність зниження розмірності часто ґрунтується на гіпотезі багатовидів, згідно з якою основна щільність ймовірності даних зосереджена біля регіонів, що мають набагато меншу розмірність, ніж оригінальний простір вхідних даних. В умовах апріорної невизначеності інформативність окремих ознак у навчальних даних невідома. Кожне спостереження може мати велику кількість статистично залежних, неінформативних та дезінформуючих ознак. При цьому можна зіштовхнутися з проблемою «прокляття розмірності» (curse of dimensionality), що полягає у збільшенні розміру простору, доступного для «розсіювання» точок даних, унаслідок збільшення кількості вимірів – ознак (рис. 1.1). Це призводить до експоненційного росту необхідного розміру навчальної вибірки для щільнішого заповнення важливої для аналізу області простору з метою виявлення будь-яких залежностей.

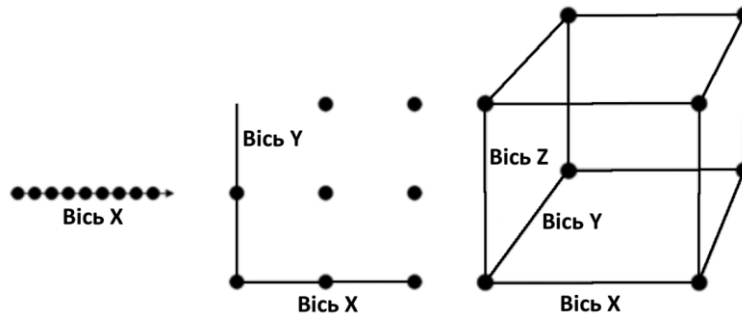


Рисунок 1.1 – Ілюстрація до проблеми «прокляття розмірності»

На боротьбу з проблемою «прокляття розмірності» спрямовано багато класичних методів зниження розмірності, серед яких найбільш популярним є метод головних компонент. Проте ефективність цих методів є невисокою в умовах складної нелінійної залежності факторів.

Для подолання проблеми «прокляття розмірності» за умов складних нелінійних залежностей між факторами було розроблено багато локальних непараметричних алгоритмів навчання. Серед них найбільш популярними стали алгоритми, що ґрунуються на ядрах (kernel machines). У більшості подібних алгоритмів використано принцип локального узагальнення (local generalization). Ці алгоритми ґрунтувалися на припущеннях про достатню гладкість цільової функції f , якій треба навчитися, тобто $f(x) \approx f(y)$ за умов $x \approx y$. Однак для ефективності ядерних алгоритмів навчальна вибірка повинна містити такі навчальні зразки, за допомогою яких можна явно вивчити всі піки та впадини цільової функції. У такому разі узагальнення досягають за рахунок локальної інтерполяції між сусідніми зразками навчальної вибірки. Проте кількість піків і впадин цільової функції може зростати експоненційно з кількістю взаємодіючих факторів, якщо алгоритм працює з «сирими» даними на вході. Тому алгоритми, в основу яких покладені властивість гладкості цільової функції та лінійна залежність факторів, краще використовувати на попередньо підготованому ознаковому просторі. Зручне подання даних спрощує завдання синтезу вирішувальних правил.

В ефективних високорівневих поданнях даних фактори зв'язані один з одним через прості залежності. Тому синтез вирішувальних правил варто

здійснювати в межах підходів, що характеризуються найбільшою ефективністю з точки зору обчислень. Якщо ознакове подання здійснювали з самого початку з метою забезпечення максимальної інформативності для конкретних вирішувальних правил, то вирішувальні правила можуть бути максимально прості, на-приклад, у вигляді лінійної моделі чи системи порогів. Проте у разі навчання інваріантного ознакового подання для ефективної адаптації до однієї чи декількох задач вирішувальні правила можуть містити додатковий шар прихованих змінних і нелінійні перетворення.

Під час синтезу моделей аналізу даних варто враховувати обмеження на зміну «крізь час і простір». Тобто спостереження, сформовані в сусідніх областях простору чи одержані послідовно в часі, повинні бути асоційованими з однаковими значеннями відповідної категорії понять, чи, приводити до невеликого руху по поверхні багатовиду високої щільності. Деякі дослідники вводили додаткову регуляризуючу складову до функціоналу якості, що враховує різницю значень ознак в різні моменти часу. Інші використовують апріорні знання про топологічну структуру даних для використання локальних рецептивних полів нейронів та операторів агрегації відгуку нейронів на сусідні ділянки вхідного простору в одне компактне подання. В обох випадках було досягнуто підвищення ефективності вирішувальних правил, побудованих із використанням одержаного ознакового подання. Крім цього, просторово-часова зв'язаність спостережень обґрунтовує спосіб розширення навчальних даних, основою якого є застосування невеликих випадкових деформацій образів в існуючих навчальних даних. При цьому деформувальні зміни повинні бути обмежені, щоб зберегти відношення згенерованих зразків до тієї ж категорії, що й оригінальний зразок.

Для будь-якого даного спостереження x лише мала частина з усіх можливих факторів є значимою. Більшість виділених ознак повинні бути нечутливими до малих змін спостереження x . Тобто більшість детектованих ознак повинна бути нульовою. Цю властивість називають розрідженістю (sparsity). Вона може бути досягнута за рахунок різноманітних технік, що ґрунтуються на ефекті редукції причини (explaining away). Редукція причини полягає у зв'язуванні двох апріорно

не зв'язаних причин події, якщо з'являється спостереження цієї події. У цих техніках можуть використовувати спеціальної форми приховані змінні h , більшість із яких прямують до нуля, або спеціальна нелінійність, значення якої знаходяться переважно близько нуля, або обмеження матриці Якобіана (або похідних функції) перетворення вхідних даних в обране подання [6]. При цьому знаходження апостеріорної ймовірності розподілу для активації прихованих факторів (причин) h , $p(h / x)$, часто використовуване як базис для екстракції ознак, виявляється складною задачею. У разі дискретного h задача взагалі може не мати розв'язку.

У працях [8] запропоновано багато методів розрідженого кодування, які вхідному вектору x ставлять у відповідність приховане подання даних h (вектор випадкових змінних або вектор ознак) через лінійний взаємозв'язок W , що називається словником. При цьому в методах розрідженого кодування використовують спеціальне обмеження для забезпечення саме розрідженої активації h під час кодування x . Розріджене кодування можна розглядати як задачу відновлення вектора ознак, що асоційований зі значенням входу x згідно з виразом

$$h^* = f(x) = \arg \min_h \|x - Wh\|_2^2 + \lambda \|h\|_1. \quad (1.1.1)$$

Навчання словника W може бути виконано в процесі мінімізації середньоквадратичної помилки реконструкції

$$W^* = \arg \min_W \sum_t \|x^{(t)} - Wh^{*(t)}\|_2^2, \quad (1.1.2)$$

де $x^{(t)}$ – t -й навчальний зразок;

$h^{*(t)}$ – відповідний розріджений код.

Розріджене кодування також має ймовірнісну інтерпретацію, згідно з якою кодування полягає у відновленні максимуму апостеріорної ймовірності h , тобто

$$h^* = \arg \max_h p(h | x). \quad (1.1.3)$$

Процес навчання словника W в межах ймовірнісної інтерпретації може здійснюватися як максимізація правдоподібності даних, одержаних шляхом використання методу максимізації апостеріорної ймовірності h^* , тобто

$$W^* = \arg \max_W \prod_t p(x^{(t)} | h^*(t)). \quad (1.1.4)$$

У працях [9, 19] було запропоновано здійснювати навчання словника W (dictionary learning) окремо на основі автоенкодерів, обмежених машин Больцмана, методів кластер-аналізу та векторного квантування. Водночас у разі використання автоенкодерів або обмежених машин Больцмана необхідно до функції втрат уводити штраф за нерозрідженість. За умови використання методів векторного квантування необхідно забезпечити оптимальну взаємну узгодженість (ступінь неортогональності) елементів словника W для ефективного кодування даних.

Після навчання словника W кодування зразка даних x у результативний вектор ознак h здійснюють на основі функції розрідженого кодувальника $f(x; W)$. Популярними алгоритмами реалізації цієї функції є алгоритми узгодженого переслідування (matching pursuit), ортогонального узгодженого переслідування (orthogonal matching pursuit), мішка переслідувачів (bag of pursuits) та інші.

Розріджене кодування навіть із надлишковим словником дозволяє здійснювати ймовірнісне виведення під час пошуку h^* шляхом вибору найбільш оптимальних базисів та обнулення решти, незважаючи на те, що вони мають високий ступінь кореляції зі входом. Цієї властивості не має ні в автоенкодерів, ні в обмежених машинах Больцмана. На відміну від автокодувальників та обмежених машин Больцмана вектор ознак (код) в алгоритмі розрідженого кодування вільно змінюється для кожного зразка даних. Платою за це є наявність

додаткового циклу для оптимізації h^* , що збільшує обчислювальні витрати під час екстракції розрідженого ознакового опису.

Для підвищення ефективності розрідженого кодування було запропоновано метод прогнозувальної розрідженої декомпозиції (Predictive Sparse Decomposition), де ресурсно затратний нелінійний ітераційний процес пошуку h (кодування) замінюється на швидку неітеративну апроксимацію під час розпізнавання [6]. Основна ідея цього методу полягає у використанні такого критерію навчання

$$J_{PSD} = \sum_t \lambda \left\| h^{(t)} \right\|_1 + \left\| x^{(t)} - Wh^{(t)} \right\|_2^2 + \left\| h^{(t)} - f_\alpha(x^{(t)}) \right\|_2^2,$$

де $x^{(t)}$ – вхідний вектор t -го навчального зразка;

$h^{(t)}$ – оптимізований прихований вектор (подання) даних для t -го навчального зразка;

$f_\alpha(\cdot)$ – функція кодувальника, яка у найпростішому варіанті має вигляд

$$f_\alpha(x^{(t)}) = \tanh(b + W^T x^{(t)})$$

У праці [10] було показано, що побудову швидкого апроксимувального кодувальника $f_\alpha(\cdot)$ можна здійснити на основі операції усадження (shrinkage), ансамблю дерев рішень та інших обчислювально ефективних моделей. В інших працях [8, 9] було показано, що критерій розрідженого кодування може бути досить негладкою та недиференційованою функцією, що ускладнює повну оптимізацію моделі подання даних. Тому багато методів спрямовані на згладжування результативних кодів розрідженого кодування, щоб забезпечити сумісне навчання етапу розрідженого кодування з іншими етапами обчислення глибокої архітектури.

У працях [12] було запропоновано архітектуру так званих сіамських нейронних мереж для побудови інваріантного подання вхідних даних та зниження розмірності. Сіамська мережа складається з двох екстракторів ознак зі спільними параметрами (рис. 1.2).

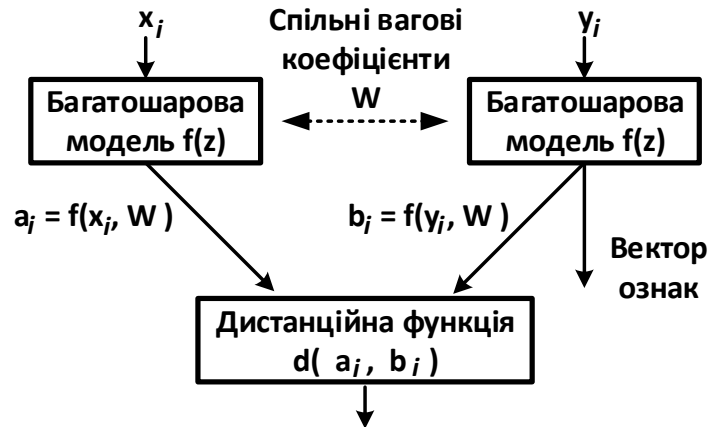


Рисунок 1.2 – Структура сіамської моделі аналізу даних

Параметри екстрактора ознак W вважають оптимальними, якщо для пари семантично схожих зразків вихід мережі має близьке до нуля значення, а для пари семантично відмінних зразків – значення виходу, близьке до одиниці. Тобто навчання сіамських мереж можна вважати навчанням метрики подібності (metric learning). Навчена сіамська мережа може виконувати роль дистанційної метрики для «лінивих» алгоритмів побудови вирішувальних правил, таких як метод k -найближчих сусідів. На цьому принципі ґрунтуються алгоритми навчання з одного погляду (one-shot learning). Також дані мережі можуть розглядатися як адаптивне ядро для ядерних алгоритмів побудови вирішувальних правил. При цьому задача побудови функції близькості (similarity function) є еквівалентною задачі побудови простору інваріантних ознак. Тому навчений екстрактор можна також використовувати окремо, без близнюка. Для навчання таких мереж використовуються контрастна або триплетна функції втрат і їх модифікації.

У галузі машинного зору набули поширення підходи, основані на застосуванні принципів лінгвістики до аналізу зображення за його фрагментами,

які асоціюють з візуальними словами [13]. При цьому визначення координат таких фрагментів відбувається за результатами роботи детектора найбільш помітних фрагментів (ключових точок), які характеризуються стійкістю до геометричних та фотометричних перетворень, або вибираються з незмінним кроком, покриваючи все зображення.

Найбільший успіх в задачах інтелектуального аналізу зображень було отримано в рамках технології згорткових нейронних мереж. В даній мережі згідно принципу просторової зв'язаності кожен нейрон має обмежене рецептивне поле і послідовно сканує зображення, а сприймає його повністю, тобто не зв'язаний з всіма пікселями. Це дозволяє значно зменшити кількість параметрів і обчислювальну складність моделі екстракції ознакового опису за рахунок повторного використання вагових коефіцієнтів нейронів.

З 1989 року по сьогодні було виконано ряд архітектурних вдосконалень згорткової нейронної мережі. Вдосконалення стосувалися таких аспектів як оптимізація параметрів, регуляризація, структурне переформулювання тощо. Проте останні тенденції в модернізації згорткової мережі пов'язані з реструктуризацією існуючих компонентів обробки даних та проектуванням нових блоків і модулів. Залежно від типу архітектурних модифікацій згорткові мережі можна класифікувати на сім різних груп (рис. 1.2) [14]:

- експлуатація простору (*spatial exploitation*), що основана на використанні різномасштабних фільтрів;
- експлуатація глибини (*depth exploitation*), тобто збільшення кількості шарів і міри щодо забезпечення незатухання градієнту на нижніх шарах;
- експлуатація мульти-шляхів (*multi-path exploitation*), тобто використання міжшарових з'єднань та псевдо-ансамблювання на основі *dropout* і *drop-connection*;
- експлуатація ширини (*width exploitation*), тобто міри щодо збільшення ширини нейронної мережі за рахунок мульти-з'єднань;
- експлуатація карти ознак (*feature map exploitation*), тобто підвищення інформативності ознак за рахунок їх селекції/зважування на основі *Squeeze-*

Excitation-модуля;

– експлуатація підсилення каналів (channel boosting exploitation), тобто додавання штучних каналів до оригінальних каналів, що сформовані за рахунок переносу знань (transfer learning), використання генеративних моделей навчання або агрегації частинних описів спостережень (на рівні модельностей);

– використання модулів уваги (attention based CNNs), тобто використання м'яких масок для карти ознак з метою фокусування моделі на важливих регіонах зображення/карти ознак.

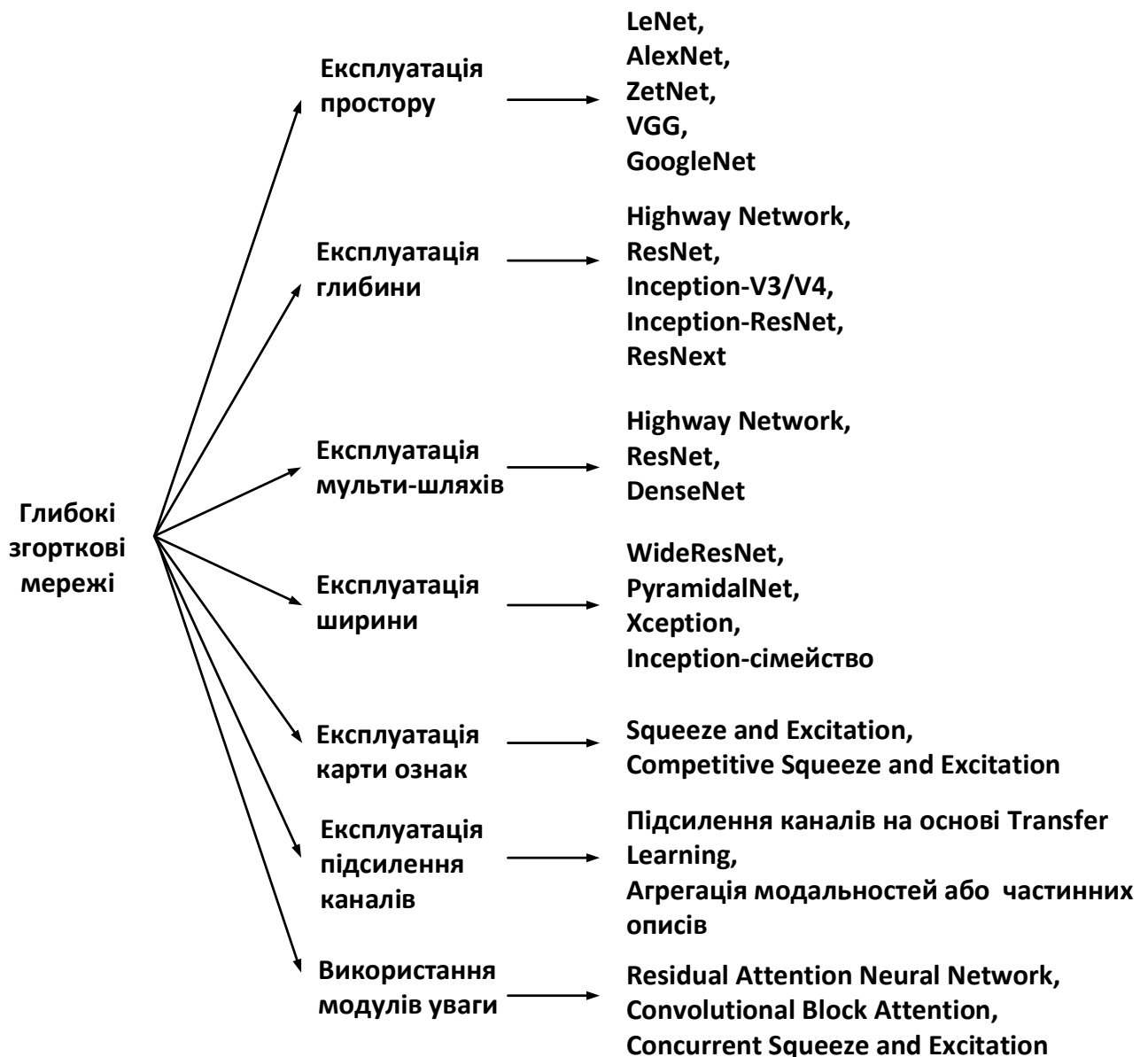


Рисунок 1.3 – Таксономія багатoshарових згорткових нейронних мереж за способом їх удосконалення

Для аналізу багатоспектральної або багатомодальної візуальної інформації

використовуються методи об'єднання (fusion) джерел різної модальності/спектру для їх обробки в рамках єдиної моделі. Прикладом подібної задачі є розпізнавання об'єктів на сцені, яка фіксується кольоровою (RGB) камерою, тепловізором та короткохвильовим радаром. Існує три основні методи об'єднання модельностей [15]:

- раннє об'єднання (Early Fusion) (рис. 1.4);
- пізнє об'єднання (Late Fusion) (рис. 1.5);
- проміжне об'єднання (Intermediate fusion) (рис. 1.6).

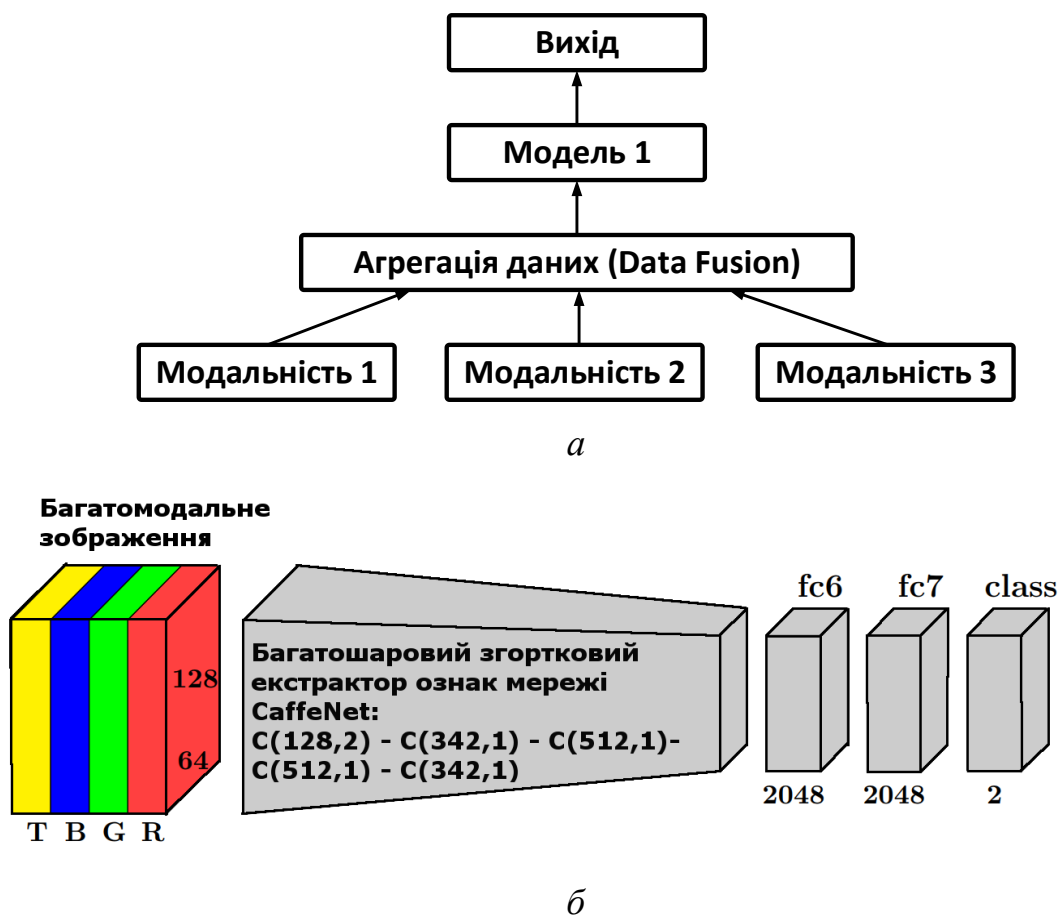


Рисунок 1.4 – Ілюстрація методу раннього об'єднання модальностей :
a – загальний випадок; *б* – приклад згорткової мережі з додатковим тепловим каналом на її вході

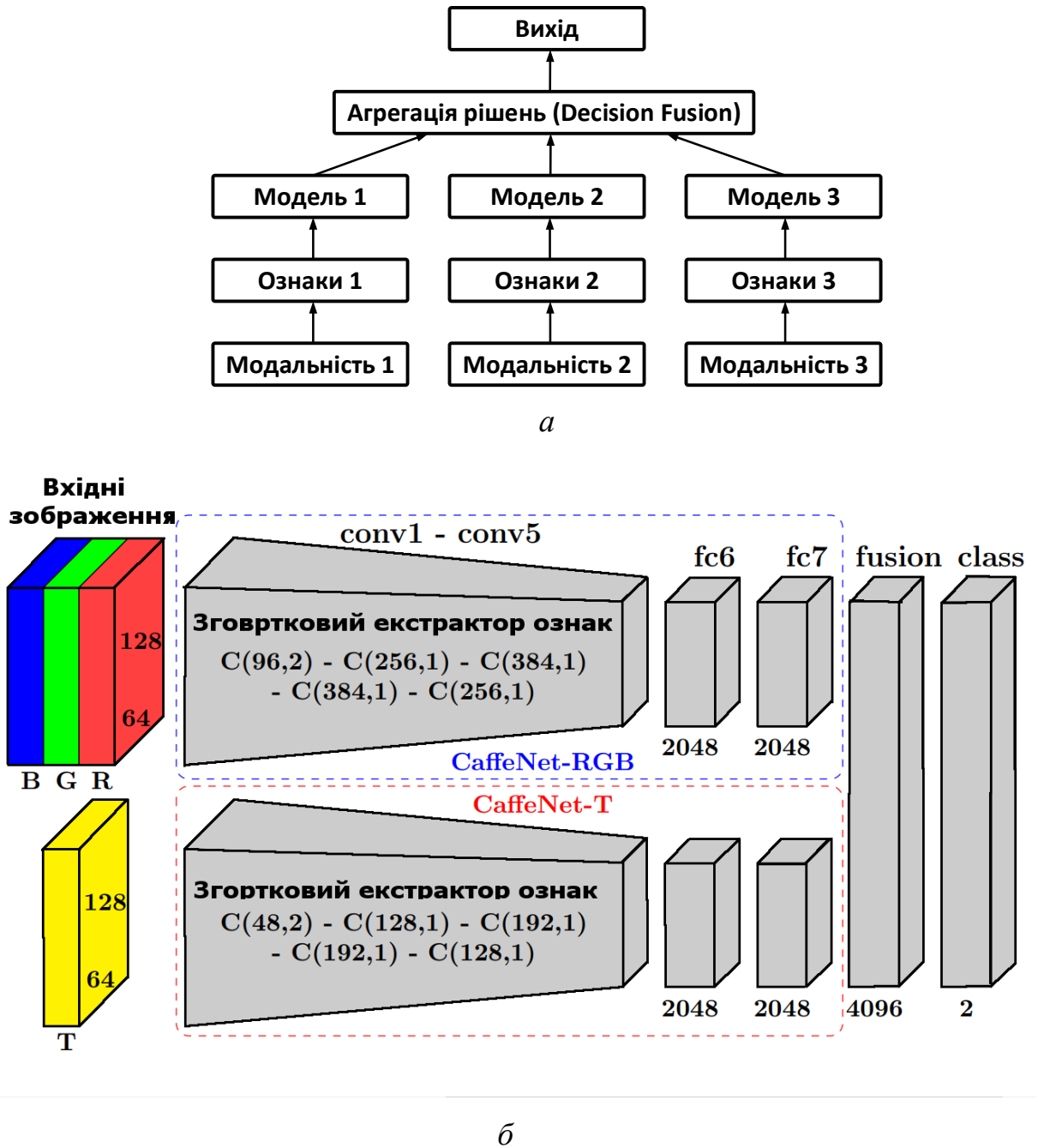


Рисунок 1.5 – Ілюстрація методу пізнього об'єднання модальностей :
а – загальний випадок; *б* – приклад агрегації виходів двох згорткових мереж зі входами різної модальності

Звичайно існують більш складні способи об'єднання джерел інформації та комплексування моделей. На рис. 1.6 проілюстровано змішану схему, або схему з проміжною агрегацією модальностей.

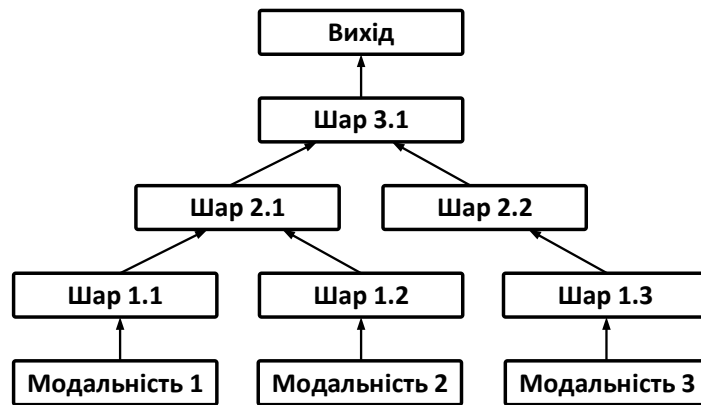


Рисунок 1.6 – Ілюстрація методу проміжного об'єднання модальностей у загальному випадку

Для локалізації об'єкта інтересу на зображенні використовують моделі і методи детектування, що використовують сформований екстрактор ознакового опису зображень. Найпростішим методом детектування об'єкта інтересу на зображенні є алгоритм ковзного вікна (sliding window). Однак, у цього методу є певний недолік. Проходження вікном всіх положень на зображенні на різних масштабах вимагає досить великої кількості часу $O(h*w)$, де h і w – висота і ширина зображення. Для прискорення алгоритму сканування зображення було запропоновано модифікацію методу sliding window – метод адаптивного кроку ковзного вікна (англ. Run-Time Adaptive Sliding Window, RASW). RASW – ефективний метод зміни кроку руху скануючого вікна під час обробки без втрати точності. Алгоритм дозволяє знизити витрати часу на обробку областей зображення, які не містять шуканих об'єктів, за рахунок збільшення кроку сканування залежно від ймовірності належності об'єкта інтересу [16].

Сучасні моделі детектування об'єктів на основі згорткових нейронних мереж розвиваються паралельно в двох різних напрямках (рис. 1.7) :

- модель з генератором пропозицій областей інтересу з наступною їх класифікацією на категорії об'єктів;
- модель, що вирішує задачу детектування шляхом прямого вирішення задач класифікації та регресії для прогнозування категорій і обмежувальних прямокутників об'єктів інтересу відповідно.

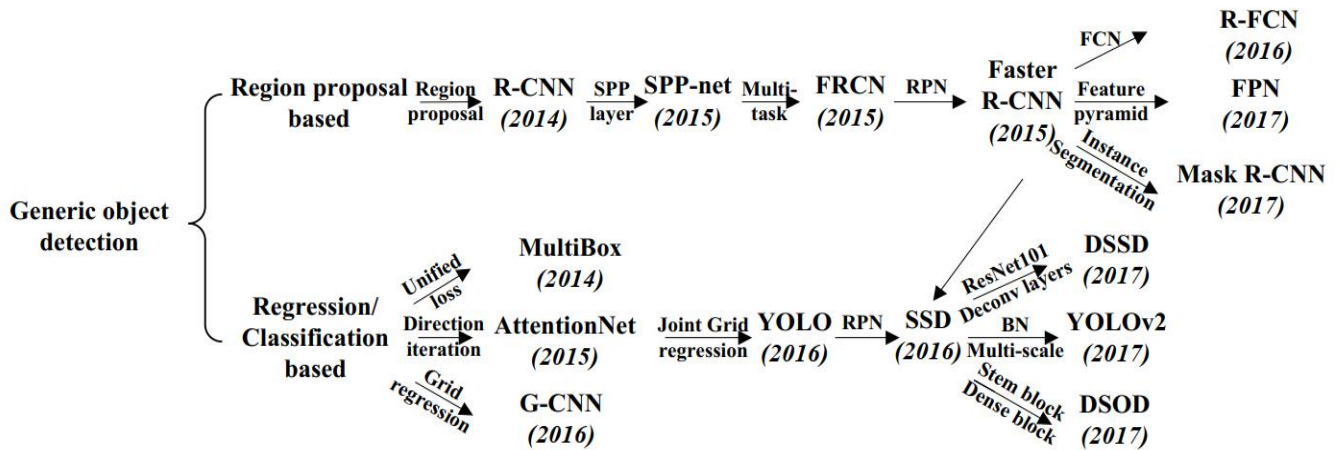


Рисунок 1.7 – Схема розвитку моделей детектування об'єктів на зображенні з використанням згорткових нейронних мереж [17]

До методів з генератором пропозиції регіонів відносяться R-CNN, SPP-мережа, Fast R-CNN, Faster R-CNN, R-FCN, FPN та Mask R-CNN. До методів, основаних на регресії і класифікації, відносяться MultiBox, AttentionNet, G-CNN, YOLO, SSD, YOLOv2, DSSD та DSOD. При цьому найбільш популярними є три архітектури : SSD, YOLOv2 та Faster R-CNN. Для забезпечення високої точності детектування, зокрема локалізації, модель Faster-RCNN потребує менше ресурсів ніж модель SSD. Однак для задач, які не вимагають високої точності локалізації чи аналізу малорозмірних об'єктів, моделі SSD мають перевагу в швидкодії.

Архітектури моделей поступово досягають насичення технічною новизною. При цьому домінуючим методом машинного навчання є застосування алгоритму зворотного поширення помилки, проте він і має ряд недоліків:

- повільна збіжність;
- потреба значного обсягу розмічених навчальних даних та обчислювальних ресурсів для забезпечення прийняттого результату;
- невизначеність щодо оптимальних значень гіперпараметрів;
- резервування інформаційної ємності відбувається за рахунок гіперпараметризації, що збільшує накладні витрати і обумовлює схильність до перенавчання.

Одним із шляхів підвищення ефективності машинного навчання є гібридизація і комплексування різних підходів і протоколів оптимізації

параметрів моделі аналізу даних. Активно досліджуються альтернативні функції втрат, методи регуляризації та пошукової оптимізації. Основною метою є зниження трудомісткості навчання з одночасним підвищення точнісних характеристик за умов обмеженого обсягу розмічених навчальних даних.

В праці [18] було показано, що використання дискретного двійкового подання даних забезпечує регуляризуючі та метарегуляризуючі ефекти. Даний підхід є ефективний з точки зору боротьби з перенавчанням і трудомісткості побудови вирішувальних правил. Схожий підхід розроблено під час розв'язання задач багатокласової класифікації шляхом її заміни еквівалентною множиною двох-класових задач. Одним з найбільш ефективних методів зведення багатокласової класифікації до серії двохкласових є двійкове кодування міток класів кодами, що коригують помилки (Error Correcting Output Codes, ECOC) [7].

Під час використання кодів, що коригують помилки, номер класу записують у вигляді k -значного двійкового числа. Для цього здійснюють навчання k -класифікаторів, кожен із яких розпізнає один із k розрядів номера класу. За результатами розпізнавання вхідного вектора кожним із класифікаторів однозначно відновлюють номер класу, до якого він належить. Якщо окремі класифікатори помиляються, то номер класу відновлюють методом заміни одержаного номера номером класу, що найближчий до одержаного за метрикою Хеммінга (рис. 1.8).

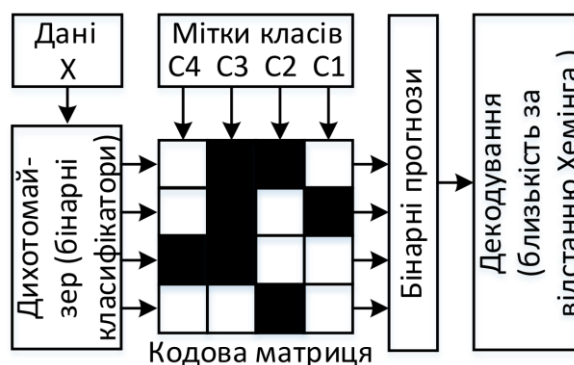


Рисунок 1.8 – Ілюстрація до схеми багатокласової класифікація з використанням двійкового кодування номера класу кодами, що виправляють помилки

На рис. 1.8 показано кодову матрицю, де чорні й білі комірочки позначають мітки 0 або 1 для зразків відповідних класів на етапі навчання бінарних

класифікаторів. У режимі екзамену цю матрицю використовують для порівняння прогнозованого коду з кодами класів. Ефективна кодова матриця повинна забезпечувати найбільшу відстань Хеммінга як між рядками кодової матриці, так і між стовбцями. Відстань між рядками забезпечує можливість самовиправлення помилок, а відстань між стовбцями забезпечує некорельованість результатів кожного з бінарних класифікаторів, що навчаються. При цьому існує багато способів формування номера класу для кодової матриці, однак саме коди, що виправляють помилки, (самокорегувальні коди) забезпечують кодування з необхідними властивостями. Поширеним способом формування самокорегувальних кодів є використання матриць і кодів Адамара [6].

Основним недоліком методу ЕСОС є ігнорування структури класів під час побудови кодової матриці без можливості оптимізації коду кожного класу в процесі навчання. Крім того, відсутність кодового радіуса для кожного номера класу, що вказує на кратність помилок, які можуть бути виправлені, ускладнює виявлення викидів або новизни в даних.

1.2 Методи оптимізації моделей класифікаційного аналізу спостережень

Використання ідей та методів теорії інформації та синтезу вирішальних правил в рамках геометричного підходу знижує вимоги до кількості розмічених даних для навчання і робить його більш робастним. Сіамські нейронні мережі є прикладом такого підходу [12]. Сіамські нейронні мережі демонструють багатообіцяючі результати в умовах обмеженого обсягу навчальних даних і часто є основою для реалізації так званого навчання з першого погляду (few-shot learning). Коди, що виправляють помилки (error-correcting output codes), та інформаційно-екстремальні методи навчання поєднують принципи теорії інформації та геометричного підходу [6]. Ці алгоритми найбільш корисні під час синтезу вирішальних правил з оптимальним словником ознак або для підвищення ефективності існуючих моделей. Таким чином, найбільш перспективним шляхом

подальшого вдосконалення моделей аналізу візуальних даних є поєднання ідей та методів сіамських нейронних мереж та інформаційно-екстремального навчання.

Запропонований метод складається з 5 основних етапів (фаз) (рис. 1.9). Перший етап включає в себе різноманітні операції з даними, такі як повороти, шкалювання, зміну яскравості і додавання шуму. Наступні 3 етапи – це безпосередньо навчання класифікатора. На останньому етапі вирішальні правила коригуються враховуючи дисперсію спостережень всередині класів в двійковому просторі Хеммінга [1, 2].

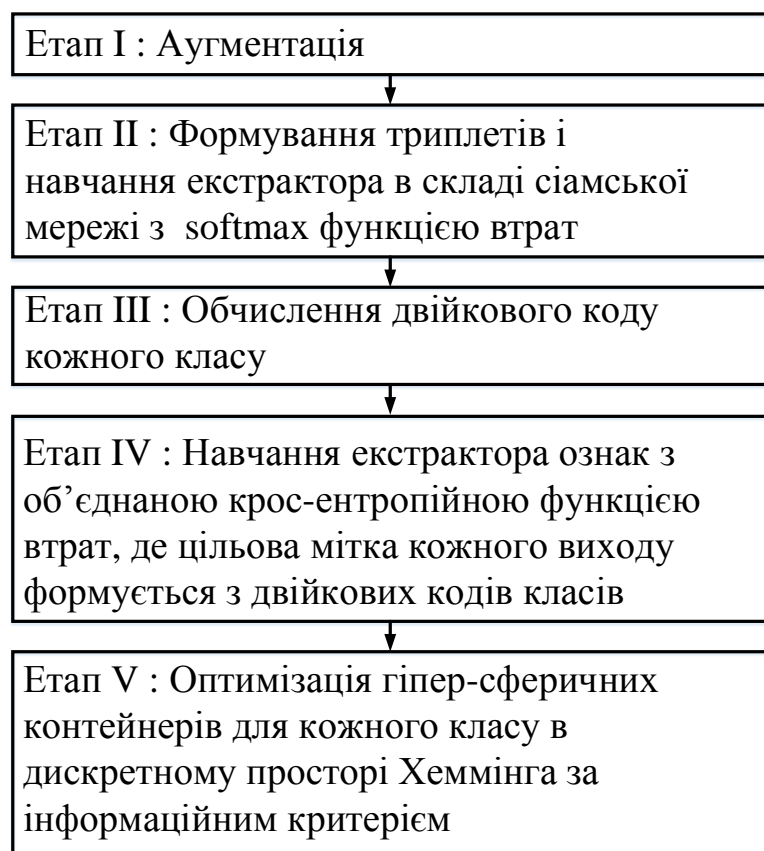


Рисунок 1.9 – Етапи запропонованого методу навчання

Другий етап навчання моделі аналізу зображення проводиться із застосуванням адаптивних алгоритмів зворотного поширення помилки, з яких найбільш популярним є Adam [17]. На вхід моделі подається міні-пакет, в якому M зображень кожного класу. Триплетна функція втрат розраховується за формулою

$$L = -\log \frac{\exp(\|f(x_a) - f(x_{ep})\|)}{\exp(\|f(x_a) - f(x_{ep})\|) + \exp(\|f(x_a) - f(x_{shn})\|)} \quad (1.2.1)$$

де $f(x)$ – функція, що описує екстрактор та встановлює залежність між вхідним зображенням та вектором виходу сигмоїдного шару; x_a – зображення, що обрано випадковим чином з міні-паketу; x_{ep} – найближчий сусід з міні-батчу, що належить до того ж самого класу, тобто

$$x_{ep} = \arg \min_{x:C(x)=C(x_a)} \|f(x_a) - f(x)\|, \quad (1.2.2)$$

де $C(x)$ – функція що повертає клас зображення; x_{shn} – зразок зображення з міні-паketу, що є найближчим серед зразків протилежних класів, проте знаходиться далі ніж зразок, тобто

$$x_{shn} = \arg \min_{\substack{C(x) \neq C(x_a) \\ \|f(x_a) - f(x)\| > \|f(x_a) - f(x_p)\|}} \|f(x_a) - f(x)\|. \quad (1.2.3)$$

Для підвищення ефективності апроксимації дискретного подання даних пропонується додати регуляризуючу складову, що враховуватиме помилку округлення вихідного сигналу мережі до двійкового коду. В результаті функція втрат набуде наступного вигляду

$$L = -\log \frac{\exp(\|f(x_a) - f(x_{ep})\|)}{\exp(\|f(x_a) - f(x_{ep})\|) + \exp(\|f(x_a) - f(x_{shn})\|)} + \lambda \left(f(x_a)^T (e - f(x_a)) + f(x_{ep})^T (e - f(x_{ep})) + f(x_{shn})^T (e - f(x_{shn})) \right), \quad (1.2.4)$$

де $e = [1, 1, \dots, 1]^T$ – одинична матриця-стовбець; λ – коефіцієнт регуляризації.

Наступна фаза потрібна для перетворення в бінарну форму вихідного вектора відповідно до принципів самокоректуючих кодів (error-correcting output

codes), однак з урахуванням внутрішньої структури класів та відношень між зразками різних класів. Для цього навчальна вибірка $\{x_{z,s} \mid z = \overline{1, Z}, s = \overline{1, n_z}\}$, що містить n_z зразків z -го класу кодується дискретним поданням $\{b_{z,s,i} \mid z = \overline{1, Z}, s = \overline{1, n_z}, i = \overline{1, N}\}$ з розмірністю N та обчислюється еталонний двійковий вектор кожного класу. Двійкове кодування здійснюється шляхом подавання зображень з множини $\{x_{z,s}\}$ на вхід моделі і округлення виходу сігмоїдного шару до цілого числа

$$b_{z,s,i} = \begin{cases} 1, & \text{if } f(x_{z,s,i}) > 0.5; \\ 0, & \text{otherwise.} \end{cases}, \quad (1.2.5)$$

Двійковий еталонний вектор b_z для z -го класу можна визначити шляхом порозрядного порівняння частоти двійкових одиниць в z -му класі з фоновою частотою одиниць в навчальній вибірці

$$b_{z,i} = \begin{cases} 1, & \text{if } \frac{1}{n_z} \sum_{s=1}^{n_z} b_{z,s,i} > \frac{1}{Z} \sum_{c=1}^Z \frac{1}{n_c} \sum_{s=1}^{n_c} b_{c,s,i}; \\ 0, & \text{otherwise.} \end{cases}, \quad (1.2.6)$$

Обчислений еталонний вектор z -го класу b_z можна використовувати як розмітку (label) зразка для подальшого навчання з використанням Joint Binary Cross Entropy Loss, яка для кожного вхідного зразка x може бути обчислена за формулою

$$L = -\sum_{i=1}^N (b_i \log f_i(x) + (1 - b_i) \log(1 - f_i(x))), \quad (1.2.7)$$

де $f_i(x)$ – значення i -го виходу сігмоїдного шару для вхідного зображення x ;

b_i – значення i -го розряду еталонного вектора класу, до якого належить зображення x .

Остання фаза машинного навчання пов'язана з оптимізацією радіусу контейнерів за інформаційним критерієм для врахування меж відхилення двійкового подання спостережень кожного класу від відповідних еталонних векторів

$$E_z^* = \max_{\{d\}} E_z(d), \quad (1.2.8)$$

де $\{d\} = \{0, 1, \dots, \left(\sum_i b_{z,i} \oplus b_{c,i} - 1 \right)\}$ – набір концентричних радіусів з центром b_z

розподілу даних в z -му класі, які обчислюються за правилом (1.2.6);

E_z – інформаційний критерій для z -го класу, який є функцією від характеристик точності [2].

Формування ознакового опису спостережень за умов ресурсних та інформаційних обмежень також варто здійснювати на основі принципів, що дозволяють утилізувати всю доступну інформацію як із розміченої й нерозміченої навчальних вибірок, так і з зовнішніх джерел. Вибіркові дані і без розмітки містять багато інформації про їх структуру, яку можна виділити на основі моделей розділення пояснювальних факторів (disentangle explanatory factors). Моделі розділення пояснювальних факторів зазвичай будують на основі автокодувальників, машин Больцмана, алгоритмів розрідженого кодування та генеративних моделей. Серед моделей розділення пояснювальних факторів заслуговують на увагу ті, що ґрунтуються на ідеях і методах розрідженого кодування. Покладений у їх основу ефект редукції причини (explaining away) дозволяє виявити приховані фактори (першопричини) й забезпечити інформативне та завадозахищене ознакове подання вибірових спостережень навіть за умов обмеженого обсягу даних.

Основним недоліком розрідженого кодування в навчанні ознакового

подання є використання ітеративної процедури під час екзамену, що уповільнює процес розпізнавання. Один із популярних способів прискорити моделювання – це використовувати принципи дистиляції знань, де надлишкова модель, яка виступає вчителем, може бути замінена обчислювально ефективнішою моделлю, що відіграє роль студента [3]. Ансамбль дерев рішень – це гнучка та обчислювально ефективна модель, яка потенційно може бути використана як модель студента для спрощення розрідженого кодера [4].

Для підвищення рівня абстрактності пояснювальних факторів екстрактор ознак може містити декілька шарів розрідженого кодування. Реалізація кодера може бути виконана на основі алгоритмів узгодженого переслідування (matching pursuit), ортогонального узгодженого переслідування (orthogonal matching pursuit), мішка переслідувачів (bag of pursuits) чи інших. Для прискорення моделі в режимі екзамену згідно принципів дистиляції знань (knowledge distillation) обчислювально складний етап оптимізації для пошуку розріджених коефіцієнтів можна замінити апроксимувальним кодером [3, 6]. Навчальна вибірка апроксимувального кодера формуватиметься як із вхідних даних, що кодуються, так і з результувального розрідженого коду. На рис. 1.10 показано схему синтезу екстрактора ознак із використанням ідей і методів розрідженого кодування.

Для розрідженого кодування використовують навчений словник атомів. Атоми мають розмірність таку саму як і будь-яке вхідне спостереження x і їх у найпростішому випадку, можна сформувати методом кластер-аналізу чи векторного квантування спостережень. У більш складному випадку навчання можуть здійснювати за схемою з кінця-в-кінець і навіть із частковим залученням учителя. Для забезпечення саморегуляції інформаційної ємності моделі атоми розрідженого кодера можуть бути знайдені за алгоритмом зростаючого розріджено кодуючого нейронного газу.

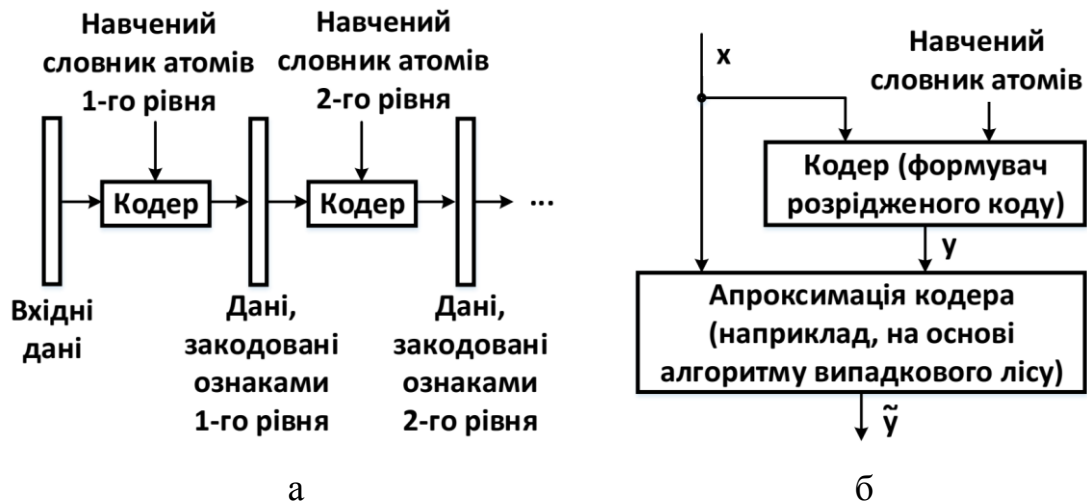


Рисунок 1.10 – Схема побудови екстрактора ознак на основі розрідженого кодування: а – послідовність оброблення даних у багатошаровому екстракторі; б – схема дистиляції знань з розрідженого кодера

У разі зображення процес його розрідженого кодування полягає в декомпозиції на патчі, що перетинаються, кожен із яких кодується кодером у багатоканальний піксель карти ознак (рис. 1.11).

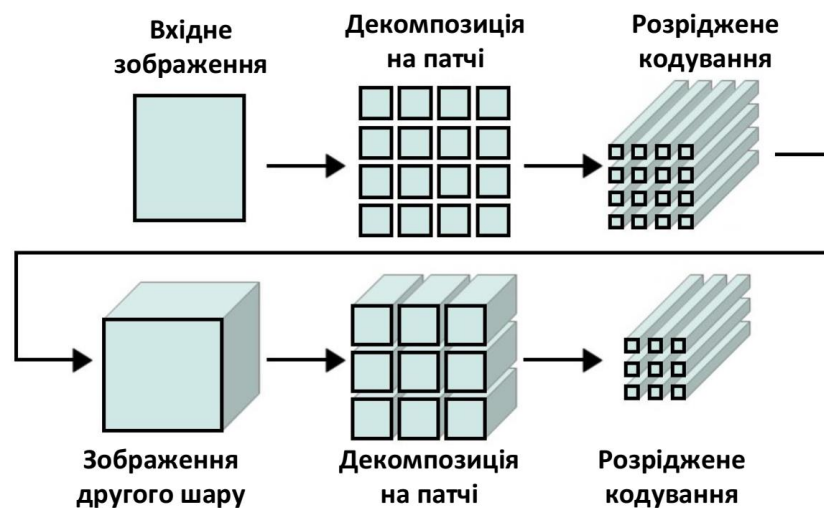


Рисунок 1.11 – Схема аналізу зображення двошаровою моделлю на основі розрідженого кодування

Оскільки кодер оснований на розрідженому кодуванні, то піксель карти ознак матиме розріджену активацію каналів. Одержану карту ознак можна сприймати як нове багатоканальне зображення, для якого можна повторити процедуру розрідженого кодування. При цьому кількість атомів (кластерів)

кодера може бути як більшою за розмірність вхідного патчу, що відповідає надповному базису (overcomplete dictionary), так і меншою за розмірність вхідного патчу, що відповідає неповному базису (undercomplete dictionary). Дослідження показують, що для задачі класифікаційного аналізу можуть бути ефективними як надповні, так і неповні базиси екстрактора ознак [6].

Для аналізу даних високої розмірності з відомою топологією можна використовувати діряві (dilated) рецептивні поля, що застосовують до локальних ділянок даних, розмір яких перевищує кількість входів кодера. Тобто частина даних, що накривається рецептивним полем, ігнорують, а аналізу піддають дані з певним просторовим або часовим кроком (рис. 1.12).

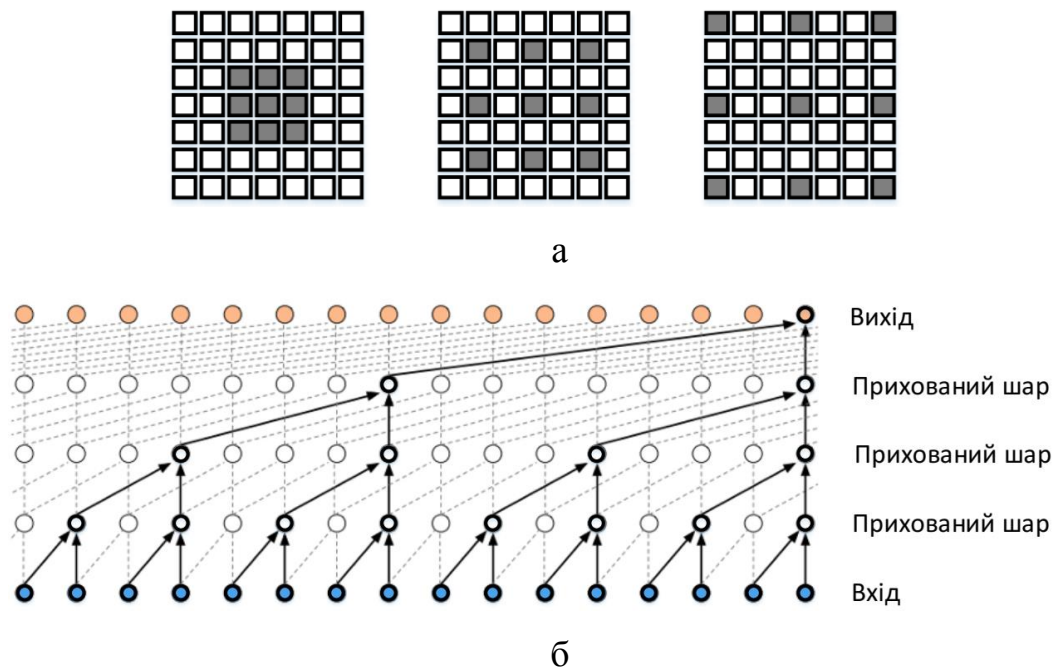


Рисунок 1.12 – Діряві рецептивні поля моделі аналізу даних:

а – 2D-топология; б – 1D-топология каузальної моделі

Стекування кодерів із дірявими рецептивними полями дозволяє досягти великого рецептивного поля моделі з використанням невеликої кількості шарів і параметрів. Разом із тим не відбувається втрат інформації, оскільки кодер здійснює сканування вхідних даних у просторі або часі. Крім того, наявність пропусків дозволяє знизити чутливість до високочастотних складових шуму і, в наслідок цього, підвищити завадозахищеність моделі.

Для дослідження залежностей на різних рівнях деталізації в межах окремого шару моделі використовують поєднання кодерів з рецептивними полями різного розміру. Малі рецептивні поля кодерів дозволяють здійснювати екстракцію детальної (fine-grained) інформації, а великі рецептивні поля кодерів забезпечують екстракцію грубої (coarse-grained), переважно, контекстної інформації. Наприклад у популярних Inception-модулях згорткових мереж використовують згорткові фільтри з ядрами 1x1, 3x3 та 5x5 для сприйняття різномасштабної просторової інформації (від детального до загубленого рівнів).

Таким чином, для побудови інформативного ознакового опису і завадозахищених вирішувальних правил варто будувати модель шляхом поєднання ідей та методів сіамських мереж та інформаційно-екстремального навчання. У випадку змінної інформативності ознакового опису синтез екстрактора ознак варто здійснювати в рамках методів розрідженого кодування для розділення пояснюючих факторів. При цьому зниження обчислювальної складності в режимі екзамену можна забезпечити шляхом застосування алгоритмів дистиляції знань.

1.3 Ентропійний інформаційний критерій ефективності вирішувальних правил бортової системи класифікаційного аналізу зображень

Важливим питанням інформаційного синтезу моделей аналізу даних є оцінка функціональної ефективності процесу навчання, яка визначає максимальну асимптотичну достовірність рішень, що приймаються на екзамені. Як критерій функціональної ефективності можуть використовуватися різні інформаційні критерії, які задовольняють таким властивостям інформаційних мір:

- інформаційна міра є величина дійсна і знакододатна як функція від імовірності;
- кількість інформації для детермінованих змінних ($p_i = 1$ або $p_i = 0$) дорівнює нулю;

- інформаційна міра має екстремум при значенні ймовірності $p_i = \frac{1}{m}$, де m

– кількість якісних ознак розпізнавання.

Серед інформаційних мір для оцінки функціональної ефективності моделі аналізу даних перевагу слід віддавати статистичним логарифмічним критеріям, які дозволяють працювати з навчальними вибірками відносно малих обсягів [6]. Серед таких критеріїв найбільшого використання знайшла ентропійна міра. Подамо нормований ентропійний критерій ефективності навчання моделі розпізнавати спостереження класу X_m^o у вигляді :

$$E_m^{(k)} = \frac{I_m^{(k)}}{I_{\max}^{(k)}} = \frac{H_m^{(k)} - H_m^{(k)}(\gamma)}{H_m^{(k)}}, \quad (1.3.1)$$

де $I_m^{(k)}$ – кількість умовної інформації, що обробляється на k -му кроці навчання моделі розпізнавати спостереження класу X_m^o ;

$I_{\max}^{(k)}$ – максимально можлива кількість умовної інформації, одержаної на k -му кроці навчання моделі розпізнавати спостереження одного із класів із заданого алфавіту $\{X_m^o\}$, $m = \overline{1, M}$;

$$H_m^{(k)} = -\sum_{l=1}^M p(\gamma_{l,k}) \log_2 p(\gamma_{l,k}) \quad (1.3.2)$$

апостеріорна (безумовна) ентропія, що існує на k -му кроці навчання моделі розпізнавати реалізації класу X_m^o ;

$$H_m^{(k)}(\gamma) = -\sum_{l=1}^M \sum_{m=1}^M p(\gamma_{l,k}) p(\mu_{m,k} / \gamma_{l,k}) \log_2 p(\mu_{m,k} / \gamma_{l,k}) - \quad (1.3.3)$$

апостеріорна (умовна) ентропія, що характеризує залишкову невизначеність після k -го кроку навчання системи розпізнавати реалізації класу X_m^o ;

$p(\gamma_{l,k})$ – безумовна ймовірність прийняття на k -му кроці навчання гіпотези $\gamma_{l,k}$;

$p(\mu_{m,k} / \gamma_{l,k})$ – апостеріорна ймовірність прийняття на k -му кроці навчання рішення $\mu_{m,k}$ за умови, що прийнята гіпотеза $\gamma_{l,k}$.

Для двохальтернативної системи оцінок ($M = 2$) і рівноймовірних гіпотез, що характеризує найбільш важкий у статистичному сенсі випадок прийняття рішень, після відповідної підстановки ентропій (1.3.2) і (1.3.3) у вираз (1.3.1) та заміни відповідних апостеріорних ймовірностей на апіорні за формулою Байєса ентропійний критерій набуває вигляду

$$E_m^{(k)} = 1 + \frac{1}{2} \left(\frac{\alpha_m^{(k)}(d)}{\alpha_m^{(k)}(d) + D_{2,m}^{(k)}(d)} \log_2 \frac{\alpha_m^{(k)}(d)}{\alpha_m^{(k)}(d) + D_{2,m}^{(k)}(d)} + \frac{\beta_m^{(k)}(d)}{D_{1,m}^{(k)}(d) + \beta_m^{(k)}(d)} \log_2 \frac{\beta_m^{(k)}(d)}{D_{1,m}^{(k)}(d) + \beta_m^{(k)}(d)} + \frac{D_{1,m}(d)}{D_{1,m}^{(k)}(d) + \beta_m^{(k)}(d)} \log_2 \frac{D_{1,m}(d)}{D_{1,m}^{(k)}(d) + \beta_m^{(k)}(d)} + \frac{D_{2,m}^{(k)}(d)}{\alpha_m^{(k)}(d) + D_{2,m}^{(k)}(d)} \log_2 \frac{D_{2,m}^{(k)}(d)}{\alpha_m^{(k)}(d) + D_{2,m}^{(k)}(d)} \right), \quad (1.3.4)$$

де $\alpha_m^{(k)}(d)$ – частота помилок першого роду (false positive rate) – точнісна характеристика рішення на k -му кроці навчання;

$\beta_m^{(k)}(d)$ – частота помилок другого роду (false negative rate);

$D_{1,m}^{(k)}(d)$ – чутливість (true positive rate, sensitivity) або перша достовірність;

$D_{2,m}^{(k)}(d)$ – специфічність (true negative rate, specificity) або друга достовірність;

d – дистанційна міра, яка визначає радіуси гіперсферичних контейнерів, побудованих в радіальному базисі простору Хеммінга.

Оскільки точнісні характеристики є функціями відстані від еталонних від геометричних центрів контейнерів (кодів) відповідних класів розпізнавання, то критерій (1.3.4) слід розглядати як нелінійний і взаємно-неоднозначний

функціонал від точністних характеристик, що потребує знаходження в процесі навчання робочої (допустимої) області для його визначення. Робочу область визначення функції інформаційного критерію можна задати у вигляді системи нерівностей

$$\begin{cases} \beta_m^{(k)}(d) < 0,5; \\ \alpha_m^{(k)}(d) < 0,5; \\ d < d(x_c \oplus x_m), \end{cases}$$

де $d(x_c \oplus x_m)$ – відстань Хемінга між центрами сусідніх класів.

Таким чином, двійкові еталонні вектори класів використовуються по аналогії з самокоректуючими кодами Хемінга, а радіус контейнера характеризує максимальну кількість помилок, які можуть бути виправлені після отримання повідомлення по каналу зв'язку з завадами. Інформаційний критерій характеризує міру зняття невизначеності після отримання повідомлення з заданого алфавіту. Оптимізація радіусу гіперсферичних контейнерів дозволяє підвищити ефективність прийняття рішень за умов перетину класів з різними габаритами розподілу спостережень та виявляти новизну в даних у випадку появи спостережень, що не потрапляють в контейнер жодного класу. При цьому критерієм оптимізації є логарифмічний інформаційний критерій, що підвищує узагальнюючу здатність моделі та забезпечує її робастність за найскладніших у статистичному розумінні випадках.

2 РЕАЛІЗАЦІЯ БОРТОВОЇ СИСТЕМИ БЕЗПІЛОТНОГО АПАРАТУ ДЛЯ АНАЛІЗУ ВІДЕОСПОСТЕРЕЖЕНЬ

2.1 Апаратне забезпечення безпілотної літального апарату

Тестовою платформою для розроблених алгоритмів є БПЛА квадрокоптерного типу. Бортову систему БПЛА складають бортовий комп'ютер та контролер польоту. Бортовий комп'ютер в свою чергу може складатися з декількох одноплатних комп'ютерів чи допоміжних обчислювальних модулів. Функціональна схема бортової системи безпілотної літального апарату показано на рис. 2.1.

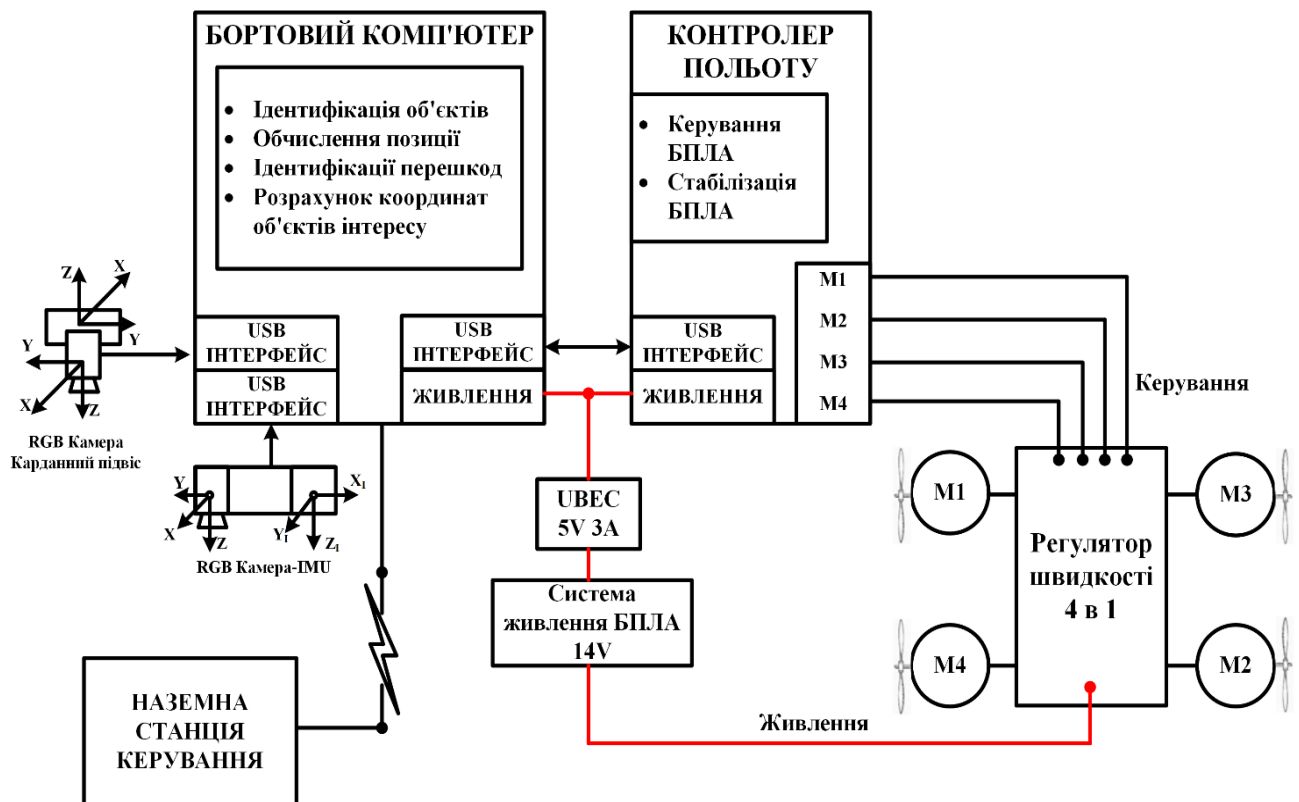


Рисунок 2.1 – Функціональна схема БПЛА для аеровідеомоніторингу місцевості

Відповідно до схеми наведеної на рис. 2.1 функції зовнішньої комунікації, формування команд для контролера польоту, автономної навігації та додаткові функції аналізу даних бере на себе модуль бортового комп'ютера. Отримані за допомогою бортових камер візуальні знімки та дані від інерційних вимірювальних пристроїв подаються до бортового комп'ютера через послідовний інтерфейс де, після їх попередньої обробки, виконується аналіз даних: розрахунок

поточного положення апарату, ідентифікація об'єктів інтересу на вхідних зображеннях та прогнозування можливих перешкод [16]. Після проведення аналізу зображень та знаходження об'єктів інтересу виконуються розрахунок координат знайдених об'єктів з використанням інформації про поточне положення БПЛА, що отримане від модуля обчислення позиції. Після цього результат роботи може бути переданий через модуль зв'язку до наземної станції, або збережений в бортовій пам'яті БПЛА. Інформація про поточні координати і орієнтацію камери БПЛА та можливі перешкоди на шляху руху надходять до модуля коригування траєкторії польоту. Цей модуль виконує відповідний розрахунок та формує завдання для польотного контролера на корекцію руху БПЛА та інформує користувача про можливу зміну траєкторії руху БПЛА за необхідністю. Контролер польоту здійснює керування двигунами БПЛА відповідно до траєкторії руху отриманої від бортового комп'ютера. Одночасно з цим контролер польоту здійснює неперервну стабілізацію апарату для компенсації поривів вітру і інших зовнішніх впливів. Наземна станція керування виконує функції модуля візуалізації інформації та людино-машинної взаємодії з БПЛА на відстані по бездротовому каналу зв'язку. При цьому наземна станція керування дозволяє перемикати чи активувати режими функціонування інтелектуальних алгоритмів бортової системи : режим корекції алгоритму геометричної візуальної навігації; режим розпізнавання об'єктів на місцевості; режим збору розмічених навчальних зразків; режим донавчання/перенавчання.

Одним з найважливіших компонентів квадрокоптера є контролер польоту, що здійснює неперервну стабілізацію апарату, протидіючи поривам вітру і неоднорідностям повітряних мас. Завдяки контролеру польоту, правильно зібраний і мінімально налаштований БПЛА здатний зависати у повітрі практично без участі пілота. Як контролер польоту випробовувалися такі модулі як Navio 2, ArduPilot Mega або Pixhawk.

Бортовий комп'ютер бере на себе функції зовнішньої комунікації, формування команд для контролера польоту та інтелектуальний аналіз даних від бортової камери БПЛА. Для побудови бортового комп'ютера було використано

одноплатний комп'ютер Raspberry Pi 3B+ на базі ARM процесора. Даний комп'ютер має 64-бітний процесор CPU Broadcom BCM2837B0, Cortex-A53 (ARMv8) з частотою 1.4 ГГц і обсяг оперативної пам'яті рівні 1 Гбайт. Офіційна операційна система Raspbian є 32-х бітною, а неофіційні 64-х бітні операційні системи мають численні проблеми з драйверами вводу-виводу. Тому була виконана компіляція Debian Stretch для 64-х бітного процесора SoC Broadcom BCM2837B0 Cortex-A53. Крім того для реалізації інтелектуальних функцій бортовий комп'ютер може бути укомплектований додатковим Raspberry Pi 3B+ та енергоефективним USB модулем нейрообчислень на основі Movidius Neural Compute Stick V2 від компанії Intel. Основою модуля Movidius є процесор машинного зору (англ. VPU) Myriad X з інтегрованим нейропроцесором (англ. NPU). Продуктивність Movidius V2 може досягати 4 Терафлопси.

Компоновочна схема квадрокоптера, що побудованого на базі Raspberry Pi 3B+, та Navio 2 показано на рис. 2.2

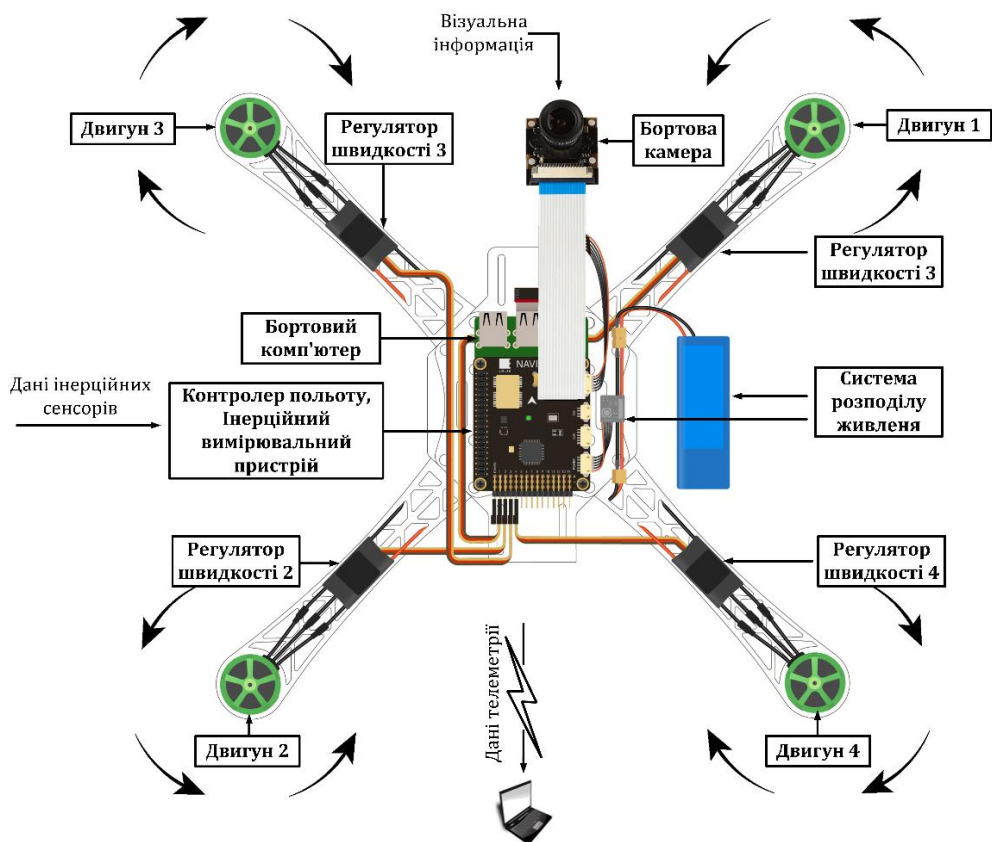
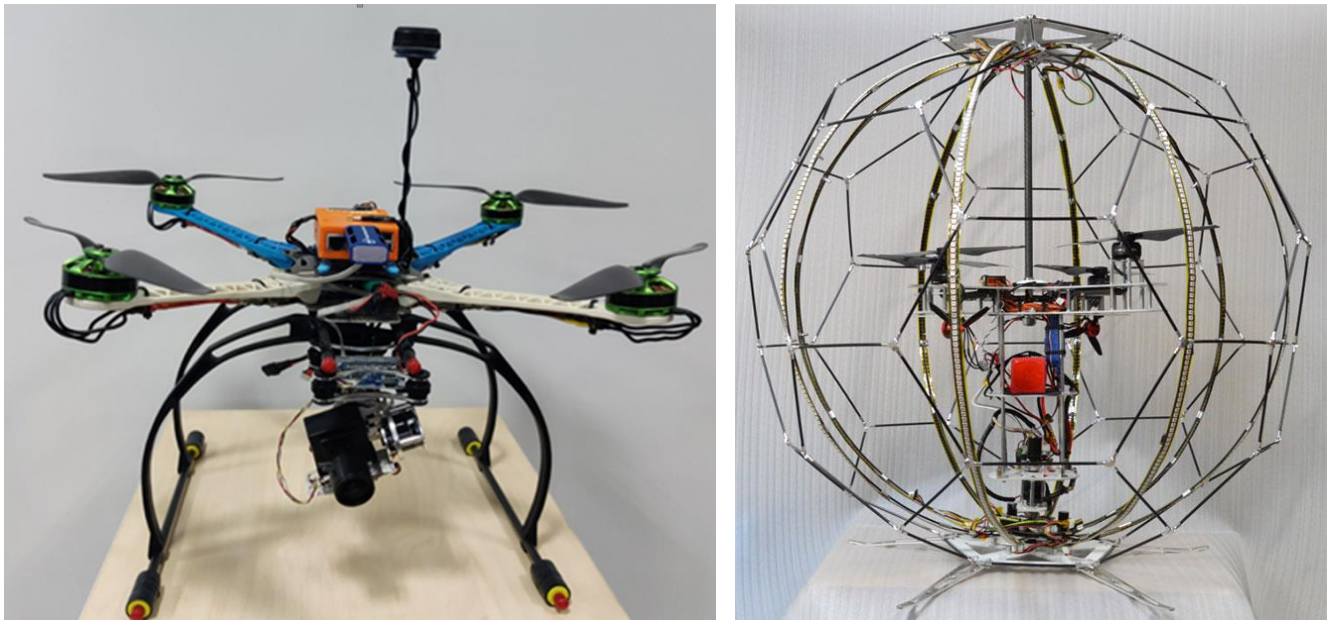


Рисунок 2.2 – Структурна схема апаратного забезпечення БПЛА

На рис. 2.3а показано розроблена апаратна частина БПЛА для тестування і відладки інтелектуальної інформаційної технології відеомоніторингу місцевості. При цьому на рис. 2.3б показано популярну конструкцію захисного каркасу БПЛА для реалізації задач інспекції промислових та інфраструктурних об'єктів.



а

б

Рисунок 2.3 – Дослідний зразок для тестування інформаційної технології :

а – базова комплектація БПЛА; б – сферичний захисний кожух для задач інспекції промислових та інфраструктурних об'єктів

В задачах визначення поточного положення БПЛА та координат об'єктів інтересу важливим є отримання внутрішніх параметрів камери (значень оптичного центру, фокусної відстані, коефіцієнтів дисторсії) [16]. Для визначення внутрішніх параметрів необхідно виконати процедуру калібрування камери Існує велика кількість програмних продуктів призначених для калібрування камери. Калібрування камери здійснюється за допомогою програмного функції `cv::calibrateCamera` бібліотеки машинного зору `opencv` [16]. Для калібрування камери використано 20 кадрів з зображенням каліброчної матриці формату A2, сформованих з різного фіксованого положення камери (рис. 2.4). Клітинка каліброчного шаблону має розмір 25 x 25 мм. В табл. 2.1 наведено результати обчислення каліброчних параметрів бортової камери БПЛА.

Аналіз табл. 2.4 показує, що значення фокусної відстані подано в пікселях по горизонталі і вертикалі. Сенсор даної камери має майже квадратні пікселі. Проте оптичний центр лінзи не співпадає із центром зображення, що призводить до похибок під час оцінювання лінії горизонту і переміщення.

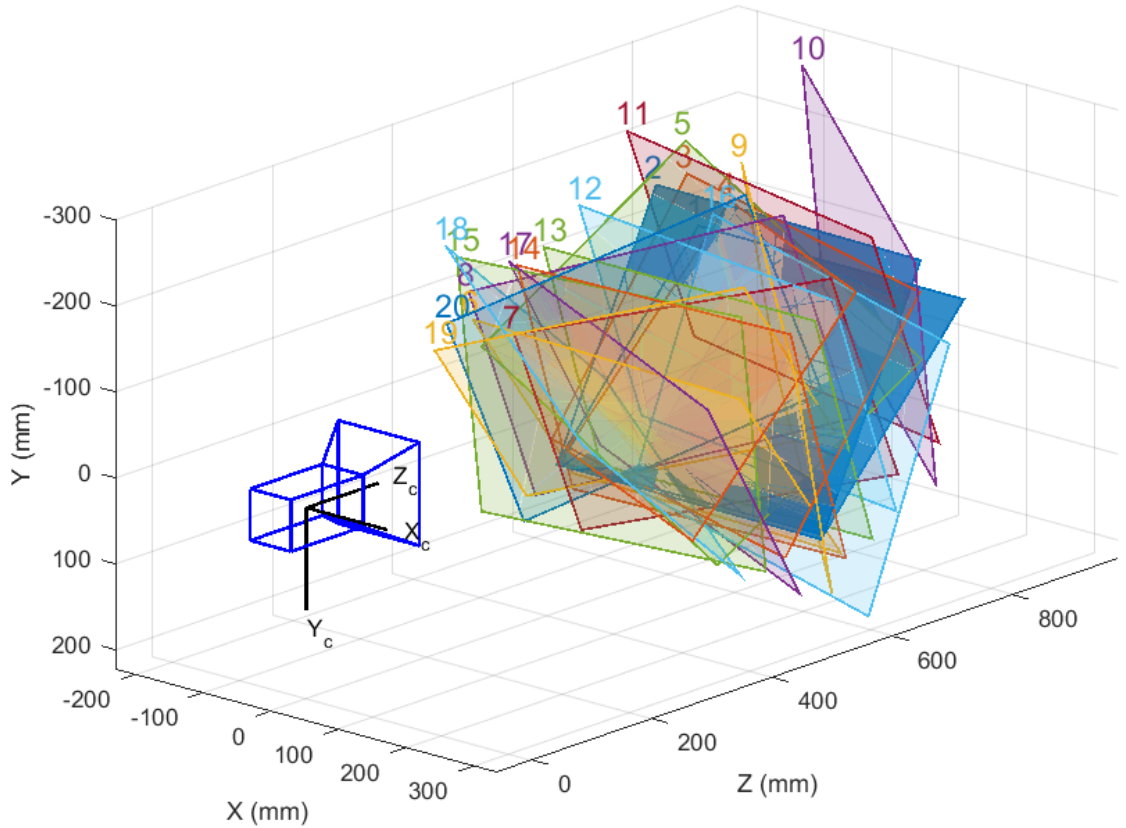


Рисунок 2.4 – Положення візуального шаблону в системі координат камери, що підлягає калібруванню

Таблиця 2.1. Внутрішні параметри калібрування камери Arducam CMOS MT9V034

Параметр	Значення параметру
Фокусна відстань f , пікс	$[461.315, 460.692] \pm [0.176, 0.176]$
Проекція оптичного центру, пікс	$[377.969, 268.412] \pm [0.300, 0.298]$
Піксельна похибка, пікс	$[0.1756, 0.1358]$
Радіальні коефіцієнти, пікс	$[-0.1036, -0.0161] \pm [0.0044, 0.0169]$
Тангенсійні коефіцієнти, пікс	$[0.0139, -0.0037] \pm [0.0243, 0.0117]$

Для забезпечення оптимального використання бортових ресурсів всі обчислювально складні алгоритми було реалізовано з використанням прискорення векторних операцій. Такі операції полягають у об'єднанні декількох однакових операцій до однієї. При цьому набір даних, до якого застосовується така операція, є вектором. Дані операції підтримують роботу з наступними типами даних: integer 8, 16, 32, 64 bit, float 32 bit.

Вимірювання координат об'єкта на місцевості з використанням бортової відеокамери БПЛА потребує узгодження і встановлення відповідності між різними системами координат: локальна навігаційна система координат Північ-Схід-Вниз (англ. North-East-Down, NED); система координат транспортного засобу; система координат корпусу безпілотної апарату; система координат карданного підвісу камери; система координат відеокамери [16], [7]. На рис. 2.5 та рис. 2.6 показано вигляд БПЛА і пов'язані з ним системи координат.

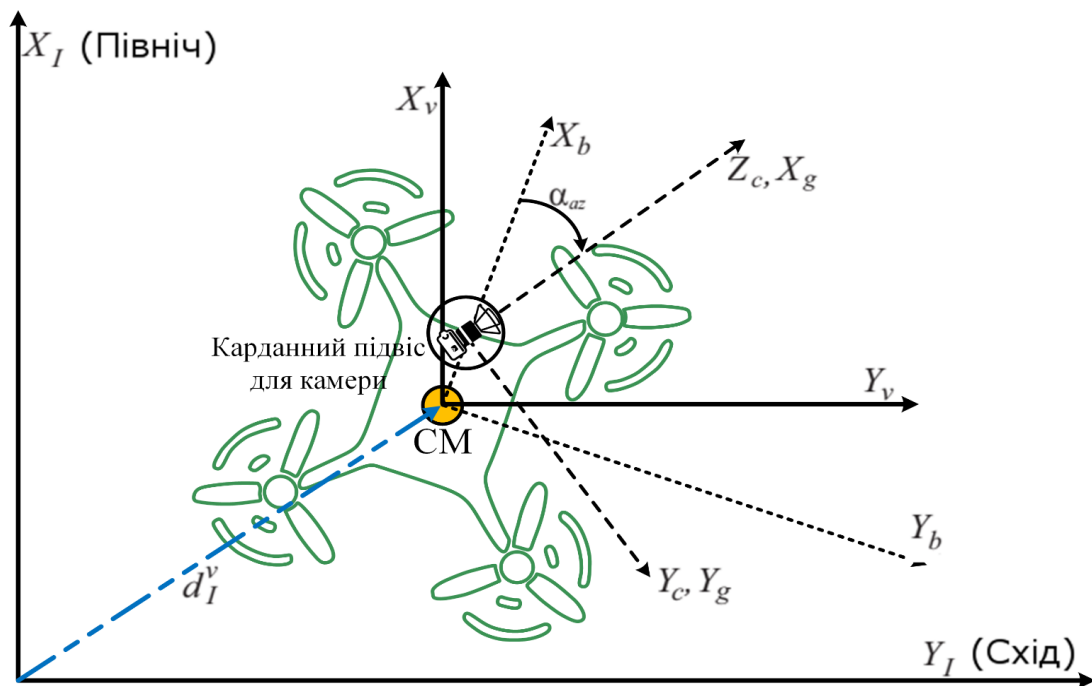


Рисунок 2.5 – Вид згори на БПЛА та пов'язані системи координат

Як видно з рис. 2.5 навігаційна система координат (X_I, Y_I, Z_I), яку ще називають інерційною, є фіксованою, де вісь X_I спрямована на північ, вісь Y_I спрямована на схід, вісь Z_I – до центру Землі. Система координат транспортного засобу (X_v, Y_v, Z_v) прив'язана до центру його мас, а його координатні вісі мають той же напрямок, що й відповідні вісі локальної навігаційної системи координат.

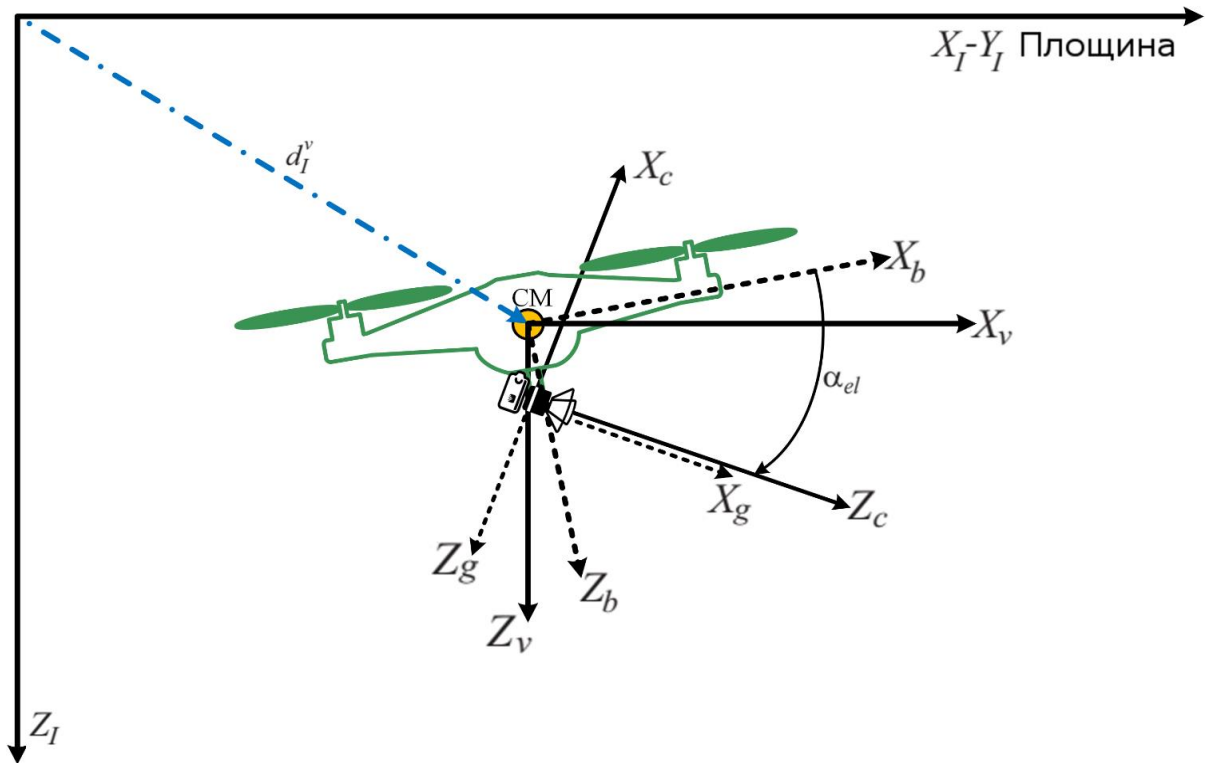


Рисунок 2.6 – Вид збоку на БПЛА та пов’язані системи координат

Система координат корпусу безпілотного апарату (X_b, Y_b, Z_b) також проходить через центр його мас, але зафіксована на корпусі, тому вісь X_b орієнтована вздовж передньої носової частини апарату, вісь Y_b перпендикулярна до X_b і орієнтована вздовж площини корпусу в напрямку правого крила, а вісь Z_b спрямована вниз, але перпендикулярно до осей X_b та Y_b (рис. 2.5).

Система координат підвісу камери (X_g, Y_g, Z_g) проходить через центр обертання підвісу камери. При цьому вісь X_g орієнтована вздовж оптичної осі камери, вісь Y_g спрямована вниз вздовж площини зображення камери, а Z_g орієнтована вправо вздовж площини зображення. Система координат камери (X_c, Y_c, Z_c) починається в оптичному центрі камери, де вісь X_c спрямована вперед перпендикулярно до площини зображення, вісь Y_c спрямована вправо вздовж площини зображення, а вісь Z_c орієнтована вниз вздовж площини камери.

Трансформація від системи координат корпусу БПЛА до системи координат карданного підвісу камери T_b^g залежить від розташування центра мас літального

апарату по відношенню до центру обертання підвісу камери. Дане зміщення задане вектором d_b^g . Для експериментального зразка дане зміщення було обчислене і становить $d_b^g = [80, 63, 45]$. Також T_b^g залежить від повороту системи координат підвісу відносно системи координат корпусу безпілотно апарату. Цей поворот задається матрицею обертання R_b^g , обчислення якої потребує вимірювання кута азимуту камери (обертання навколо Z_g) та кута її нахилу (обертання навколо Y_g). Нехай α_{az} позначає азимутальний кут повороту навколо Z_g , а α_{el} – кут нахилу під час повороту навколо Y_g після повороту на α_{az} . Значення кутів α_{el} та α_{az} можна отримати з контролера сервоприводів карданного підвісу. Таким чином T_b^g може бути обчислена за формулою

$$T_b^g = \begin{pmatrix} R_b^g & -d_b^g \\ 0 & 1 \end{pmatrix},$$

де

$$R_b^g = R_{y, \alpha_{el}} R_{z, \alpha_{az}} = \begin{bmatrix} c_{el} & 0 & s_{el} \\ 0 & 1 & 0 \\ -s_{el} & 0 & c_{el} \end{bmatrix} \begin{bmatrix} c_{az} & s_{az} & 0 \\ -s_{az} & c_{az} & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} c_{el}c_{az} & c_{el}s_{az} & s_{el} \\ -s_{el} & c_{az} & 0 \\ -s_{el}c_{az} & -s_{el}s_{az} & c_{el} \end{bmatrix}.$$

де $c_\varphi = \cos \varphi$ та $s_\varphi = \sin \varphi$.

Трансформація від інерційної системи координат до системи координат транспортного засобу T_I^v є простим переміщенням і визначається за формулою

$$T_I^v = \begin{pmatrix} I & -d_I^v \\ 0 & 1 \end{pmatrix},$$

де

$$d_I^v = \begin{pmatrix} x_{UAV} \\ y_{UAV} \\ -h_{UAV} \end{pmatrix}, \quad (2.1.1)$$

де x_{UAV} , y_{UAV} – координати розташування за осями Північ-Схід, що вимірюються з використанням GPS чи візуальної одометрії;

h_{UAV} – виміряна висота, на якій знаходиться БПЛА.

Трансформація від системи координат транспортного засобу до системи координат корпусу безпілотного апарату T_v^b складається з вимірювань поворотів. Нехай ϕ , θ та ψ є крен (англ. roll), тангаж (англ. pitch) і рискання (англ. yaw) в радіанах, тоді T_v^b визначається за формулою

$$T_v^b = \begin{pmatrix} R_v^b & 0 \\ 0 & 1 \end{pmatrix},$$

де

$$R_v^b = \begin{pmatrix} c_\theta c_\psi & c_\theta s_\psi & -s_\theta \\ s_\phi s_\theta c_\psi - c_\phi s_\psi & s_\phi s_\theta s_\psi + c_\phi c_\psi & s_\phi c_\theta \\ c_\phi s_\theta c_\psi + s_\phi s_\psi & c_\phi s_\theta s_\psi - s_\phi c_\psi & c_\phi c_\theta \end{pmatrix},$$

Кути Ейлера дають ϕ , θ та ψ на борту малогабаритного БПЛА можуть бути виміряні за даними гіроскопів і акселерометра (MEMS-технологія), використовуючи двоетапний фільтр Калмана [16].

Трансформація системи координат карданного підвісу камери до системи координат камери T_g^c залежить від вектора d_g^c , який описує розташування центру обертання підвісу відносно оптичного центру камери. T_g^c також залежить від фіксованого повороту R_g^c , що визначається орієнтацією закріпленої на підвісі камери таким чином, що $X_c = -Z_g$ та $Z_c = X_g$. Тому трансформація T_g^c може бути обчислена за формулою

$$T_g^c = \begin{pmatrix} R_g^c & -d_g^c \\ 0 & 1 \end{pmatrix},$$

де

$$R_g^c = \begin{pmatrix} 0 & 0 & -1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}, \quad (2.1.2)$$

На рис. 2.7 показано просту модель проекційної камери, де точка $q = (x_{ip}, y_{ip}, 1, 1)^T$ є гомогенною проекцією точки $p_{obj}^c = (p_x, p_y, p_z, 1)^T$ на площину зображення в пікселях. При цьому p_{obj}^c позначає розташування об'єкта p відносно центру камери.

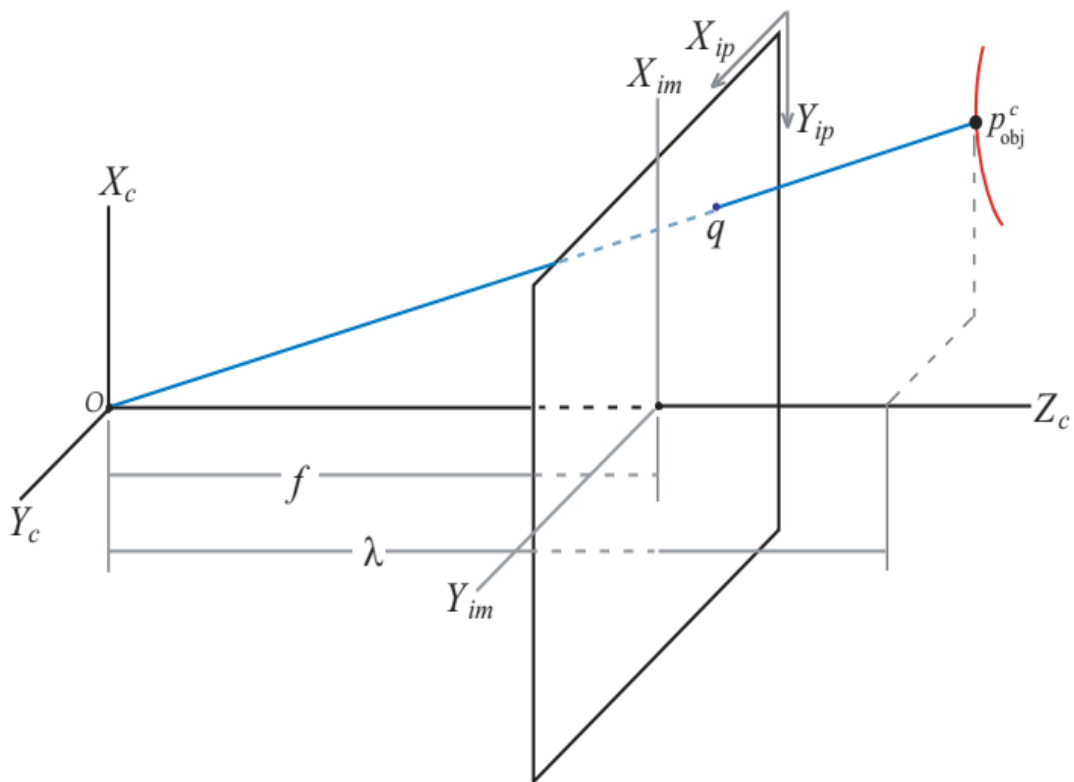


Рисунок 2.7 – Система координат, пов'язана з камерою

З камерою пов'язано три системи координат. Перша система задає координати (X_c, Y_c, Z_c) , що відраховуються від центру камери, де одиницею виміру є метр. Друга система задає координати $(X_{im}, Y_{im}, Z_{im} = Z_c - f)$, центр яких відраховується від площини зображення, де одиницею виміру є метр. Третя система координат задає координати (X_{ip}, Y_{ip}) , центр яких знаходиться в верхньому лівому куті зображення, де одиницею виміру є піксель. При цьому перехід від пікселів до метрів здійснюється за формулами

$$x_{im} = (-y_{ip} + 0_y)S_y, \quad y_{im} = (-x_{ip} + 0_x)S_x,$$

де (x_{ip}, y_{ip}) – координати в пікселях;

(x_{im}, y_{im}) – координати в метрах;

$0_x, 0_y$ – позначають зміщення x і y від верхнього лівого кута до центру зображення в пікселях;

S_x, S_y – позначають коефіцієнти перетворення від пікселів до метрів.

Відповідно до рис. 2.19 можна сформулювати такі відношення

$$\frac{x_{im}}{f} = \frac{p_x}{p_z}, \quad \frac{y_{im}}{f} = \frac{p_y}{p_z}.$$

Нехай $\lambda \square p_z$, тоді отримаємо рівність

$$\Lambda q = \underbrace{\begin{bmatrix} 0 & f_x & 0_x & 0 \\ -f_y & 0 & 0_y & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}}_C p_{obj}^c,$$

де $f_x \square \frac{f}{S_x}$, $f_y \square \frac{f}{S_y}$ та $\Lambda = \begin{pmatrix} \lambda I & 0 \\ 0 & 1 \end{pmatrix}$;

C – матриця калібрування.

Кінцевою метою є визначення координат p_{obj}^I позицій об'єктів в інерціальній системі (X_I, Y_I, Z_I) , тому, використовуючи розглянуті гомогенні перетворення, отримаємо такі рівняння :

$$\Lambda q = C p_{obj}^c = C T_g^c T_b^g T_v^b T_I^v p_{obj}^I, \quad p_{obj}^I = \left[C T_g^c T_b^g T_v^b T_I^v \right]^{-1} \Lambda q, \quad (2.1.3)$$

Як видно з формули (2.1.3) координати об'єкта p_{obj}^I можна визначити, якщо

відома змінна λ , яку ще називають глибиною зображення. Глибина зображення λ – це відстань вздовж оптичної осі камери до об'єкта інтересу. Розглянемо процес оцінки глибини λ .

Нехай p_{cc} є оптичним центром камери. Якщо p_{cc} визначено в системі координат камери, то $p_{cc}^c = (0, 0, 0, 1)^T$, а p_{cc}^I визначається за формулою

$$p_{cc}^I = \begin{pmatrix} x_{cc}^I \\ y_{cc}^I \\ z_{cc}^I \\ 1 \end{pmatrix} = [T_g^c T_b^g T_v^b T_I^v]^{-1} \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}.$$

На рис. 2.8 також показано розташування $q = [x_{ip} \quad y_{ip} \quad 1 \quad 1]^T$.

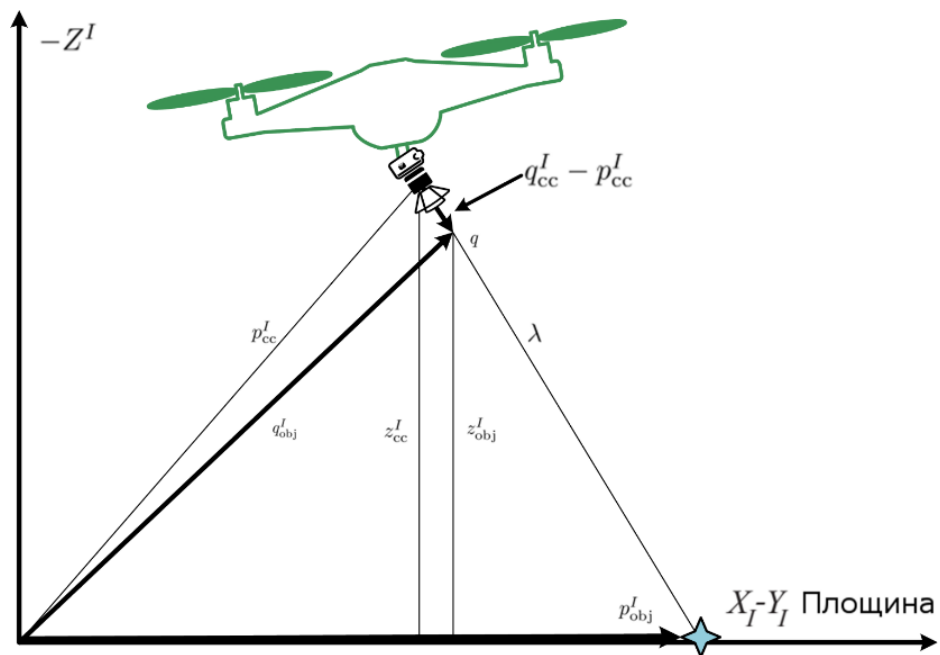


Рисунок 2.8 – Ілюстрація до опису моделі оцінки глибини λ за умов припущення про плоску поверхню і наявності знань про позицію і орієнтацію безпілотного апарату і його камери

Визначимо q_{obj}^I як q , тоді

$$q_{obj}^I = \begin{pmatrix} x_{obj}^I \\ y_{obj}^I \\ z_{obj}^I \\ 1 \end{pmatrix} = [CT_g^c T_b^g T_v^b T_I^v]^{-1} q \quad (2.1.4)$$

Як видно з рис. 2.8, якщо використовувати модель плоскої землі, то зв'язок між z-компонентами z_{obj}^I і p_{cc}^I визначається формулою

$$0 = z_{cc}^I + \lambda(z_{obj}^I - z_{cc}^I) \quad (2.1.5)$$

Якщо модель місцевості відома, то нуль в лівій частині (2.1.5) потрібно замінити відображенням висоти у точці, де оптична вісь перетинає місцевість. Оскільки як z_{cc}^I , так і z_{obj}^I відомі з (2.1.4) та (2.1.5), то λ можна обчислити за формулою

$$\lambda = \frac{z_{cc}^I}{z_{cc}^I - z_{obj}^I} \quad (2.1.6)$$

Оскільки по осі Z в інерціальній системі значення Z_I є доданім по відношенню до центру Землі, то z_{cc}^I буде від'ємною для абсолютних висот польоту. Таким чином, формула (2.1.6) дає додатне значення для λ , як і очікувалося.

З урахуванням розрахованого значення λ розміщення об'єкта в інерціальній системі може обчислюватися за формулою

$$p_{obj}^I = [CT_g^c T_b^g T_v^b T_I^v]^{-1} \Lambda q = T_v^I T_b^v T_g^b T_c^g C^{-1} \Lambda q$$

або аналогічно, у більш обчислювально ефективній формі

$$\bar{p}_{obj}^I = \bar{p}_{cc}^I + \lambda(\bar{q}_{obj}^I - \bar{p}_{cc}^I) \quad (2.1.7)$$

де \bar{p} – перші три елементи p .

Від точності вимірювання висоти залежить точність визначення координат об'єктів та ефективність стабілізації положення БПЛА. При цьому барометричний сенсор висоти є досить неточним, особливо на низьких висотах. Тому розроблений БПЛА використовує легкий і недорогий лідар Leddar One Lidar, який забезпечує 40-метровий діапазон та швидкість оновлення 70 Гц. На рис. 2.9 показано зовнішній вигляд дальноміру для більш точного вимірювання висоти.



Рисунок 2.9 – Зовнішній вигляд оптичного дальноміру Leddar One Lidar

Підключення сенсору висоти відбувається за послідовним інтерфейсом, що з'єднанується через вільний порт на контролері польоту.

Таким чином апаратне забезпечення безпілотного апарату складається з популярних компонентів, які добре документовані і мають довгострокову підтримку. При цьому здійснено калібрування параметрів камери і її підвісу, а також запропоновано формули розрахунку координат детектованих об'єктів для заданої конфігурації бортової системи БПЛА.

2.2 Програмне забезпечення бортової системи безпілотної літальної апарату для аеровідеомоніторингу

Програмне забезпечення інтелектуального БПЛА розподілене між його апаратними компонентами і взаємодіє по відкритих програмних протоколах і інтерфейсах (рис. 2.10).

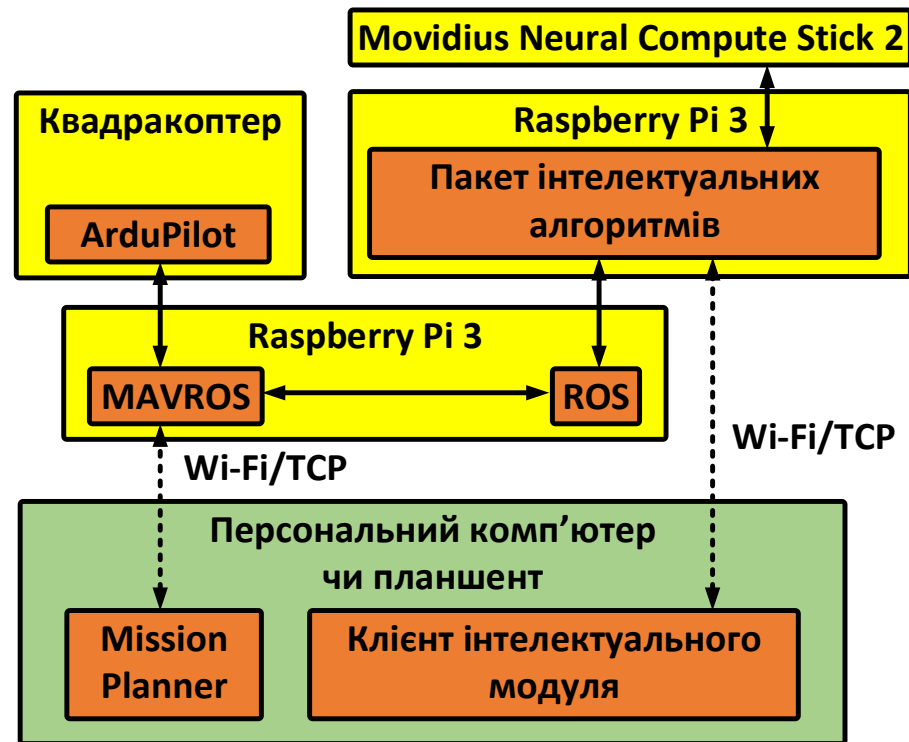


Рисунок 2.10 – Місцезнаходження основних програмних компонентів в апаратному забезпеченні еспериментальної системи моніторингу місцевості

Для комунікації з квадрокоптером використовується стандартизована система обміну повідомленнями MAVLink. Ця систем зручна для навігації та інших цілей, оскільки вона дозволяє передавати конкретні маршрути, а також команди керування орієнтацією для БПЛА. При цьому фреймворк під назвою “Операційна система роботів”, ROS, є спільною основою для обміну даними і командами в роботизованих системах. ROS призначений для модульного і мережевого функціонування, що дозволяє ефективно використовувати різні частини програмного забезпечення на пристроях, які знаходяться в одній і тій же

мережі. ROS містить плагіни для забезпечення додаткової функціональності. MAVROS є одним з таких плагінів, що формує програмний міст між ROS та MAVLink. Це дозволяє використовувати зв'язок в стилі ROS (вузли Publisher, Subscribers, і Services) для керування квадрокоптером, який приймає формат повідомлень MAVLink. ROS підтримує мови програмування C++, Python та Java.

Застосування ROS дозволило створити гнучку, та зручну для функціонального масштабування платформу та покращити модульність системи [7], [16]. ROS забезпечує стандартні служби операційної системи, такі як апаратну абстракцію, низькорівневий контроль пристроїв, реалізацію часто використовуваних функцій, передачу повідомлень між процесами та керування пакетами. ROS засновано на архітектурі графів, де обробка інформації відбувається в вузлах (англ. nodes), які можуть отримувати та передавати між собою повідомлення (англ. messages) за визначеною темою (англ. topics), а також підписатись на необхідну тему повідомлень. У відповідності з даною концепцією можна описати структурну схему взаємодії компонентів системи аеровідеомоніторингу з використанням програмного пакету ROS у вигляді графу (рис. 2.11).

Відповідно до наведеної схеми на рис. 2.11 вузли Camera_Det та Camera_Nav отримують поточні візуальні дані з визначених цифрових камер. Вузол Image_Proc призначено для низькорівневої обробки зображення та зменшення ефекту дисторсії. Розрахунок поточного положення за зображенням виконується у вузлі Visual_Navigation. Розпізнавання об'єктів інтересу виконується у вузлі Object_Detection. Вузол Object_Localization використовуючи інформацію, отриману від повідомлень теми roi_object (координати розпізнаних об'єктів на зображенні) та est_pose (обчислена поточна позиція системи в просторі), виконує розрахунок координат об'єкта інтересу відносно БПЛА в системі координат NED. Одночасно вузол Command_Trajectory виконує формування команд на зміну траєкторії польоту відповідно до завдання, або для уникнення зіткнення з розпізнаними перешкодами. Motor_Control реалізує

керування БПЛА відповідно до польотного завдання. Процес відбувається безперервно до отримання команди на припинення роботи.

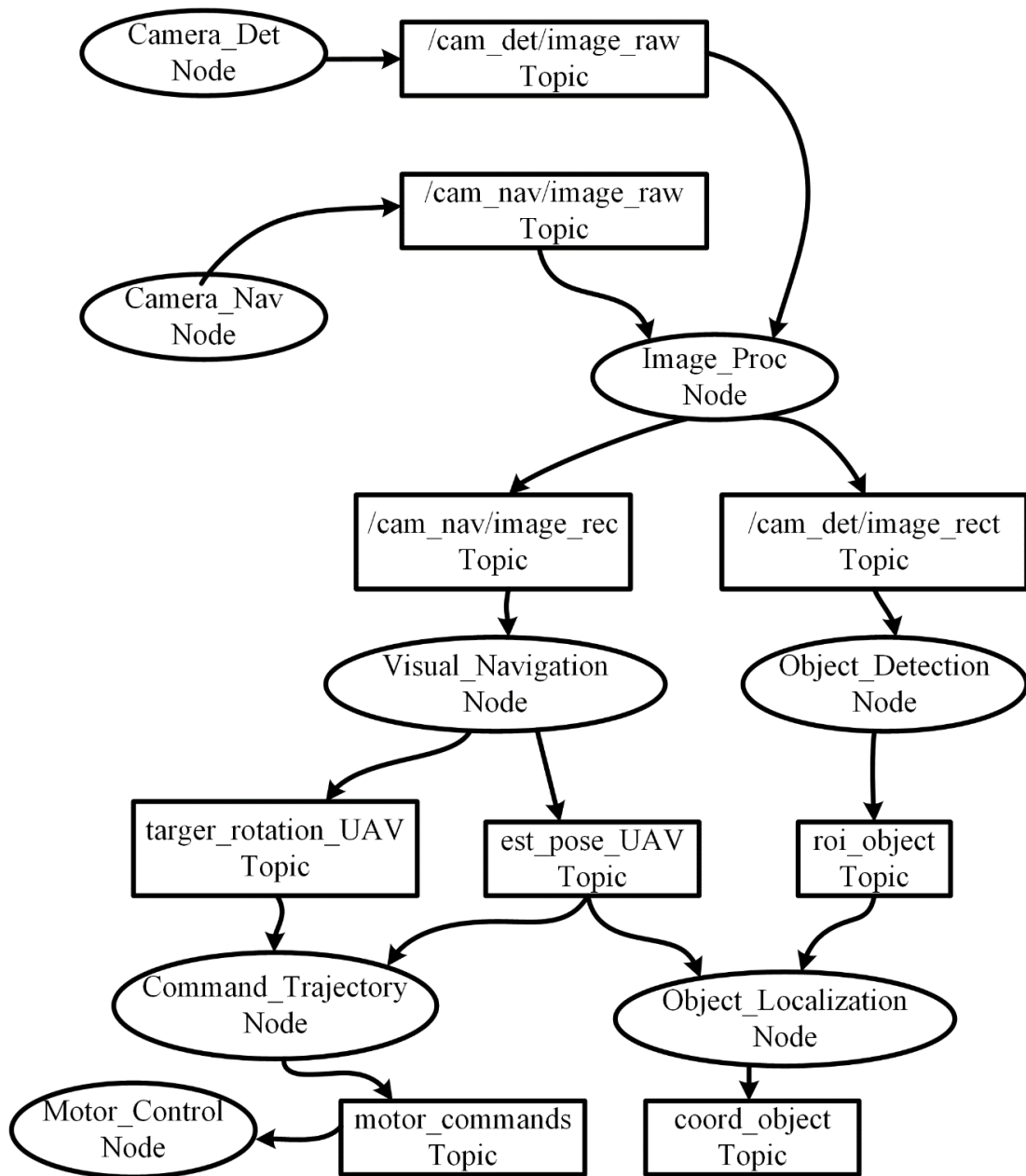


Рисунок 2.11 – Граф взаємодії компонентів системи аеромоніторингу з використання програмного пакету ROS

Для настройки і конфігурування місії БПЛА використовується програмне забезпечення Mission Planner Ground Control Station або Mission Planner GCS [16]. Головний екран Mission Planner показано на рис. 2.12. Головний екран містить вікно HUD (Heads-up Display) (рис. 2.13).

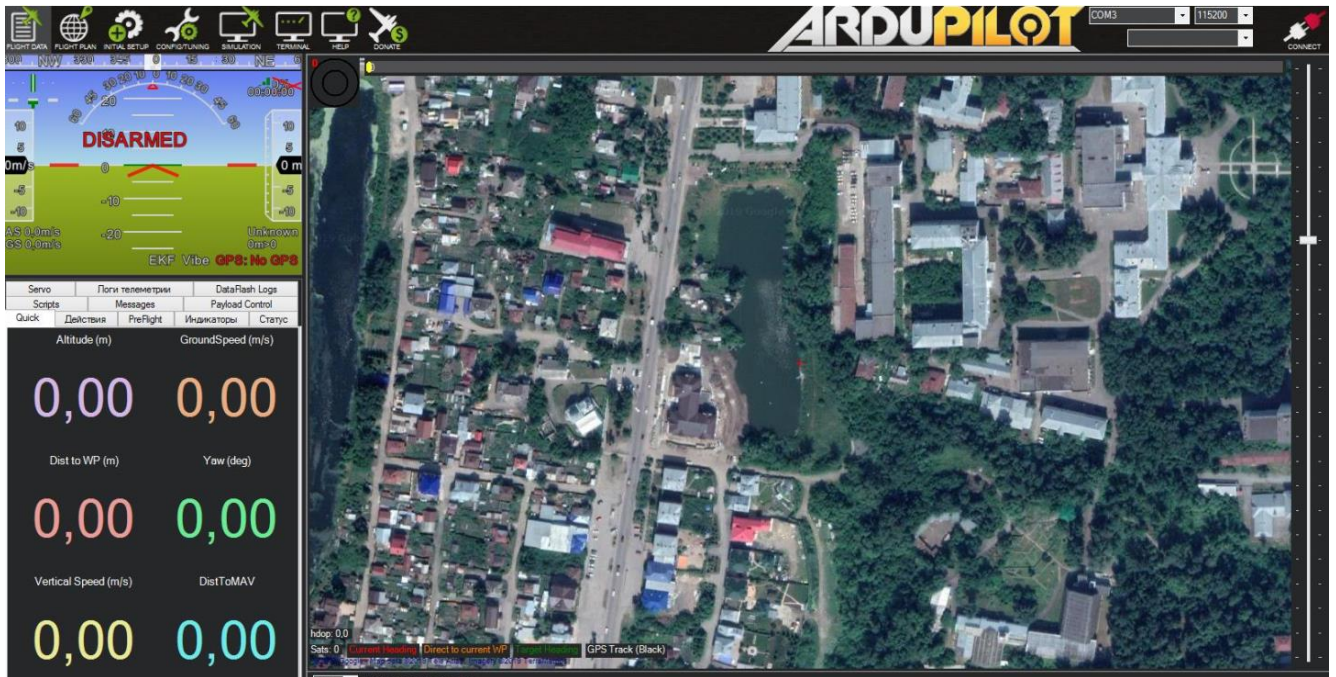


Рисунок 2.12 – Зовнішній вигляд головного екрану Mission Planner

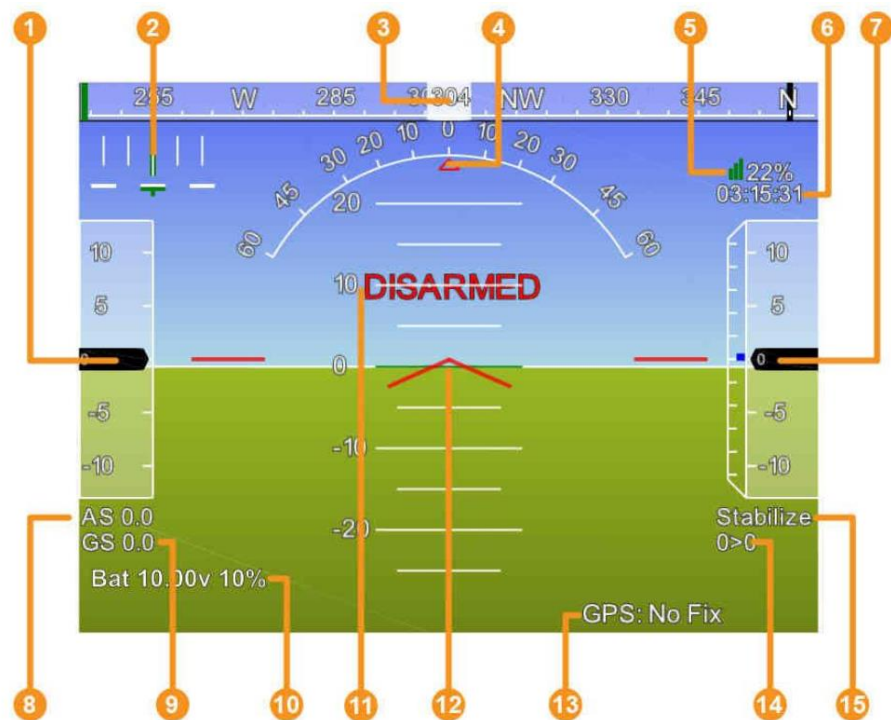


Рисунок 2.13 – Зовнішній вигляд HUD вікна в Mission Planner

На рис. 2.13 мітками помічено такі дані про стан БПЛА : 1) швидкість відносно землі; 2) помилка слідування треку і швидкість повороту; 3) курс, тобто напрямок руху; 4) кут крена; 5) якість передачі телеметрійних даних (% втрачених пакетів); 6) час отриманий по даним, від GPS; 7) висота по даним сенсора висоти (синя мітка показує швидкість зміни висоти); 8) швидкість відносно повітря при

наявності відповідного сенсору; 9) швидкість відносно землі; 10) стан батареї; 11) штучний горизонт; 12) кут тангажу; 13) стан GPS-модуля; 14) відстань до поточної точки маршруту і її номер; 15) поточний режим автопілота.

Для того щоб політний контролер коректно сприймав команди MAVlink від Mission Planner, він повинен працювати в одному з “автопілотованих” режимів : Stabilize, Flyby-Wire, Auto та інші. Якщо контролер знаходиться в режимі Manual, то всі команди будуть ігноруватися. Використання Mission Planner дозволяє до початку польоту виконати калібрування акселерометра, зовнішнього компаса та приймо-передавальної системи.

Програмно-апаратне забезпечення БПЛА містить поширені і недорогі компоненти, призначені для побудови безпілотних і робототехнічних систем. При цьому використання відкритого програмного забезпечення дозволяє здійснювати функціональне масштабування і доробку системи з мінімальними витратами ресурсів. Це забезпечує можливість швидкого прототипування, виробництва і впровадження розробки в практику.

На рис. 2.14 показано структурну схему програмного забезпечення бортової системи БПЛА для автономного аеровідеомоніторингу місцевості.

Аналіз рис. 2.14 показує, що до складу програмного забезпечення інтелектуальної бортової системи аналізу даних повинні входити модуль ідентифікації об’єкта інтересу для розпізнавання і визначення його координат. При цьому, модуль візуальної одометрії обчислює локальну позицію апарату в системі NED (англ. North-East-Down) для її перерахунку в глобальні координати, що дозволяє визначати локацію об’єкта інтересу у відповідних системах координат.

Модуль корекції одометра та оцінки перешкод здійснює паралельну корекцію результатів оцінки переміщення та пропонує повороти для уникнення прогнозованих перешкод.

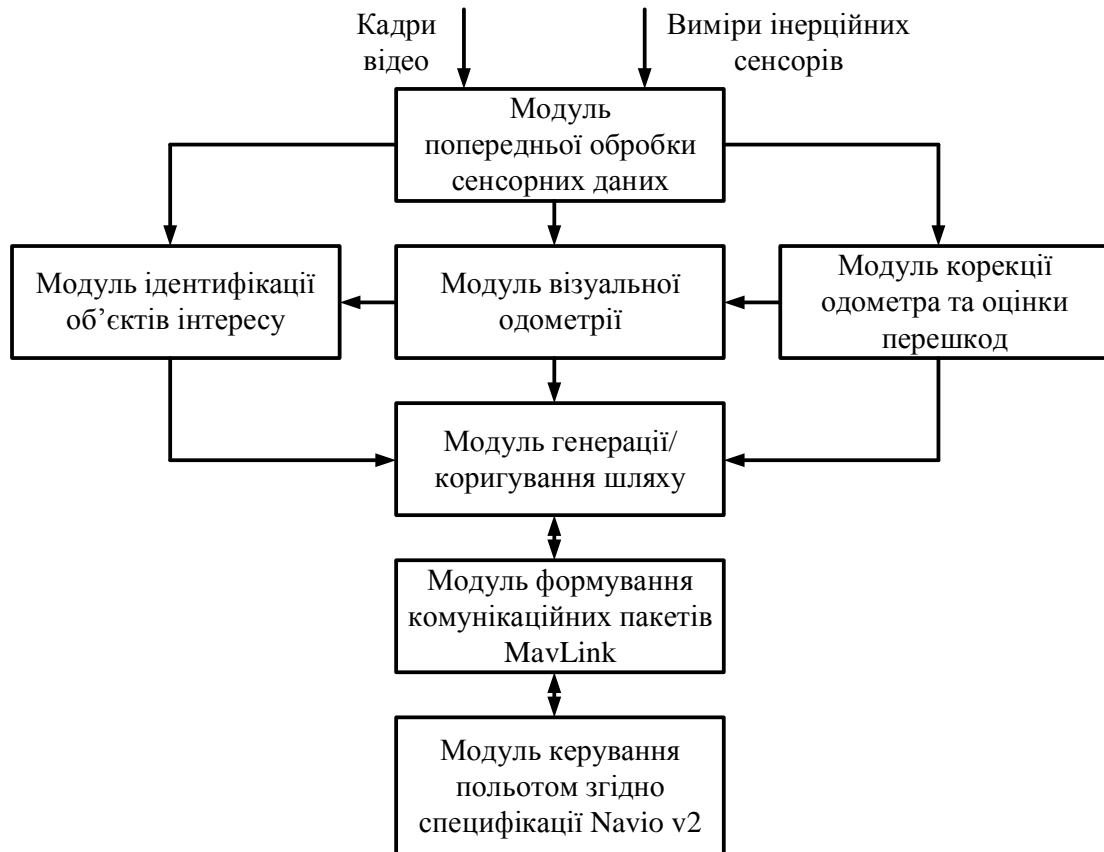


Рисунок 2.14 – Структурна схема програмного забезпечення інтелектуальної бортової системи БПЛА

Модуль ідентифікації об'єктів інтересу та модуль корекції одометра та оцінки перешкод можуть працювати в режимі екзамену та навчання для адаптації до нових умов функціонування. Дані модулі обчислюють власний ознаковий опис спостережень для максимального врахування контексту кожної задачі.

Програмна реалізація режимів навчання і екзамену здійснюється з використанням бібліотек Tensorflow HRT та OpenCV 4.1 з додатковими модулями (англ. contributed modules), а також Robot Operating System Melodic Morenia (для arm64) з бібліотекою ORB-SLAM. Для керування пам'яттю було встановлено Jemalloc. Бібліотеки оптимізовані з використанням SIMD-технології (NEON-інструкції). Обробка зображення здійснюється на основі бібліотеки OpenCV. Алгоритм розрідженого кодування [16] і його апроксимована версія у вигляді ансамблю дерев рішень, а також вирішувальні правила реалізуються з використанням фреймворку Tensorflow як в режимі навчання, так і екзамену.

Для реалізації ефективного розгортання і розпаралеленого виконання алгоритму зростаючого розріджено кодуемого нейронного газу на великому різноманітті обчислювальних платформ розроблено граф обчислень з використанням операцій фреймворку Tensorflow [13]. Його структурні елементи показано на рис. 2.15 та рис. 2.16.

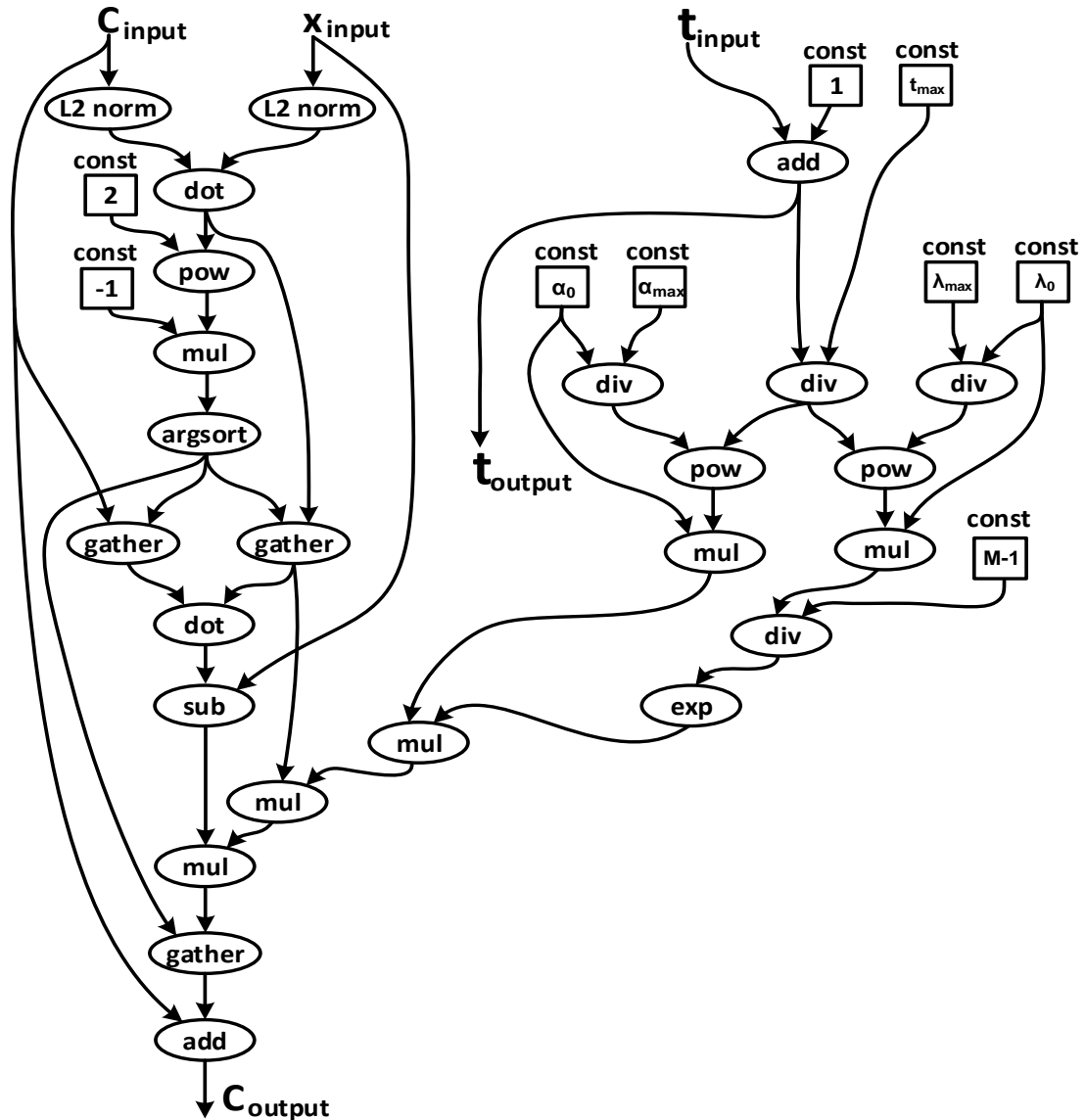


Рисунок 2.15 – Граф обчислень для реалізації однієї ітерації розріджено кодуемого нейронного газу

Як видно з рис. 2.15 на вхід графу надходить лічильник кількості ітерацій t_{input} , масив (тензор) нейронів C_{input} з зарезервованою місткістю під максимально допустиму кількість, вхідний одномірний вектор x_{input} . Опис операцій графу обчислень (рис. 2.15 та рис. 2.16), приведено у табл. 2.2.

Таблиця 2.2 – Операції фреймворку Tensorflow для реалізації зростаючого розріджено кодуючого нейронного газу

Назва операції	Короткий опис
add_n	Поелементна сума списку тензорів (масивів)
argmax	Знаходження індексу найбільшого елемента тензора
argmin	Знаходження індексу найменшого елемента тензора
bool_mask	Застосування бінарної маски до тензора
concat	Конкатенація списку тензорів вздовж одного виміру
count_nonzero	Кількість не нульових елементів тензора
cond	Оператор розгалуження потоку даних і операцій
cumsum	Обчислення сукупної суми тензора
less_equal	Поелементна перевірка умови менше або дорівнює X
log	Поелементний розрахунок значення натурального логарифму
max	Знаходження максимуму двох тензорів
one_hot	Унітарне кодування тензора
ones	Створення тензора елементи якого дорівнюють значенню 1
range	Створення тензора в заданих обмеженнях
reduce_sum	Розрахунок суми тензора за розмірами тензора
shape	Розрахунок форми тензора
shift	Поелементний зсув значень тензора
sign	Поелементний розрахунок функції sign тензора
size	Розрахунок розмірів тензора
split	Розбиття тензора на складові

Об'єднання карт ознак з різних шарів екстрактора ознак призводить до проблеми прокляття розмірності. Для усунення даної проблеми пропонується використовувати аналіз головних компонент. Його використання дозволяє знизити розмірність не беручи до розгляду ознаки, які є нечутливими до цільової

доменної області застосування. Вибір кількості головних компонент пропонується здійснювати у відповідності з критерієм Кайзера: вибір лише основних компонентів з власними значеннями, що перевищують 1.

Для забезпечення завадозахищеності та інформативності ознакового подання, пропонується розраховувати пікселі карти ознак за допомогою алгоритму швидкого узгодженого переслідування [16]. Для реалізації ефективного розгортання і розпаралеленого виконання алгоритму розрідженого кодування на великому різноманітті обчислювальних платформ розроблено граф обчислень з використанням операцій фреймворку Tensorflow (рис. 2.16). Опис операцій відповідного графу обчислень приведено у табл. 2.3.

Таблиця 2.3 – Операції фреймворку Tensorflow для реалізації розрідженого кодування

Найменування операції	Короткий опис
while_loop	Оператор повторення виконання заданого графу (body) до тих пір, доки виконується умова (cond) змінюючи з кожною ітерацією стан заданих змінних (loop_vars)
less	По елементна перевірка умови “менше ніж” з поверненням булевого масиву результатів
set_difference	Визначення різниці двох множин (тензорів)
matrix_inverse	Обчислення інверсії квадратної матриці (чи її спряжених транспозицій)

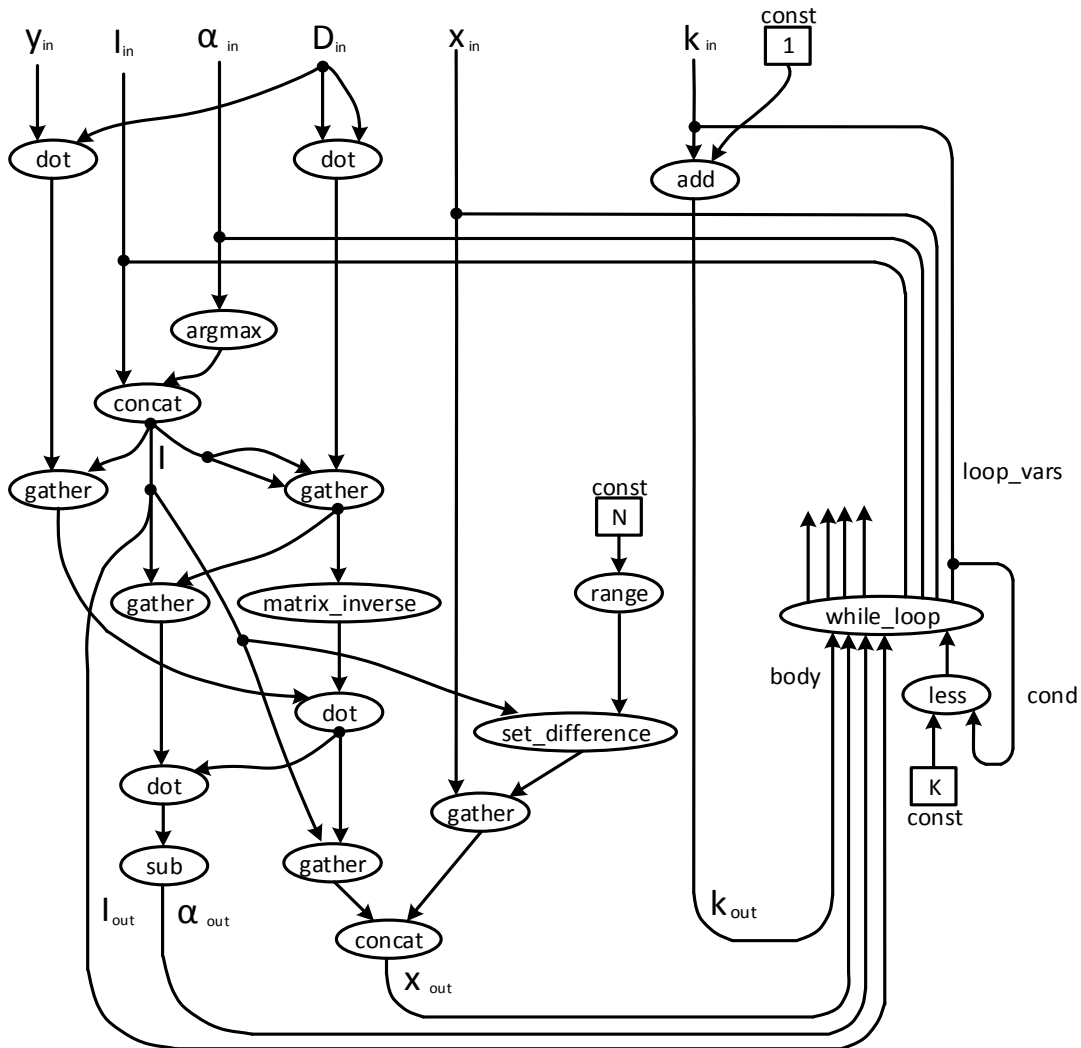


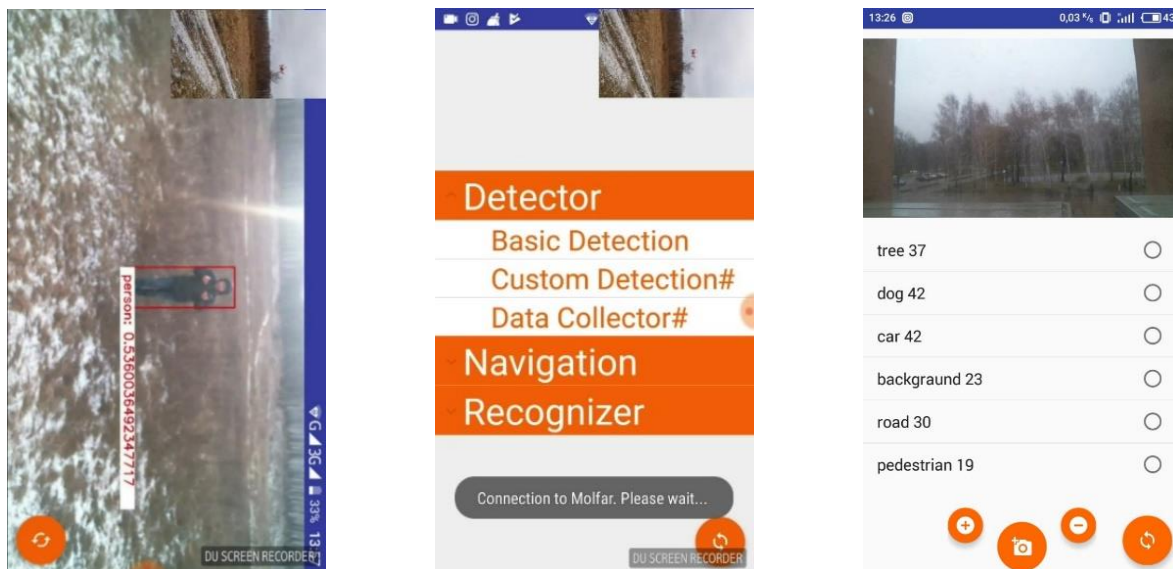
Рисунок 2.16 – Граф обчислень для реалізації однієї ітерації алгоритму швидкого узгодженого переслідування для розрідженого кодування

Для зменшення обчислювальної складності моделі аналізу даних шари екстрактор ознак з розрідженим кодуванням можуть бути апроксимовані ансамблем дерев рішень. Так само і вирішувальні правила можуть бути побудовані на основі ансамблю кластеризуючих дерев рішень, що виконують двійкове кодування спостережень. У фреймворку tensorflow для побудови ансамблю дере рішень є класи `tf.estimator.BoostedTreesRegressor`, `tf.estimator.BoostedTreesClassifier` та `tensorflow.contrib.tensor_forest`.

Останнім етапом алгоритму навчання є виконання тонкого настроювання високорівневих фільтрів екстрактора ознак, отриманих після навчання без вчителя, для врахування суттєвої незбалансованості між об'єктами інтересу та патчами фонових ділянок зображень. Тонке настроювання високорівневих шарів

здійснюється на основі алгоритму симуляції відпалу або зворотного поширення помилки з низьким `learning_rate`.

Реалізація клієнту інтелектуального модулю БПЛА було виконано на мові програмування Java у вигляді додатку для операційної системи Android. Архітектура даного додатку відповідає сучасним принципам побудови об'єктно-орієнтованих програм [16]. Використання шаблону Model-View-Controller (MVC), дозволило створити гнучку, та зручну для функціонального масштабування архітектуру [13]. Принципи інкапсуляції використовуються при розділенні MVC-компонентів системи моніторингу на рівні інтерфейсів та пакетів, що покращує модульність програми і підтримку її частин. Графічний інтерфейс системи моніторингу місцевості показано на рис. 2.17.



а – в режимі
розпізнавання об'єктів;

б – головне меню
вибору режимів;

в – режим донавчання
новим класам
розпізнавання

Рисунок 2.17 – Графічний інтерфейс роботи системи моніторингу

На рисунку рис. 2.17а показано роботу системи в режимі ідентифікації об'єктів інтересу, де більшу частину займає візуальне відображення розпізнаних об'єктів. На рис. 2.17б відображено загальну структуру меню вибору режимів

роботи системи. На рис. 2.17*в* показано роботу системи в режимі адаптації до розпізнавання нового алфавіту класів.

Таким чином, розроблена інтелектуальна система моніторингу місцевості за зображеннями адаптована для розгортання на різних мікропроцесорних платформах з використанням оптимізованих бібліотек графових обчислень та лінійної алгебри. При цьому графічний інтерфейс для конфігурування і навчання системи виконано у вигляді мобільного додатку для смартфона з використанням шаблону проектування Модель-Вид-Контролер, що спрощує функціональне масштабування.

2.3 Результати машинного навчання бортової системи безпілотного апарату на прикладі задачі інспекції труб

Інтеграція сучасних інформаційних технологій створює умови для виникнення як локальних, так і національних кіберфізичних систем. Автоматизація контролю та керування інфраструктурними об'єктами “розумних міст” є пріоритетним напрямком розвитку та впровадження інтелектуальних технологій аналізу даних.

Каналізаційні труби є важливими об'єктами інфраструктури, які потребують частого моніторингу. Звичайний метод аналізу стану каналізаційних труб передбачає огляд цих труб за допомогою відеоспостереження на предмет дефектів усередині труби. Відеоінспекцію труб, як правило, виконують за допомогою кравлерів, що є рухою платформою для переміщення камери в трубі. Запуск кравлера в трубу, керування переміщенням та витягування кравлера з труби здійснюється оператором. Однак здійснення інспекції в містах з інтенсивним транспортним потоком вимагає підвищення рівня автоматизації і автономності кравлерів. У розвинених країнах, на зразок Британії, вводяться штрафи за перевищення заданого часу стояння зупиненого автомобіля на дорозі, тому служби інспекції мають потребу в автономних безпілотних апаратах, які б дозволи звести процес інспекції до зупинки автомобіля, впускання безпілотника в

люк труби і вилучення безпілотною в наступному люці. На рис. 2.18 показано один з перспективних варіантів реалізації безпілотною апарату для інспекції стічних труб.



Рисунок 2.18 – Зовнішній вигляд БПЛА для інспекції стічних труб

Виявлені дефекти кодуються в рамках одного зі стандартів, серед яких найбільшого поширення набули британський MSCC5 та американський RASCP6 або RASCP7 стандарти [1]. При цьому для коректного кодування необхідно мати інформацію про локалізацію, орієнтацію, форму, тяжкість та близькість дефекту до вхідних-вихідних терміналів і відгалужень. Однак для екстракції цієї інформації необхідно мати дані про орієнтацію і розміщення камери спостереження в трубі, однак ці дані в явному вигляді, як правило, недоступні. Тому розпізнавання контексту спостереження є актуальною задачею. Крім того розпізнавання контексту спостережень має важливе значення для ефективної навігації з метою уникнення перешкод та оцінки власної орієнтації за умов недоступності глобальної супутникової навігації.

Алфавіт класів для розпізнавання контексті спостереження містить 9 основних контекстів (рис. 2.19) [2]. Клас X_1^0 характеризує орієнтацію камери вперед вздовж труби. Класи X_2^0 , X_3^0 , X_4^0 та X_5^0 характеризують орієнтацію

камери вліво, вправо, вгору та вниз відповідно, при якій спостерігач розуміє куди орієнтована камера, тобто поворот є неповним. Клас X_6^0 характеризує орієнтацію камери впритул до стінки зліва, справа, знизу чи згори, коли важко зрозуміти яка саме це частина труби (верхня, нижня, ліва чи права). Клас X_7^0 характеризує точку входу в трубу для інспекції, починаючи від люка до початку робочої області труби. Клас X_8^0 характеризує ситуації, які потрібно ігнорувати і не обробляти. До цих ситуацій відносяться втрата видимості, кадри відео до входу чи після виходу з труби. Клас X_9^0 характеризує завал труби, що перешкоджає наступному руху вздовж неї.



Рисунок 2.19 – Приклади зображень кожного з класів :

a – клас X_1^0 ; $б$ – клас X_2^0 ; $в$ – клас X_3^0 ; $г$ – клас X_4^0 ; $д$ – клас X_5^0 ;
 $е$ – клас X_6^0 ; $ж$ – клас X_7^0 ; $з$ – клас X_8^0 ; $и$ – клас X_9^0

Крім контексту зображення необхідно кожен кадр аналізувати на предмет наявності дефектів. У дослідженні з усього різноманіття труб розглядаються лише бетонні (Concrete Pipe) та скло-керамічні (Vitrified Clay) труби, для яких характерна широка номенклатура можливих дефектів. Алфавіт класів містить 9-ть найбільш поширених дефектів стічних труб (рис. 2.20) [5]. Клас X_1^o відповідає дефекту “roots”, що відповідає проникненню коренів дерев в трубу. Клас X_2^o відповідає дефекту “deposits”, що призводить до зменшення робочого перерізу стічної труби. Класи X_3^o , X_4^o , X_5^o , X_6^o та X_7^o характеризують різноманітні структурні дефекти і руйнування труби такі як “surface damage”, “crack”, “fracture”, “broken” та “hole” відповідно. Клас X_8^o відповідає дефекту “infiltration”, який виникає внаслідок входження води через дефект або пошкоджений стик, або через пористу ділянку на стінці труби. Клас X_9^o відповідає дефекту “Obstruction”, що суттєво утруднює потік або зменшує гідравлічну потужність. X_{10}^o відповідає нормальному стану труби і необхідний для регуляризації.

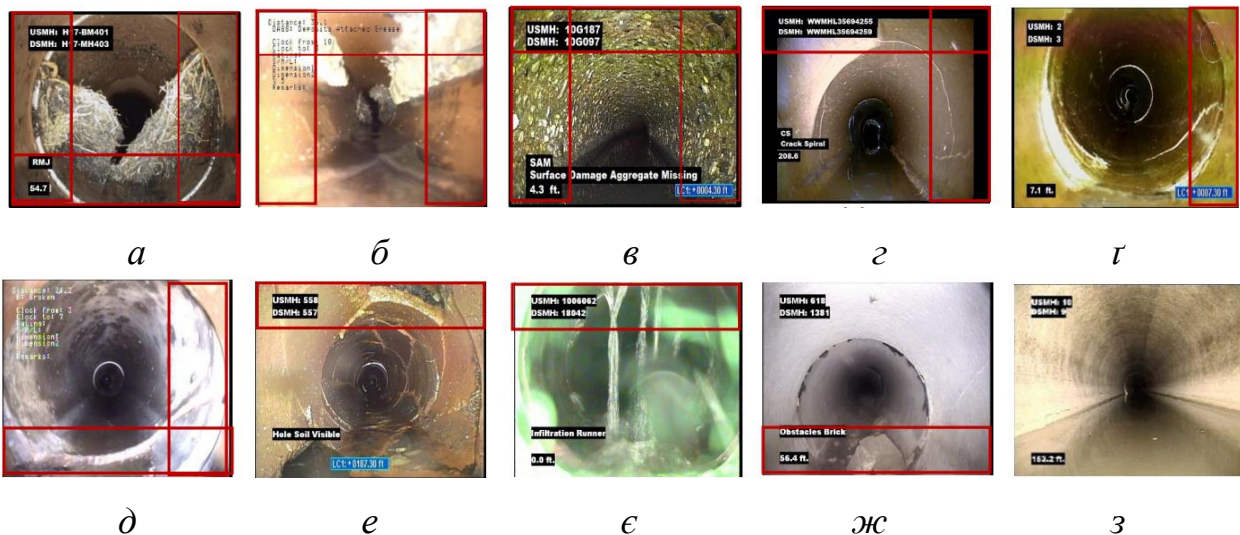


Рисунок 2.20 – Приклади зображень кожного з класів дефектів труби :

- a – клас X_1^o ; b – клас X_2^o ; c – клас X_3^o ; d – клас X_4^o ; e – клас X_5^o ;
 $ж$ – клас X_9^o ; $з$ – клас X_{10}^o

Аналіз дефектів труб відбувається не на всій фотографії, а на бокових частинах кадру, оскільки саме для них відомі значення з одометру, що спрощує їх локалізацію. Тому з кадру вирізаються прямокутні зображення, що відповідають 15% від зображення зверху, знизу та по бокам. Потім ці зображення «ресайзяться» до роздільної здатності 160x160 пікселів (рис. 2.21).



Рисунок 2.21 – Конвеєр обробки розмічених патчів зображення дефектів на крайових ділянках кадрів, для яких відоме значення одометра

Для кожного класу зібрано по 60 навчальних та тестових зразків. Дані для перевірки концепції надала компанія Ace Pipe Cleaning, Inc (Канзас Сіті, США). Звичайно кожен з цих класів має численну кількість підкласів, однак тут не буде розглядатися подальший аналіз класифікованого зображення. Кожне зображення використовується для розширення навчального набору в 4 рази шляхом застосування невеликих випадкових викривлень, яка включає варіацію яскравості, масштабу та поворотів до 10% від повного діапазону, а також накладання шумів типу сіль та перець.

Для експериментів і перевірки концепції як основу для екстрактора ознак пропонується використати згорткову мережу загального призначення MobileNet без повнозв'язних вихідних шарів з коефіцієнтом ємності, що дорівнює 0,25 [5]. Структуру запропонованої моделі класифікаційного аналізу даних показано на рис. 2.22а. Для визначення ефективності запропонованого підходу варто дослідити ефективність даного екстрактор ознак з класичними вихідними шарами. На рис. 2.20б показано базову (baseline) модель для порівняння ефективності з запропонованим підходом.

MobileNet екстрактор ознак (backbone)	MobileNet екстрактор ознак (backbone)
Шар Global Average Pooling 2D	Шар Global Average Pooling 2D
Шар Dropout (rate=0.5)	Шар Dropout (rate=0.5)
Повнозв'язний шар, Dense (128 вузлів)	Повнозв'язний шар, Dense (128 вузлів)
Сігмоїдний шар, Sigmoid	Сігмоїдний шар, Sigmoid
Округлючий шар	Повнозв'язний шар
Радіально-базисний шар	Нормуючий Softmax шар

a *б*

Рисунок 2.22 Структура моделі класифікатора : *a* – запропонована модель; *б* – модель з класичними вихідними шарами для порівняння ефективності

Спочатку навчимо класичну модель (рис. 2.22б) для класифікації контекстів спостереження. Для цього буде використано та categorical cross-entropy loss та оптимізатора Adam [2]. Розмір міні-батчу становить 32 зображення, а learning rate встановлено рівним 0,0001. На рис. 2.23 показано криві зміни точності та функції втрат на тестовій та навчальній вибірках залежно від кількості епох навчання класифікатора контекстів з класичним вихідним шаром softmax.

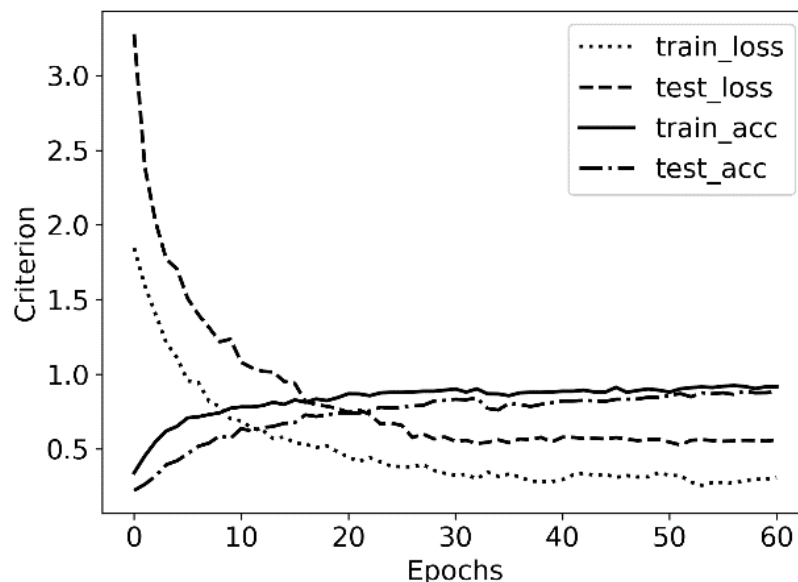


Рисунок 2.23 – Графік залежності точності та функції втрат під час навчання класифікатора контекстів від кількості епох навчання в рамках традиційного підходу

Аналіз рис. 2.23 показує, що зменшення помилки за тестовою (`test_loss`) вибіркою припинилося на 31 епосі. При цьому точність отриманої моделі (`test_acc`) становить 90%.

На рис. 2.24 показано результат навчання екстрактора ознак з сігмоїдним вихідним шаром, де перші 30 епох навчання відбувається з використанням функції помилки (1.2.1) та наступні 30 епох з використанням функції помилки (1.2.7). При цьому кожні 5 епох будуються інформаційно-екстремальні вирішальних правила та обчислюється усереднене за алфавітом класів значення інформаційного критерію ефективності (1.3.4).

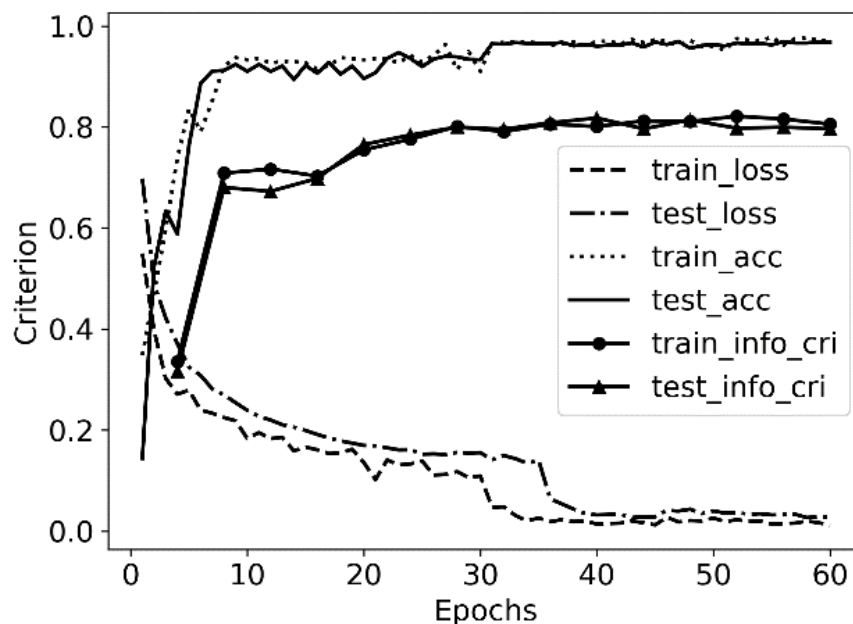


Рисунок 2.24 – Графік залежності точності, функції втрат та інформаційного критерію від кількості епох навчання в рамках запропонованого підходу

Аналіз рис. 2.24 показує, що починаючи з 10-ї епохи зменшення помилки за тестовою вибіркою дуже вповільнюється, а починаючи з 31-ї епохи відбувається різке підвищення точності вирішальних правил більше ніж на 2%. Проте після 40-ї епохи навчання покращення результату практично припиняється. Для навченої моделі точність за тестовою вибіркою становить 96,8%, а значення інформаційного критерію за навчальною і тестовою вибірками мало відрізняються

і рівне 0,81. При цьому середня відстань між центрами контейнерів, яка рівна 10, перевищує значення середнього радіусу контейнерів, який рівний 5,5. Це свідчить про високу завадозахищеність інформаційно-екстремальних вирішальних правил.

Таким чином, використання багатofазної схеми навчання з інформаційно-екстремальними вирішувальними правилами дозволяє підвищити точність моделі для розпізнавання контекстів спостереження під час інспекції труб на 6,8% порівняно з традиційною схемою навчання з softmax вихідним шаром.

Розглянемо результати машинного навчання класифікатора дефектів на патчах кадрів у контексті орієнтації камер прямо (forward). На рис. 2.25 показано зміни точності моделі на тестовій (test_acc) і навчальній (train_acc) вибірках. Кожен міні-батч містить 32 зображення, а коефіцієнт швидкості навчання дорівнює 0,0001.

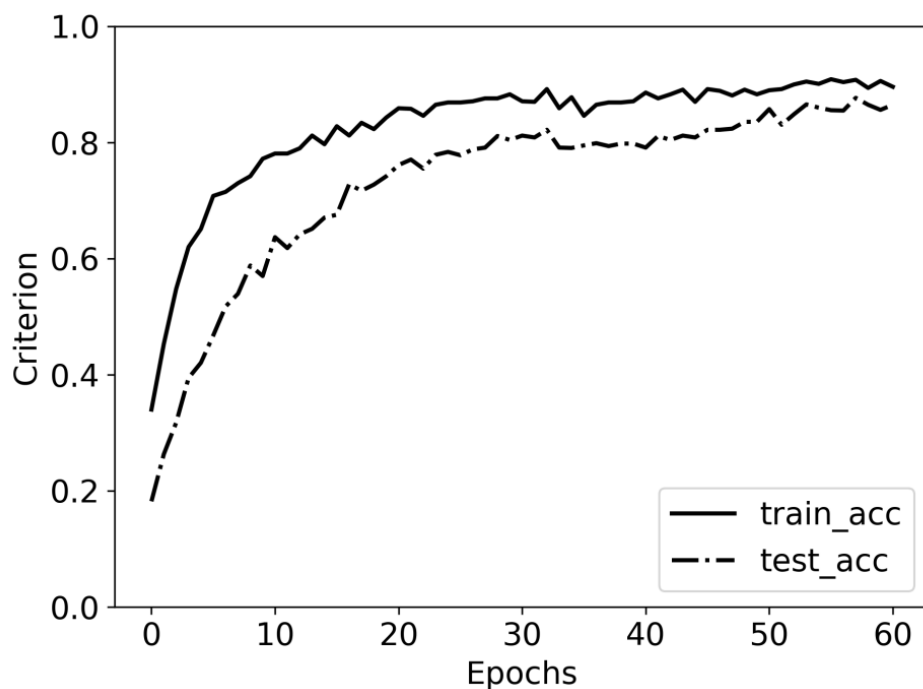


Рисунок 2.25 – Графік залежності точності та функції втрат під час навчання класифікатора дефектів від кількості епох навчання в рамках традиційного підходу

Аналіз рис. 2.25 показує, що підвищення точності зупиняється після 40 епох навчання з максимальним значенням точності результуючої моделі, що дорівнює

86% на стестовій вибірці. При цьому відстань між кривою на тестовій і на навчальній вибірці свідчить про наявність невеликого ефекту перенавчання.

На рис. 2.26 показано результати навчання запропонованого екстрактора ознак з сігмоїдним шаром і функцією втрат (1.2.4). При цьому кожні 5 епох відбувається побудова інформаційно-екстремальних вирішувальних правил і обчислення усередненого за алфавітом класів інформаційного критерію (1.3.4) на тестовій (*test_info_cri*) та навчальній (*train_info_cri*) вибірках. Після 30 епох навчання модель було збережено для наступного експерименту.

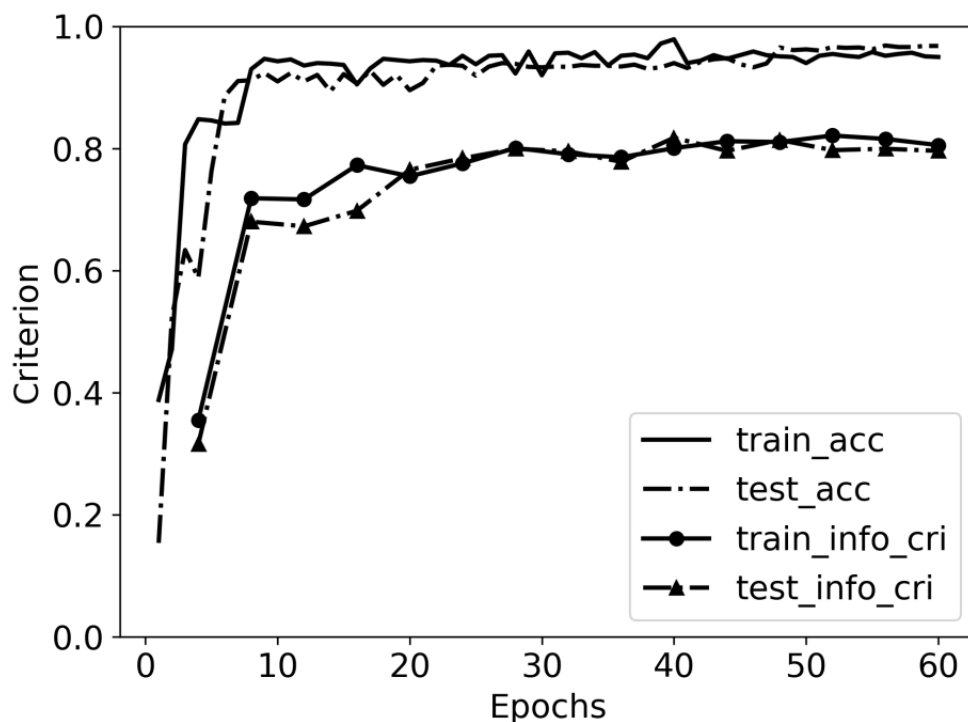


Рисунок 2.26 – Графік залежності точності та інформаційного критерію від кількості епох навчання в рамках запропонованого підходу з функцією втрат (1.2.4)

Аналіз рис. 2.26 показує, що з 10 епохи зростання точності значно уповільнилося, а після 40 епох було досягнуто максимальну точність, що дорівнює 95%, що на 9% більше ніж в базовій (*baseline*) моделі. Однак після 40 епох навчання, подальше покращення практично припинилося. При цьому тестові і навчальні криві майже співпадають, що свідчить про високу узагальнюючу

здатність отриманих вирішувальних правил. Усереднене за алфавітом класів значення інформаційного критерію дорівнює 0,815 а середня відстань між центрами гіперсферичних контейнерів дорівнює 19 кодових одиниць, а середній радіус контейнерів класів рівний 8 кодових одиниць. Це свідчить про високу завадозахищеність вирішувальних правил [2].

На рис. 2.27 показано подальші результати навчання моделі, збереженої на попередньому етапі після 30 епох навчання. Наступне навчання здійснюється також 30 епох, але з використанням функції втрат (1.2.7) з метою зменшення внутрішньокласової дисперсії в просторі Хемінга.

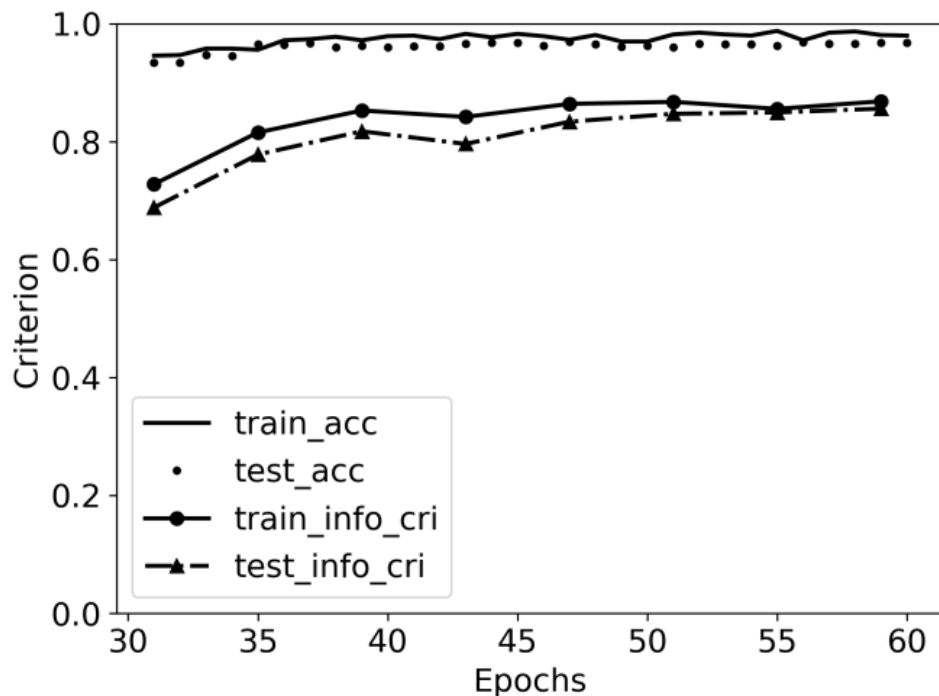


Рисунок 2.27 – Графік залежності точності та інформаційного критерію від кількості епох навчання в рамках запропонованого підходу з функцією втрат (1.2.7)

Аналіз рис. 2.27 показує, що використання функції joint binary cross-entropy loss (1.2.7) забезпечує зростання точності на 3% порівняно з попереднім етапом. При цьому усереднене за алфавітом класів значення інформаційного критерію (1.3.4) дорівнює 0,856.

Таким чином мультифазне навчання з інформаційно-екстремальними вирішувальними правилами забезпечує підвищення точності розпізнавання дефектів порівняно з традиційним підходом на 11% за ту ж кількість епох навчання за умов обмеженого обсягу розмічених навчальних даних.

ВИСНОВКИ

У науково-технічній роботі виконано важливе наукове завдання – підвищення ефективності бортової системи БПЛА, що в автономному режимі здійснює класифікаційний аналіз аеровідео спостережень за умов інформаційних та ресурсних обмежень.

Основні наукові та практичні результати роботи зазначені нижче.

1. Проаналізовано сучасний стан та тенденції розвитку інформаційних технологій розпізнавання зображень і їх придатність для реалізації відеомоніторингу місцевості з використанням автономного БПЛА. Встановлено, що архітектури моделей аналізу візуальних даних поступово досягають насичення технічною новизною, а домінуючим методом машинного навчання є застосування алгоритму зворотного поширення помилки. Проте даний метод має ряд недоліків: повільна збіжність; потреба значного обсягу розмічених навчальних даних та обчислювальних ресурсів для забезпечення прийняттого результату; невизначеність щодо оптимальних значень гіперпараметрів; резервування інформаційної ємності відбувається за рахунок гіперпараметризації, що збільшує накладні витрати і обумовлює схильність до перенавчання. При цьому одним із шляхів підвищення ефективності машинного навчання є гібридизація і комплексування різних підходів і протоколів оптимізації параметрів моделі аналізу даних, серед яких перспективним є поєднання ідей і методів теорії інформації з принципами синтезу моделей в рамках геометричного підходу.

2. Уперше запропоновано поєднання ідей і методів розрідженого кодування і дистиляції знань для синтезу багат шарового екстрактора ознакового опису зображень, що дозволяє знизити вимоги до обсягу розмічених даних під час навчання та знизити обчислювальну складність моделі під час екзамену.

3. Уперше розроблено метод синтезу моделей аналізу візуальних даних, що оснований на поєднанні критеріїв і принципів теорії інформації і кодування з ідеями регуляризації в рамках геометричного підходу для забезпечення робастності за найскладніших у статистичному відношенні умов. В рамках даного

підходу здійснюється пропонується використання дискретного ознакового подання, що забезпечує регуляризаційний і мета-регуляризаційний ефекти і дозволяє зменшувати кількість конфігураційних налаштувань бортової системи. При цьому інформаційний критерій оптимізації забезпечує робастність до складних у статистичному сенсі ситуацій, що дозволяє зменшити кількість інтерактивних взаємодій з візуальним інтерфейсом для донавчання/перенавчання бортової системи.

4. Досліджено ефективність запропонованих моделей і методів навчання на прикладі задачі візуальної інспекції труб. Фізичне моделювання показало, що використання запропонованих алгоритмів порівняно з традиційними дозволяє підвищити точність розпізнавання контекстів спостереження на 6,8%, а дефектів труб на 11% за умов обмеженого обсягу розмічених даних.

5. Розроблено апаратну частину БПЛА квадрокоптерного типу. Встановлено та сконфігуровано операційну систему бортового комп'ютера та пакетів фреймворку Robot Operating System. Вивчено можливості існуючих інформаційних технологій для побудови гнучких та функціонально масштабованих інтелектуальних систем. Показано, що розроблені інтелектуальні компоненти БПЛА легко інтегруються з БПЛА за рахунок використання інформаційної інфраструктури операційної системи ROS. При цьому використано бібліотеки та апаратні компоненти, що забезпечують прийнятну для практичного використання достовірність та оперативність рішень бортової системи БПЛА.

6. Результати досліджень використано під час надання послуг за господарською темою №52.17-2020.СП/01 від 11 лютого 2020 р. : «Послуги з розробки складової частини програмного забезпечення інтелектуального модуля машинного зору для наземного транспорту» (замовник: ТОВ «РОЗУМНІ ТЕХНОЛОГІЇ «ТИТУЛ»», м. Суми). Також результати досліджень впроваджено в початковий процес під час підготовки лекційного матеріалу з дисциплін «Мови програмування інтелектуальних систем» та «Introduction to Data Science». Також захищено кандидатську дисертацію на тему «Моделі і методи інформаційної

технології автономного відеомоніторингу місцевості безпілотним літальним апаратом”.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Moskalenko A. Deep feature extractor with information-extreme decision rules for visual classification of sewer pipe defects and its training method / A. Moskalenko, V. Moskalenko, M. Zaretskyi, V. Lysyuk // Data Stream Mining & Processing 2020. – Lviv, Ukraine, 21-25 Aug. 2020. – pp.191-195. (Scopus & Wos).
2. Moskalenko, V. Sewer Pipe Defects Classification Based on Deep Convolutional Network with Information-extreme Error-correction Decision Rules / A. S. Moskalenko, V.V. Moskalenko, M. O. Zaretskyi, V. Lysyuk // Communications in Computer and Information Science (CCIS-2020) – in press.
3. Moskalenko V. Multi-Layer Model and Training Method for Information-Extreme Malware Traffic Detector / V. Moskalenko, A. Moskalenko, A. Shaiekhov, M.Zaretskyi // The Third International Workshop on Computer Modeling and Intelligent Systems – April 27-May 1. – 2020. – p. 288-299. (Scopus).
4. Moskalenko V. V. Multi-Layer Model And Training Method For Malware Traffic Detection Based On Decision Tree Ensemble / V. V. Moskalenko, A. S. Moskalenko, M. O. Zaretsky, A. M. Kudryavtsev, V. A. Semashko // Radioelectronic and Computer Systems – No 2. – 2020. – P. 92-101.
5. Москаленко В.В. Модель і метод навчання класифікатора контекстів спостереження на зображеннях відеоінспекції стічних труб / В.В. Москаленко, М. О. Зарецький, Я. Ю. Ковальський, С. С. Мартиненко // Радіоелектронні і комп'ютерні системи. – DOI 10.32620/reks.2020.3.06 – в друці.
6. Москаленко В. В. “Моделі і методи інтелектуального аналізу багатовимірних даних да умов апіорної невизначеності”: монографія / В. В. Москаленко. – Суми: Видавництво СумДУ, 2019. – 184 с.
7. Інтелектуальна автономна бортова система безпілотного літального апарату для ідентифікації об'єктів на місцевості [Текст]: Розробка інформаційного та програмного забезпечення бортової системи безпілотного апарату, що функціонує в режимі навчання вирішальних правил та класифікаційного аналізу спостережень: звіт про НДР (проміжний) / кер. В.В. Москаленко. – Суми: СумДУ,

2019. – 67 с. – Режим доступу: https://essuir.sumdu.edu.ua/bitstream-download/123456789/79046/1/Moskalenko_intelektualna_1538.pdf

8. Gwon Y. Deep Sparse-coded Network (DSN) / Y. Gwon, H. Cha, H. T. Kung // International Conference on Pattern Recognition (ICPR). – México, Dec.4-8. – 2016. – pp. 2611–2616. – DOI: <https://doi.org/10.1109/ICPR.2016.7900029>.

9. Tariyal S. Greedy Deep Dictionary Learning / S. Tariyal, A. Majumdar, R. Singh, M. Vatsa // IEEE Access. – 2016. – Vol. 4 – P. 10096–10109. – DOI: [10.1109/ACCESS.2016.2611583](https://doi.org/10.1109/ACCESS.2016.2611583).

10. Patel H. Dictionary Properties for Sparse Representation: Implementation and Analysis / H. Patel, H. Mewada // Journal of Artificial Intelligence. – 2018. – Vol. 11, Issue 1. – P. 1–8.

11. Li H.-T. Layer-Level Knowledge Distillation for Deep Neural Network Learning / S.-C. Lin, C.-Y. Chen, C.-K. Chiang // Applied Sciences. – 2019. – Vol. 9. – P. 1966.

12. Zhan H. Deepshoe: An improved multi-task viewinvariant cnn for street-to-shop shoe retrieval / H. Zhan, B. Shi, L. Y. Duan, A. Kot // Computer Vision And Image Understanding. – 2019. – Vol. 180. – P. 23–33. – DOI: <https://doi.org/10.1016/j.cviu.2019.01.001>.

13. Інтелектуальна автономна бортова система безпілотного літального апарату для ідентифікації об'єктів на місцевості [Текст]: Розробка інформаційного та програмного забезпечення бортової системи безпілотного апарату, що функціонує в режимі самонавчання екстрактора ознакового опису середовища для навігації та класифікаційного аналізу спостережень: звіт про НДР (проміжний) / кер. В. В. Москаленко. – Суми: СумДУ, 2018. – 91 с. – Режим доступу: https://essuir.sumdu.edu.ua/bitstream-download/123456789/73907/1/Moskalenko_V._intelektualna_1477.pdf.

14. Khan A. A Survey, of the Recent Architectures of Deep Convolutional Neural Networks / A. Khan, A. Sohail, U. Zahoora, A. S. Qureshi // 2019. – P. 1–62. – Available at: <https://arxiv.org/ftp/arxiv/papers/1901/1901.06032.pdf>.

15. Ramachandram D. Deep Multimodal Learning / D. Ramachandram, G.W. Taylor // IEEE Signal Processing Magazine. – 2017. – P. 96 – 108.
16. Моделі і методи інформаційної технології автономного відеомоніторингу місцевості безпілотним літальним апаратом [Текст] : автореф. дис. ... канд. техн. наук : 05.13.06 / Коробов Артем Геннадійович; Нац. аерокосм. ун-т ім. М. Є. Жуковського "Харків. авіац. ін-т". – Харків, 2019. – 20 с.
17. Zhao Z. Object Detection With Deep Learning: A Review / Z. Zhao, P. Zheng, S. Xu, X. Wu // IEEE Transactions on Neural Networks and Learning Systems. – 2019. – P. 1 – 21. – DOI: 10.1109/TNNLS.2018.2876865.
18. Hu W. Learning discrete representations via information maximizing self augmented training. / W. Hu, T. Miyato, S. Tokui, E. Matsumoto // Proceedings of International Conference on Machine Learning. – Vol. 70. – P. 1558–1567.