

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. Чорноус

ЗВІТ

ПРО НАУКОВО-ДОСЛІДНУ РОБОТУ

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

РОЗРОБКА ІНФОРМАЦІЙНОГО ТА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ  
БОРТОВОЇ СИСТЕМИ БЕЗПІЛОТНОГО АПАРАТУ, ЩО ФУНКЦІОНУЄ  
В РЕЖИМІ САМОНАВЧАННЯ ЕКСТРАКТОРА ОЗНАКОВОГО ОПИСУ  
СЕРЕДОВИЩА ДЛЯ НАВІГАЦІЇ ТА КЛАСИФІКАЦІЙНОГО  
АНАЛІЗУ СПОСТЕРЕЖЕНЬ

(проміжний)

Начальник НДЧ  
канд. фіз.-мат. наук,  
старш. наук. співроб.  
Керівник НДР  
канд. техн. наук

Д.І. Курбатов

В.В. Москаленко

2018

Рукопис завершено 26 грудня 2018 р.

Результати цієї роботи розглянуто на засіданні наукової ради СумДУ, протокол від  
27.11.2017 № 5

## СПИСОК АВТОРІВ

Керівник НДР,  
канд. техн. наук,  
провідн. наук. співроб.

(26.12.2018)

Москаленко В. В.  
(вступ, висновки,  
підрозділи 1.1, 1.2, 1.5, 2.3)

Відповідальний виконавець  
канд. техн. наук,  
старш. наук. співроб.

(26.12.2018)

Москаленко А. С.  
(підрозділи 1.4, 2.3, 2.4)

Виконавці:  
канд. техн. наук,  
старш. наук. співроб.

(26.12.2018)

Берест О. Б.  
(підрозділи 1.1, 2.4)

канд. техн. наук,  
старш. наук. співроб.

(26.12.2018)

Мартиненко С. С.  
(підрозділи 2.1, 2.3)

канд. фіз.-мат. наук,  
старш. наук. співроб.

(26.12.2018)

Журба В. О.  
(підрозділи 2.1, 2.2)

канд. техн. наук,  
старш. наук. співроб.

(26.12.2018)

Нагорний В. В.  
(підрозділ 1.3)

|  |   |
|--|---|
| мол. наук. співроб.<br><br>(26.12.2018)                    | Окопний Р. П.<br>(підрозділ 2.2)            |
| аспірант,<br>мол. наук. співроб.<br><br>(26.12.2018)       | Коробов А. Г.<br>(підрозділи 1.3, 1.6, 2.2) |
| канд.техн.наук.<br>мол. наук. співроб.<br><br>(26.12.2018) | Бойко О. В.<br>(підрозділ 2.3)              |
| аспірант,<br>лаборант<br><br>(26.12.2018)                  | Зарецький М. О.<br>(підрозділ 2.4)          |

## РЕФЕРАТ

Звіт про НДР: 91 с., 9 табл., 28 рис., 40 джерел.

БОРТОВА СИСТЕМА БЕЗПІЛОТНОГО ЛІТАЛЬНОГО АПАРАТУ, ЕКСТРАКТОР ОЗНАК, ІНФОРМАЦІЙНИЙ КРИТЕРІЙ, ІНФОРМАЦІЙНО-ЕКСТРЕМАЛЬНА ІНТЕЛЕКТУАЛЬНА ТЕХНОЛОГІЯ, МАШИННЕ НАВЧАННЯ, НАВІГАЦІЯ, ОПТИМІЗАЦІЯ, РОЗРІДЖЕНЕ КОДУВАННЯ

Об'єкт дослідження – процеси машинного навчання бортової системи безпілотного апарату та автономної ідентифікації об'єктів на місцевості за даними різнотипних сенсорів.

Мета роботи – підвищення функціональної ефективності бортової системи безпілотного літального апарату, що здійснює у автономному режимі локальну навігацію та класифікаційний аналіз спостережень за даними різнотипних сенсорів за умов ресурсних та інформаційних обмежень, і неповної визначеності, обумовленої довільними умовами спостереження та нестационарністю середовища.

Методи дослідження – методи технології нейронних мереж, розрідженого кодування та інформаційно-екстремальної інтелектуальної технології оцінки ефективності моделей аналізу даних.

Розроблено нові моделі однорівневої та ієрархічної екстракції ознакового опису середовища для задач навігації та детектування об'єктів на місцевості. Їх архітектура дозволяє підвищити точність вирішальних правил за рахунок врахування більшої кількості контекстної інформації, порівняно з аналогами. Вперше розроблено нові категорійні моделі та функціональні діаграми процесу навчання екстрактора ознак, що полегшує розробку, аналіз та реінжиніринг алгоритмів функціонування бортової системи. Вперше розроблено графи обчислень для реалізації алгоритмів навчання екстрактора ознакового опису спостережень без вчителя, що дозволяє здійснювати їх ефективне розгортання для розпаралеленого виконання на великому різноманітті обчислювальних платформ. Виконана програмна реалізація запропонованих алгоритмів. Вперше розроблено алгоритм зростаючого

розріджено кодуючого нейронного газу, оснований на схемі м'якої конкуренції, правилі Ойа та вставці нових нейронів за порогом досяжності, для навчання верхніх шарів ієрархічного екстрактора ознакового опису. Це дозволяє ефективно використовувати нерозмічені навчальні дані, автоматично визначати необхідну кількість нейронів та забезпечити близький до оптимального розподіл нейронів по даним. Удосконалено алгоритм тонкої настройки параметрів моделі шляхом використання симуляції відпалу для пошуку глобального оптимуму комплексного критерію, що враховує ефективність вирішальних правил та обчислювальну складність екстрактора ознак. На відміну від алгоритму зворотного поширення помилки, у запропонованому алгоритмі використовується мультиплікативна згортка прямих критеріїв та метаевристичний пошук, що покращує конвергенцію до глобального оптимуму.

Результати виконання роботи впроваджено в навчальний процес при підготовці лекційних курсів та лабораторних робіт з навчальних дисциплін «Мови програмування інтелектуальних систем», «Інтелектуальні системи керування» та «Introduction to Data Science», а також при написанні п'яти магістерських робіт.

## ЗМІСТ

|  |    |
|--|----|
| СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ .....   | 7  |
| ВСТУП .....  | 8  |
| 1 МОДЕЛІ ЕКСТРАКЦІЇ ОЗНАКОВОГО ОПИСУ ДЛЯ<br>КЛАСИФІКАЦІЙНОГО АНАЛІЗУ СПОСТЕРЕЖЕНЬ ТА НАВІГАЦІЇ .....   | 10 |
| 1.1 Аналіз об'єкту та обґрунтування напрямку дослідження .....   | 10 |
| 1.2 Формалізована постановка задачі синтезу екстрактора ознак.....   | 17 |
| 1.3 Категорійно-функторні моделі процесу навчання бортової системи<br>аналізу спостережень.....  | 20 |
| 1.4 Критерій функціональної ефективності моделі аналізу даних .....  | 26 |
| 1.5 Модель екстракції ознакового опису для задачі детектування<br>об'єктів.....  | 32 |
| 1.6 Модель екстракції ознакового опису для задачі автономної навігації ..  | 43 |
| 2 ОПТИМІЗАЦІЯ ПАРАМЕТРІВ ФУНКЦІОНУВАННЯ МОДЕЛІ<br>АНАЛІЗУ ДАНИХ .....  | 48 |
| 2.1 Алгоритми пошукової оптимізації параметрів екстрактора ознак.....  | 48 |
| 2.2 Алгоритм та результати машинного навчання однорівневого<br>екстрактора ознакового опису для задачі детектування об'єктів на<br>місцевості..... | 54 |
| 2.3 Алгоритм та результати машинного навчання ієрархічного<br>екстрактора ознакового опису для задачі детектування об'єктів на<br>місцевості.....  | 63 |
| 2.4 Алгоритм машинного навчання ієрархічного екстрактора ознакового<br>опису для задачі автономної навігації .....                                 | 74 |
| ВИСНОВКИ.....  | 83 |
| ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ .....   | 86 |

## **СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ**

БПЛА – безпілотний літальний апарат;

ІЕІ – технологія – інформаційно-екстремальна інтелектуальна технологія;

ОУП – алгоритм ортогонального узгодженого переслідування.

## ВСТУП

БПЛА знайшли широке застосування у військовій, промисловій та побутовій сферах. Бурхливий розвиток ринку БПЛА призводить до підвищення рівня конкуренції і обумовлює актуальність пошуку нових шляхів покращення співвідношення ціни та якості. При цьому БПЛА від провідних виробників в плані конструктивного виконання досягли меж насичення технічною новизною і найбільш перспективним напрямком їх удосконалення є впровадження інтелектуальних інформаційних технологій для розширення функціональних можливостей та підвищення рівня автономності. Все частіше з'являються спроби застосування технологій штучного інтелекту в інформаційному забезпеченні БПЛА, однак вони спрямовані на вирішення вузького кола задач і адаптовані до конкретних умов функціонування, що робить їх неуніверсальними, а також ненадійними в нештатних ситуаціях.

Останні досягнення в галузі інтелектуальних інформаційних технологій пов'язані перед усім з розвитком глибоких нейронних мереж. При цьому існуючі моделі нейронних мереж, що вирішують задачі детектування та геолокалізації об'єктів інтересу, для успішного навчання потребують великі обсяги розмічених навчальних зразків та обчислювальних ресурсів. Проте для переважної більшості задач не створено відповідного навчального набору даних, а існуючі нейромережеві моделі аналізу даних не придатні для реалізації оперативного перенавчання за умов обмеженого обсягу ресурсів БПЛА.

Інтелектуальні моделі аналізу даних умовно можна поділити на дві частини – екстрактор ознакового опису спостережень та вирішальні правила. При цьому від ефективності екстрактора ознакового опису напряму залежить ефективність вирішальних правил. Зокрема, від складності моделі і алгоритмів функціонування екстрактора ознак залежить оперативність оптимізації параметрів всієї моделі інтелектуального аналізу сенсорних даних бортової системи. Тому інформативність ознакового подання та обчислювальна складність алгоритмів функціонування екстрактора ознак є ключовими під час проектування моделі



аналізу даних бортової системи БПЛА.

Науково-дослідна робота, що виконується на кафедрі комп'ютерних наук Сумського державного університету, спрямована на розв'язання важливої науково-практичної задачі підвищення функціональної ефективності бортової системи БПЛА, що здійснює у автономному режимі навігацію та ідентифікацію об'єктів інтересу за умов ресурсних та інформаційних обмежень, і неповної визначеності, обумовленої довільними умовами спостереження та нестационарністю середовища. Зміст роботи на даному етапі полягає у розробленні інформаційного та програмного забезпечення бортової системи БПЛА, що функціонує в режимі самонавчання екстрактора ознакового опису середовища для навігації та класифікаційного аналізу спостережень.

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

Перший розділ присвячено аналізу об'єкту, обґрунтуванню напрямку дослідження і формалізованій постановці задачі та синтезу моделі екстракції ознакового опису для автономної навігації та ідентифікації об'єктів на місцевості.

У другому розділі виконано опис алгоритмів оптимізації гіперпараметрів та машинного навчання моделі інтелектуального аналізу аеро- відео- спостережень бортової системи БПЛА. Також виконано аналіз результатів фізичного моделювання і підтверджено придатність розроблених моделей та алгоритмів для практичного використання.

Результати наукових досліджень, одержаних виконавцями проекту, опубліковано в працях [1]–[12].

# 1 МОДЕЛІ ЕКСТРАКЦІЇ ОЗНАКОВОГО ОПИСУ ДЛЯ КЛАСИФІКАЦІЙНОГО АНАЛІЗУ СПОСТЕРЕЖЕНЬ ТА НАВІГАЦІЇ

## 1.1 Аналіз об'єкту та обґрунтування напрямку дослідження

Зниження ресурсних потреб бортової системи БПЛА сприяє підвищенню рівня його автономності і дозволяє зарезервувати ресурс для розширення функціональних можливостей БПЛА. При цьому питання підвищення функціональної ефективності за умов ресурсних та інформаційних обмежень все ще залишаються недостатньо дослідженими через науково-методологічні ускладнення, пов'язані з неповною визначеністю даних.

Бортові відеокамери, в тому числі інфрачервоні та багатоспектральні, є однією з найбільш інформативних сенсорних систем БПЛА, однак збільшення кількості камер підвищує обчислювальне навантаження для обробки і аналізу відео потоку. При цьому зображення аеровідеозйомки характеризуються низькою інформаційною ємністю – тобто, більша частина растрового зображення (наприклад, місця однотонного або рівномірно розподіленого фону) не містять інформації, яка впливає на процес розпізнавання. Тому велика кількість виробників робототехніки віддають перевагу у використанні стереозору, який характеризується більшою інформаційною ємністю, оскільки здійснює реконструкцію глибини тривимірної сцени, що полегшує реалізацію як алгоритмів автономної навігації, так і алгоритмів розпізнавання об'єктів інтересу.

Ефективність стереозору залежить від відстані між камерами (база стереопари) та відстані до об'єктів і їх топології в полі зору камери. При цьому стереозір вироджується до монокулярного випадку коли відстань між камерами є набагато меншою ніж відстань до орієнтирів. Однак обробка серії послідовних кадрів, що формуються з необхідною частотою, від монокулярної камери також дозволяє реконструювати глибину тривимірної сцени [13], [14]. Тому, у загальному випадку, перспективним напрямком досліджень є підвищення ефективності використання монокулярної камери для задач розпізнавання та

геолокалізації об'єктів на місцевості.

Задача формування інформативного і компактного ознакового подання спостережень є визначальною для побудови ефективної моделі аналізу даних за умов ресурсних на інформаційних обмежень. При цьому можна виділити два підходи до проектування екстракторів ознак – однорівневий та ієрархічний (багатошаровий). Обидва підходи здійснюють подання фрагментів зображення (або його проміжного подання) кодами, які агрегуються у компактне подання за допомогою операторів пулінгу.

Для навчання кодера фрагментів зображення використовують як навчання без вчителя на основі автоенкодерів, алгоритмів кластер-аналізу та методів декомпозиції матриць, так і навчання з учителем з кінця-в-кінець на основі міток анованих зображень та звичайного алгоритму зворотнього поширення помилки. Для безпосереднього кодування як правило використовують такі методи : техніку м'якого призначення (Soft assignment), грубого призначення (Hard assignment), розрідженого кодування (Sparse coding, SC), ортогонального узгодженого переслідування (Orthogonal matching pursuit, OMP- $\lambda$ ), сигмоїдна функція (Sigmoid), лінійної ректифікації (Rectified Linear Unit, ReLU), триангуляція (Triangle) та м'який поріг (Soft threshold) [15].

Оператор пулінгу приймає на вхід декілька кодів, що відповідають сусіднім фрагментам спостереження з заданої області, які можливо перетинаються, і повертає узагальнений вектор фіксованої довжини. На практиці набули поширення використання оператора Average pooling, що обчислює середнє значення кодів по регіону, та оператора Max pooling, що обчислює максимум кожного компонента замість його середнього значення [16]. Можливі і інші реалізації оператора пулінгу для врахування структури просторових чи часових залежностей в окремому регіоні спостереження. Оператори пулінгу здійснюють зниження розмірності та варіативності даних, забезпечуючи стійкість до трансляцій локальної ознаки в межах окремого регіону. Тобто факт наявності ознаки важливіший знань місця його точного розміщення на зображенні. Таким чином модель забезпечує інваріантність до просторових викривлень.

В однорівневих екстракторах ознак для опису візуальних спостережень значного поширення набуло використання локальних ознак (ознаки Хаара, гістограми орієнтованих градієнтів, локальні бінарні шаблони) та різноманітні методи їх компактного подання [17], [18]. Використання локальних ознак дозволяє ефективно здійснювати класифікаційний аналіз та співставлення зображень або їх частин, але лише в квазістаціонарних умовах спостереження і за обмеженої варіативності даних. За умов великої варіації спостережень неієрархічні моделі для забезпечення прийнятної достовірності результатів потребують значні обсяги ресурсів. При цьому ці моделі, як правило, мають велику кількість гіперпараметрів і низьку узагальнюючу здатність, оскільки ігнорують значну кількість глобальної контекстної інформації.

Крім розпізнавання об'єктів на зображенні місцевості для їх геолокалізації потрібно мати геокоординати самого безпілотного апарату, що здійснює спостереження. Для реалізації автономної геолокалізації набули широкого використання різноманітні методи візуально-інерційної навігації в рамках геометричного підходу [19], [20]. Ці методи основані на екстракції різноманітних локальних орієнтирів і тому можуть бути успішними у випадку достатнього перекриття кадрів, квазістаціонарності умов спостереження, текстурованості спостережень і високій точності калібрування сенсорів. Тому слабким місцем даних методів є висока чутливість як до параметрів камери і калібрування сенсорів так і до варіації умов спостереження (ділянки збіднені ознаками, розмиття рухом, зміни освітлення, рухомі об'єкти в полі зору тощо).

Найбільшого успіху в задачах аналізу візуальних образів досягли методи глибоких нейронних мереж. При цьому найбільш універсальними серед них є згорткові нейронні мережі, оскільки вони здатні аналізувати як просторові так і короткострокові часові шаблони. Для підвищення інформативності ознакового опису до архітектури додають різноманітні залишкові зв'язки, використовують фільтри з різними ядрами для формування спільної карти ознак і тому подібне. При цьому для підвищення оперативності машинного навчання використовується техніка переносу знань та різноманітні мікро-архітектурні рішення для

декомпозиції великих згорток на серію менших [21], [22].

Глибокі нейронні мережі знайшли своє застосування як в задачах детектування об'єктів на зображенні, так і в задачах автономної візуальної навігації, зокрема одометрії [23]–[25]. У задачах детектування об'єктів інтересу на зображенні здійснюється класифікаційний та регресійний аналіз пікселів різномасштабних карт ознак, що дозволяє детектувати і визначати межі об'єктів різного розміру за умов високої варіації спостережень. У задачах візуальної одометрії здатність до навчання і детального сприйняття образів дозволяє знизити чутливість до параметрів камери і складних умов спостереження [24]–[26]. Проте традиційні алгоритми зворотного поширення помилки для реалізації успішного навчання глибоких нейронних мереж потребують значного обсягу розмічених навчальних даних та обчислювальних ресурсів. Крім того вибір кількості згорткових фільтрів на кожному рівні, так само як і вибір кількості шарів, попередньо навчених згідно техніки переносу знань, є слабоформалізованими.

Зниження обчислювальної складності в режимі екзамену може бути досягнуте шляхом використання окремих “полегшених” моделей для конкретного середовища. Однак навіть “легкі” глибокі моделі для навчання або перенавчання потребують значні обсяги обчислювальних ресурсів та розмічених даних, що ускладнює реалізацію можливості швидкої адаптації до нових умов функціонування та/або задач.

Аналіз праць [15], [16] дозволяє сформулювати такі загальні принципи конструювання екстракторів ознакового опису спостережень для задач інтелектуального аналізу інформації та прийняття рішень :

- гладкість – навчена модель, що описується функцією  $f$ , передбачає виконання умови  $f(x) \approx f(y)$  при  $x \approx y$ ;
- множинні пояснюючі фактори – навчання моделі новому фактору призводить до його узагальнення в конфігураціях інших факторів;
- ієрархія в організації пояснюючих факторів – поняття, які використовуються для опису середовища, можуть бути визначені в термінах інших більш абстрактних понять, тобто ієрархічно;

- часткове навчання з учителем – подання даних, зручне для обчислення  $P(X)$ , є зручним і для обчислення  $P(Y|X)$ , що дозволяє встановити статистичний взаємозв'язок між машинним навчанням з учителем і без учителя;

- спільні фактори від задачі до задачі – у багатозадачних моделях ті задачі, що використовують спільні з іншими задачами пояснюючі фактори, використовують статистичний взаємозв'язок навчання між різними задачами;

- різноманіття – основна щільність ймовірності даних зосереджена біля регіонів, які мають набагато меншу розмірність, ніж оригінальний простір, де знаходяться дані;

- природна кластеризація – локальні зміни в різноманітті мають тенденцію до збереження значень самої категорії, а лінійна інтерполяція між зразками різних класів проходить у загальному випадку через регіони з низькою щільністю ймовірності;

- часова і просторова зв'язаність – спостереження, сформовані у сусідніх областях простору чи отримані послідовно в часі, повинні прагнути асоціюватися з однаковими значеннями відповідної категорії понять, чи приводити до невеликого руху по поверхні різноманіття високої щільності, (іншими словами повинне бути введено обмеження на зміну «крізь час і простір»);

- розрідженість – для будь-якого даного спостереження  $x$  тільки мала частина з усіх можливих факторів є значимою, тобто більша частина виділених ознак нечутлива до малих змін спостереження  $x$ , тому більша частина детектованих ознак повинна бути нульовою;

- простота залежності факторів – в хороших високорівневих поданнях даних фактори зв'язані один з одним через прості залежності.

У працях [15], [27] пропонується навчання згорткових мереж без вчителя на основі автоенкодера або обмеженої машини Больцмана, які потребують великий обсяг навчальних даних і тривалий час навчання для отримання прийняттого результату. У працях [28], [29] пропонується поєднання принципів нейронного газу та розрідженого кодування для навчання згорткових фільтрів за нерозміченими даними. Даний підхід має м'яку конкуруючу схему навчання, що

підвищує ймовірність збіжності алгоритму до оптимального розподілу нейронів на навчальних даних. При цьому впровадження алгоритмів розрідженого кодування дозволяє підвищити завадозахищеність і узагальнюючу здатність ознакового подання. Однак вибір кількості нейронів не формалізований та задається на розсуд розробника, або налаштовується шляхом оптимізації, що призводить до збільшення кількості ітерацій навчання.

Необхідну кількість згорткових фільтрів у високорівневих згорткових шарах наперед оцінити важко, тому перспективним підходом до навчання згорткових фільтрів є використання принципів зростаючого нейронного газу, який дозволяє автоматично визначити необхідну кількість нейронів. Однак механізм додавання нових нейронів в алгоритмі нейронного газу на основі заданого періоду вставки часто призводить до викривлення утворених структур і нестабільності процесу навчання. Проте у праці [30] було показано, що забезпечити стабільність навчання можна шляхом задавання «радіусу досяжності» нейронів. Це передбачає використання замість періоду вставки нейронів порогу максимального віддалення нейрону від кожної з віднесених до нього точок навчальної множини. Однак досі не було розглянуто механізми оновлення нейронів та оцінки віддаленості точок вхідного простору до нейронів з метою адаптації процесу навчання до процедури розрідженого кодування спостережень.

Ефективне ознакове подання дозволяє будувати прості, але точні вирішальні правила. Тому для оцінки ефективності екстрактора ознакового опису необхідно використовувати критерій, що враховує ефективність простих вирішальних правил та складність екстрактора ознакового опису. При цьому важливою умовою підвищення оперативності корекції ознакового опису є використання швидких алгоритмів навчання вирішальних правил, наприклад на основі ансамблю дерев рішень (випадковий ліс, адаптивний бустінг, модель випадкових підпросторів) чи рандомізованих алгоритмів пошуку рішення задачі найменших квадратів (алгоритм псевдо-інверсії Мура-Пенроуза, метод Грамма-Шмідта та інші).

У працях [31]–[33] було запропоновано так звану інформаційно-екстремальну інтелектуальну технологію (ІЕІ-технології) для класифікаційного аналізу високорівневого ознакового подання. Ця технологія основана на адаптивному двійковому кодуванні і побудові оптимальних в інформаційному розумінні радіально-базисних вирішальних правил в двійковому просторі Хеммінга. При цьому двійкове кодування спостережень може здійснюватися на основі ансамблю дерев рішень, що дозволяє адаптивно нарощувати модель до досягнення необхідної достовірності рішень [34]. У праці [35] було представлено регресійну модель на основі екстремальної машини навчання (Extreme Learning Machine), для якої характерна висока швидкість навчання. При цьому також було запропоновано застосування методу інкрементального навчання машини екстремального навчання шляхом послідовного додавання прихованих вузлів для уникнення проблеми перенавчання [35].

Як було показано у працях [36]–[37] перспективним шляхом вирішення задачі оптимізації параметрів моделі аналізу даних є застосування метаевристичних пошукових алгоритмів як альтернативи алгоритму зворотного поширення помилки. Серед таких алгоритмів варто виділити ройові алгоритми, такі як рій частинок, косяк риб та система заряджених частинок, та траєкторні, такі як симуляції відпалу чи сходження на пагорб, які характеризуються швидкою конвергенцією та малою ймовірністю застрягання в «поганому» локальному оптимумі [36]. Проте, використання подібних алгоритмів в задачах тонкого настроювання параметрів екстрактора ознак все ще залишається не достатньо дослідженим.

У працях [37], [38] було запропоновано використовувати категорійно-функтурні моделі у вигляді направлених графів відображення множин, задіяних в інтелектуальному аналізі даних. Категорійно-функтурне моделювання полегшує розробку, аналіз та реінжиніринг алгоритмів функціонування інтелектуальної моделі аналізу даних. Тому розробка категорійно-функтурних моделей аналізу даних бортовою системою БПЛА є важливим етапом розробки інформаційного забезпечення БПЛА.



Аналітичний огляд підходів до проектування моделі та алгоритму навчання екстрактора ознакового опису для задач автономної ідентифікації та локалізації об'єктів на місцевості показує, що для квазістаціонарних умов функціонування одними із найбільш перспективних підходів є удосконалення однорівневих екстракторів ознак, а для нестационарного середовища з високою варіативністю спостережень – використання багатосарових ієрархічних екстракторів ознак. При цьому як критерій оптимізації параметрів екстрактора ознак слід розглядати комплексний критерій, що враховує обчислювальну складність екстрактора ознак та ефективність вирішальних правил. Крім того, розробку алгоритмів навчання перспективно здійснювати з використанням ідей і методів інформаційно-екстремальної інтелектуальної технології, машини екстремального навчання, нейронного газу та зростаючого нейронного газу, розрідженого кодування та метаевристичних алгоритмів пошукової оптимізації.

Обраний напрямок досліджень потребує розв'язання низки науково-методологічних задач, пов'язаних з аналізом і синтезом здатних навчатися систем ідентифікації об'єктів на місцевості, таких як :

- 1) розробка категорійно-функтурних моделей функціонування бортової системи БПЛА в режимі навчання екстрактора ознак та аналізу даних;
- 2) розробка моделей екстрактора ознак, здатного враховувати як локальну, так і контекстну інформацію спостережень;
- 3) розробка алгоритму машинного навчання екстрактора ознак, здатного ефективно використовувати як розмічені, так і нерозмічені навчальні зразки;
- 4) розробка критерію функціональної ефективності машинного навчання екстрактора ознак для задач детектування об'єктів та навігації;
- 5) оптимізація параметрів функціонування розроблених моделей аналізу даних.

## **1.2 Формалізована постановка задачі синтезу екстрактора ознак**

Нехай  $D_{\text{train}}^{\text{det}} = \{ I_k, B_k \mid k = \overline{1, K_1} \}$  та  $D_{\text{test}}^{\text{det}} = \{ I_k, B_k \mid k = \overline{1, K_2} \}$  є набором навчальних та тестових даних, де  $I_k$  –  $k$ -те зображення з кольоровою схемою

RGB з уніфікованою роздільною здатністю,  $V_k$  – набір правильно заданих обмежувальних рамок, які відповідають об’єктам інтересу на  $k$ -му зображенні,  $K_1, K_2$  – обсяги навчального та тестового наборів даних відповідно. Нехай алфавіт класів розпізнавання  $\{X_z^o | z = \overline{1, Z}\}$  характеризує об’єкти інтересу на місцевості.

Нехай  $D_{\text{train}}^{\text{odom}} = \{I_k, t_k, \theta_k | k = \overline{1, K_3}\}$  та  $D_{\text{test}}^{\text{odom}} = \{I_k, t_k, \theta_k | k = \overline{1, K_4}\}$  є набором навчальних та тестових даних, де  $I_k$  –  $k$ -те зображення з уніфікованою роздільною здатністю,  $t_k$  – вектор переміщення системи координат камери між кадрами зображення  $I_{k-1}$  та  $I_k$ ,  $\theta_k$  – вектор повороту системи координат камери між кадрами зображення  $I_{k-1}$  та  $I_k$  відповідно,  $K_3, K_4$  – обсяги навчального та тестового наборів даних відповідно. Також задано алфавіт класів розпізнавання  $\{X_u^o | u = \overline{1, U}\}$ , що характеризує характерні перешкоди і необхідність повороту на фіксований кут для уникнення зіткнення.

Дано структурований вектор просторово-часових параметрів функціонування інтелектуальної бортової системи БПЛА, який у загальному випадку має структуру

$$g = \langle e_1, \dots, e_{\xi_1}, \dots, e_{\Xi_1}, f_1, \dots, f_{\xi_2}, \dots, f_{\Xi_2} \rangle, \quad (1.1)$$

$$\Xi_1 + \Xi_2 = \Xi$$

де  $e_1, \dots, e_{\xi_1}, \dots, e_{\Xi_1}$  – генотипні параметри функціонування, які прямо впливають на параметри алгоритмів екстракції ознакового опису спостережень;

$f_1, \dots, f_{\xi_2}, \dots, f_{\Xi_2}$  – фенотипні параметри функціонування, які прямо впливають на ефективність вирішальних правил моделі аналізу даних.

При цьому відомі обмеження на параметри функціонування :

$$\begin{aligned} R_{\xi_1}(e_1, \dots, e_{\xi_1}, \dots, e_{\Xi_1}) &\leq 0, \\ R_{\xi_2}(f_1, \dots, f_{\xi_2}, \dots, f_{\Xi_2}) &\leq 0 \end{aligned} \quad (1.2)$$

Необхідно в процесі машинного навчання інтелектуальної моделі аналізу даних на борту автономного БПЛА визначити оптимальний вектор параметрів  $g$ , що забезпечує максимум комплексного критерію ефективності

$$J = F(J_{Cls}, J_{Reg}, J_{Complexity}) \quad (1.3)$$

$$g^* = \arg \max_G \{J(g)\}, \quad (1.4)$$

де  $F(\dots)$  – функція-агрегатор, що здійснює згортку часткових критеріїв в один комплексний критерій (може мати адитивну, мультиплікативну та мультиплікативно-адитивну природу);

$J_{Cls}$  – нормований інформаційний критерій функціональної ефективності класифікаційних вирішальних правил;

$J_{Reg}$  – нормований критерій точності регресійних вирішальних правил;

$J_{Complexity}$  – нормований критерій трудомісткості моделі аналізу даних;

$G$  – допустима область значень параметрів, які впливають на екстракцію ознак і прийняття рішень.

Метод інформаційного синтезу екстрактора ознакового опису спостережень бортовою системою БПЛА повинен відповідати таким основним вимогам :

- прямо та об’єктивно характеризувати функціональну ефективність екстрактора ознакового опису;
- враховувати обчислювальну складність екстрактора ознак;
- оптимізувати просторово-часові параметри функціонування екстрактора ознакового опису спостережень з метою побудови вирішальних правил, які забезпечує на экзамені повну достовірність рішень, наближену до максимальної асимптотичної, що визначається ефективністю навчання системи;
- забезпечувати високу функціональну ефективність навчання екстрактора

ознакового опису для задач автономної навігації та ідентифікації об'єктів на місцевості за умов нестаціонарності середовища та мінливості об'єктів інтересу, що має місце на практиці;

– реалізувати принцип навчання з частковим залученням вчителя та переносу знань для ефективного використання нерозмічених навчальних зразків та знань, акумульованих в навчених нейронних мережах.

Таким чином, задача інформаційного синтезу моделі інтелектуального аналізу аеро- відео- спостережень для автономної бортової системи БПЛА полягає в оптимізації за комплексним критерієм ефективності параметрів функціонування системи, які впливають на її ефективність.

### 1.3 Категорійно-функторні моделі процесу навчання бортової системи аналізу спостережень

У загальному випадку математичну (категорійну) модель будь-якої інтелектуальної моделі аналізу даних можна подати у вигляді діаграми відображень множин, що задіяні в процесі трансформації даних та прийняття рішень (рис. 1.1).

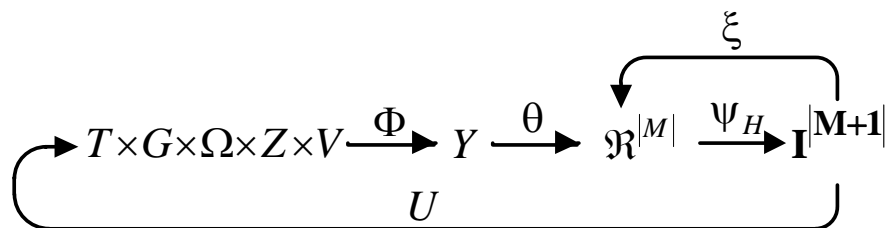


Рисунок 1.1 – Узагальнена категорійна модель інтелектуальної моделі аналізу даних

На рис. 1.1 прийнято такі позначення:

T – множина моментів часу зняття інформації;

G – простір вхідних сигналів (факторів), які діють на систему;

$\Omega$  – простір ознак розпізнавання;

$Z$  – простір можливих функціональних станів, у яких перебуває система;

$V$  – множина типів вирішальних правил;

$Y$  – вибіркова множина (вхідна навчальна матриця);

$\Phi: G \times T \times \Omega \times Z \times V \rightarrow Y$  – оператор формування вибіркової навчальної множини  $Y$ ;

$\theta$  – оператор відображення вибіркової множини  $Y$  в простір вторинних ознак, де сформовано  $M$  вирішальних правил  $\tilde{\mathfrak{R}}^{|M|}$  ( $\tilde{\mathfrak{R}}^{|M|} \subset \Omega$ );

$\psi$  – оператор класифікації або регресії;

$I^{|M+1|}$  – множина допустимих гіпотез (рішень), де  $\gamma_{M+1}$  – гіпотеза про відмову прийняття рішення;

$\xi$  – оператор зворотного зв'язку;

$U: I^{|M+1|} \rightarrow T \times G \times \Omega \times Z \times V$  – оператор регламентації процесу функціонування інтелектуальної системи аналізу сенсорних даних.

На рис. 1.1 оператор зворотного зв'язку  $\xi$  корегує геометричні параметри вирішальних правил  $\tilde{\mathfrak{R}}^{|M|}$  з метою покращення точнісних характеристик моделі аналізу даних і саме спосіб реалізації цього оператора задається різними методами та підходами до машинного навчання.

В задачах детектування об'єктів аерозображеннях місцевості існує необхідність класифікаційного аналізу ознакового подання ділянок зображення. Так само і в задачі автономної навігації існує необхідність класифікаційного аналізу перешкод для прогнозування повороту. За умов обмеженої варіації спостережень, що буває у випадку апіорно відомої місцевості, діапазону висот та задач, може бути виправданим використання однорівневого екстрактора ознак. Однак для забезпечення необхідної достовірності класифікаційних рішень необхідно оптимізувати параметри екстрактора в інформаційному та вартісному сенсах. На рис. 1.2 показано категорійну модель процесу навчання однорівневого екстрактора ознак для класифікаційного аналізу спостережень.

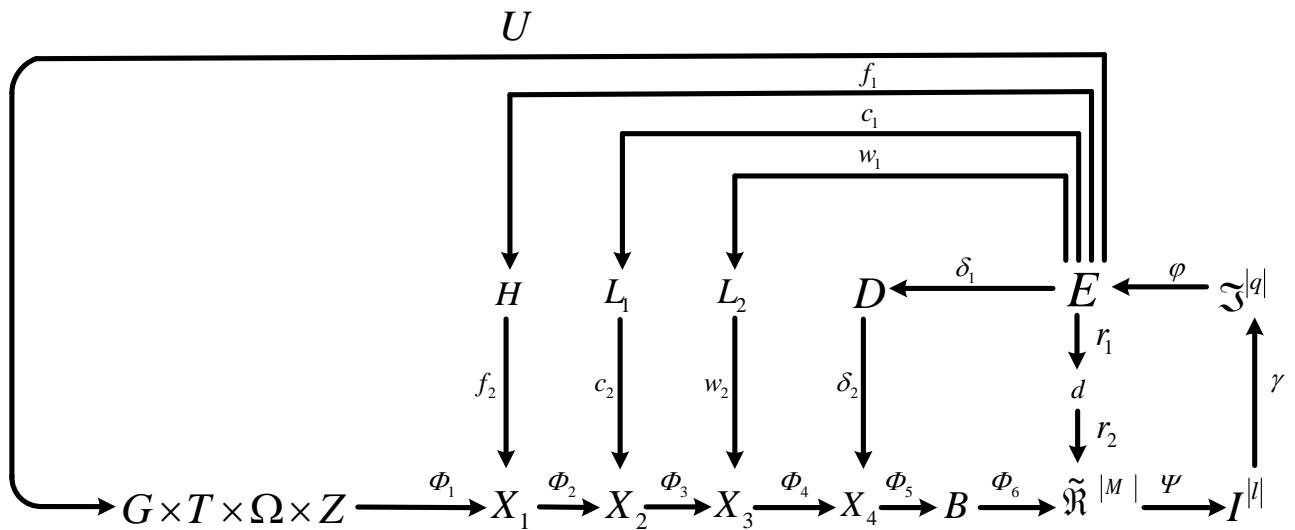


Рисунок 1.2 – Категорійна модель навчання однорівневого екстрактора ознак для класифікаційного аналізу даних

У діаграмі (рис. 1.2) показано оператор  $\Phi_2: X_1 \rightarrow X_2$ , який здійснює екстракцію локальних ознак і доповнює зображення координатами та масштабом ключових фрагментів. При цьому композиція операторів  $f_1 \circ f_2$  регламентує оптимізацію порогу, що відповідає за врахування чи відкидання локальних ознак. Оператор  $\Phi_3: X_2 \rightarrow X_3$  здійснює векторне квантування дескрипторів інформативних фрагментів зображень для формування словника ключових фрагментів зображень (візуальних слів). Композиція операторів  $c_1 \circ c_2$  реалізує контур оптимізації потужності словника візуальних слів. Оператор  $\Phi_4: X_3 \rightarrow X_4$  здійснює кодування зображень або їх ділянок гістограмами візуальних слів та фраз (комбінація сусідніх візуальних слів). При цьому композиція операторів  $w_1 \circ w_2$  реалізує контур оптимізації розмірності гістограми частот появи візуальних слів та фраз.

При обґрунтуванні гіпотези нечіткої компактності має місце нечітке розбиття  $\tilde{\mathfrak{R}}^{|M|} \subset \Omega$ . При цьому оператор  $\Phi_5: X_4 \rightarrow B$  здійснює відображення вибіркової множини в бінарний простір ознак розпізнавання, а оператор  $\Phi_6: B \rightarrow \tilde{\mathfrak{R}}^{|M|}$  відновлює на кожному кроці навчання оптимальне в

інформаційному розумінні розбиття на  $M$  класів. Оператор класифікації  $\psi: \tilde{\mathfrak{R}}^{|M|} \rightarrow I^{|I|}$  перевіряє основну статистичну гіпотезу про належність реалізацій  $\{b_m^{(j)} | j=1, \overline{n}\}$  класу  $X_m^o$ . Оператор  $\gamma: I^{|I|} \rightarrow \mathfrak{Z}^{|q|}$  шляхом оцінки статистичних гіпотез формує множину точнісних характеристик  $\mathfrak{Z}^{|q|}$ , де  $q=l^2$  – кількість точнісних характеристик. Оператор  $\varphi: \mathfrak{Z}^{|q|} \rightarrow E$  обчислює множину значень інформаційного критерію, який є функціоналом точнісних характеристик. Контур оптимізації геометричних параметрів розбиття  $\tilde{\mathfrak{R}}^{|M|}$  шляхом пошуку максимуму інформаційного критерію навчання розпізнаванню реалізацій класу  $X_m^o$  замикається оператором  $r_1 \circ r_2: E \rightarrow \tilde{\mathfrak{R}}^{|M|}$ . Відповідна функціональна схема алгоритму навчання показана на рис. 1.3.

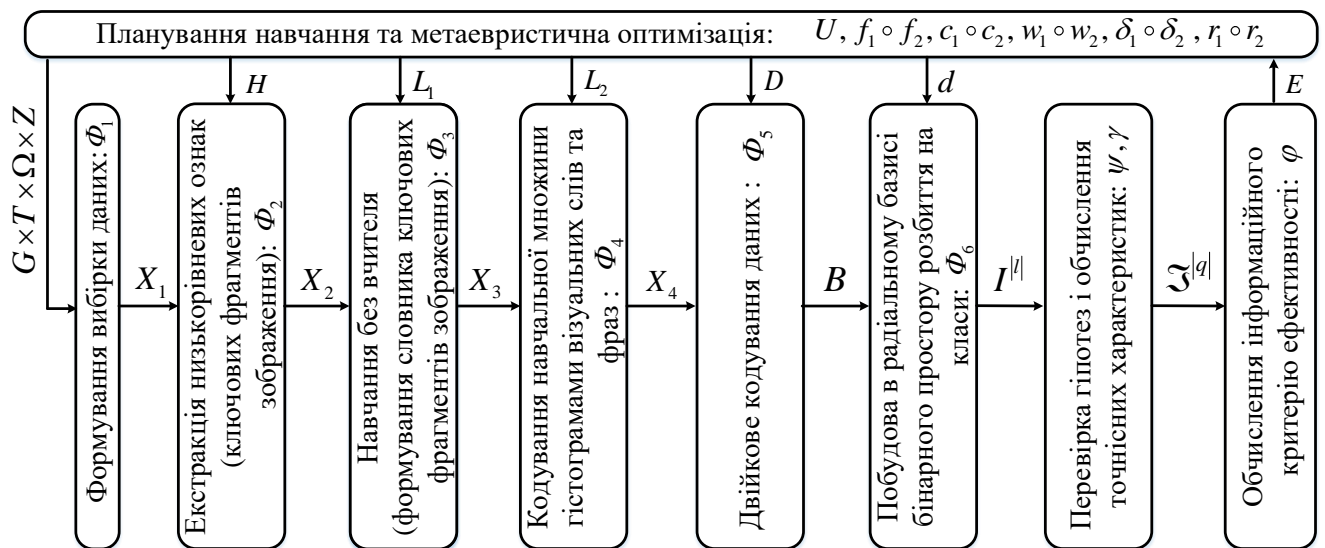


Рисунок 1.3 – Функціональна схема алгоритму навчання однорівневого екстрактора ознак для класифікаційного аналізу даних

За умов великої варіації спостережень кодування зображень слід здійснювати з використанням ієрархічного ознакового подання, оскільки воно характеризується вищою ємністю і інформативністю при меншій кількості параметрів. На рис. 1.4 показано категорійну модель процесу навчання ієрархічного екстрактора ознак для класифікаційного аналізу спостережень.

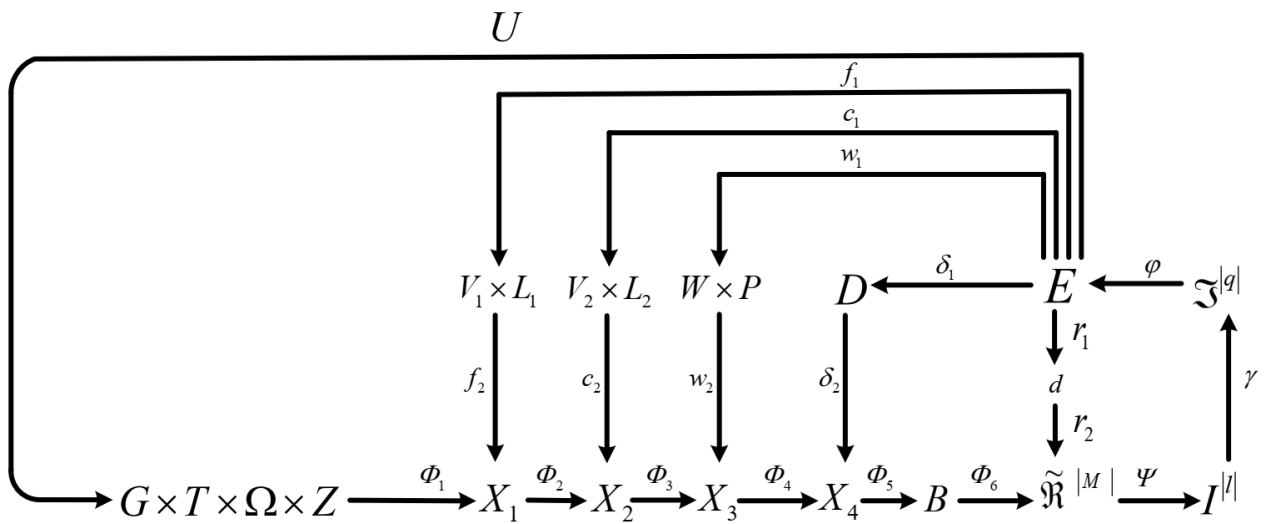


Рисунок 1.4 – Категорійна модель навчання ієрархічного екстрактора ознак для класифікаційного аналізу даних

На рис. 1.4 прийнято такі позначення :

$G$  – множина факторів, що впливають на систему спостереження і середовище;

$T$  – множина моментів зняття інформації;

$Z$  – множина об'єктів спостереження, що зустрічаються в полі зору;

$\Omega$  – множина ознак, що зчитується;

$X_1$  – вибірка необроблених спостережень;

$X_2$  – спостереження, закодовані картами ознак із запозичених шарів навченої з учителем згорткової мережі на датасеті ImageNet;

$X_3$  – спостереження, закодовані картами ознак, навченими без вчителя на нерозміченому наборі даних з доменної області;

$B$  – спостереження, закодовані двійковим кодом в результаті інформаційно-екстремального навчання;

$\mathfrak{R}^{|M|}$  – розбиття на  $M$  класів у двійкового просторі Хеммінга;

$I^{|l|}$  – множина гіпотез при  $l$ -альтернативному рішенні;

$\mathfrak{S}^{|q|}$  – множина точнісних характеристик рішень;

$E$  – множина значень інформаційного критерію при різних значеннях



параметрів функціонування;

$V_1$  – множина навчених моделей згорткових нейронних мереж на наборі даних ImageNet;

$L_1$  – множина масок на верхні шари навчених нейронних мереж;

$V_2$  – множина значень параметра, що впливає на точність реконструкції даних в алгоритмі навчання без вчителя;

$L_2$  – множина значень кількості шарів, що навчаються без вчителя;

$D$  – множина впорядкованих порогових значень на ознаки розпізнавання;

$d$  – множина значень кодових радіусів гіперсферичних контейнерів класів розпізнавання.

Відповідна функціональна схема алгоритму навчання ієрархічного екстрактора ознак для класифікаційного аналізу спостережень показана на рис. 1.5.

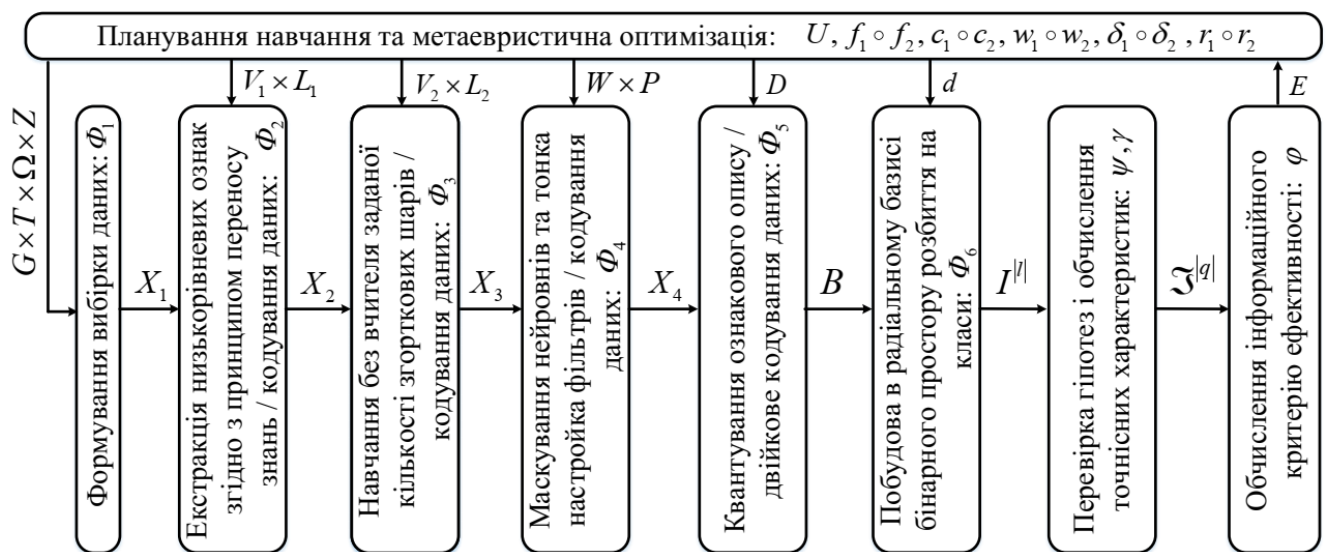


Рисунок 1.5 – Функціональна схема алгоритму навчання ієрархічного екстрактора ознак для класифікаційного аналізу даних

Діаграму відображень множин при функціонуванні моделі аналізу даних в робочому режимі (режим екзамену) показано на рис. 1.6.

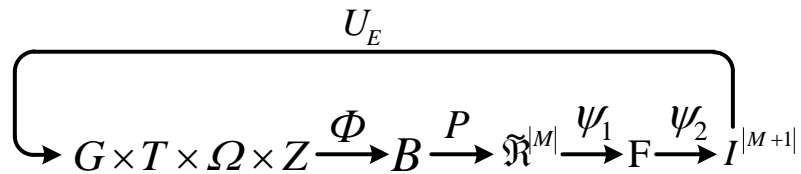


Рисунок 1.6 – Категорійна модель функціонування моделі класифікаційного аналізу даних в режимі екзамену

У діаграмі оператор  $\Phi = \Phi_1 \circ \Phi_2 \circ \Phi_3 \circ \Phi_4 \circ \Phi_5$  відображає універсум випробувань на вибіркову множину  $B$ , яка утворює бінарну екзаменаційну матрицю  $\|b_i^{(j)}\|_{i=\overline{1,N}, j=\overline{1,n}}$ , аналогічну за структурою та параметрами формування навчальній матриці. Оператор  $P$  відображає вибіркову множину  $X^o \subset \Omega_B$ , що розпізнається, на побудоване на етапі навчання розбиття  $\tilde{\mathfrak{R}}^{|M|}$ , а оператор класифікації  $\psi$  утворює композицію двох операторів:  $\psi_1: \tilde{\mathfrak{R}}^{|M|} \rightarrow F$ , де  $F$  – множина функцій належності, і оператор  $\psi_2: F \rightarrow I^{|M+1|}$ , який вибирає гіпотезу за максимальним значенням функції належності.

Суттєвою перевагою категорійних моделей у вигляді діаграм відображень множин є те, що вони дозволяють на етапі системного аналізу не тільки встановлювати відношення між елементами інформаційного забезпечення та інформаційними потоками оброблення інформації, але і суттєво полегшують реінжиніринг, рефакторинг та інші етапи розробки інформаційного забезпечення БПЛА.

#### 1.4 Критерій функціональної ефективності моделі аналізу даних

Оскільки ефективність екстрактора ознакового опису визначається його обчислювальною складністю та ефективністю вирішальних правил, то оптимізаційний критерій повинен бути комплексним. При цьому для отримання екстрактора ознак, що одночасно забезпечує максимальні значення частинних критеріїв, як комплексний критерій ефективності навчання бортової системи

відповідно до (1.3) розглядається мультиплікативна згортка частинних критеріїв

$$J = J_{Cls} \cdot J_{Reg} \cdot J_{Complexity}.$$

Мультиплікативна згортка дозволяє зменшити ефект компенсації одного частинного критерію за рахунок іншого, що особливо важливо в умовах наявності попарної суперечливості і нелінійної залежності параметрів функціонування моделі. При цьому для задачі класифікаційного аналізу образів нормований інформаційний критерій функціональної ефективності  $J_{Cls}$  рівний

$$J_{Cls} = \frac{1}{E_{\max}} \frac{1}{Z} \sum_{z=1}^Z E_z, \quad (1.5)$$

де  $E_z$  – інформаційний критерій ефективності навчання класифікатора розпізнавати реалізації класу  $X_z^o$ ;

$E_{\max}$  – максимальне граничне значення інформаційного критерію ефективності навчання класифікатора.

Для загального випадку багатоальтернативного рішення, коли потужність алфавіту класів більша двох ( $Z > 2$ ), задачі оцінки функціональної ефективності та оптимізації параметрів мультикласових вирішальних правил найчастіше зводяться до відповідних задач серії двокласових вирішальних правил, оскільки це дозволяє суттєво підвищити оперативність навчання, не втрачаючи в достовірності результуючого класифікатора. Для реалізації зведення багатокласової оптимізаційної задачі до серії двокласових пропонується використовувати принцип «кожний проти всіх» – під час побудови вирішального правила для класу  $X_z^o$  кожна  $j$ -та реалізація цього класу вважається позитивною, а реалізації решти класів – негативними. При цьому для уникнення проблеми незбалансованості класів розпізнавання, обумовленої переважанням у навчальному наборі даних негативних зразків, здійснюється введення

синтетичного класу, який є альтернативним для  $X_z^o$ . Синтетичний клас представлено  $n_z$  векторами з інших класів, які найбільш близькі до опорного вектору  $b_z$ , де  $n_z$  – обсяг навчального набору класу  $X_z^o$ .

У рамках ІЕІ-технології під час вибору критерія ефективності двокласового класифікатора перевагу слід віддавати статистичним логарифмічним критеріям, оскільки вони, завдяки згладжуючому ефекту логарифмічної функції, забезпечують високу узагальнюючу здатність вирішальних правил, синтезованих за навчальною вибіркою відносно малого розміру. Серед таких критеріїв найбільшого використання знайшли ентропійна міра К. Шеннона та інформаційна міра С. Кульбака [1], [3].

Спочатку розглянемо модифіковану в праці [5] інформаційну міру С. Кульбака для двоальтернативних рішень, яка обчислюється як добуток логарифмічного відношення правдоподібності у вигляді відношення обчисленої на  $k$ -му кроці машинного навчання повної ймовірності правильного прийняття рішень  $P_{t,z}^{(k)}$  до повної ймовірності помилкового прийняття рішень  $P_{f,z}^{(k)}$ , на міру відхилень відповідних ймовірностей. Для двоальтернативних гіпотез модифіковану міру Кульбака як критерій навчання системи розпізнавати реалізації класу  $X_z^o$  представимо у вигляді

$$E_z^{(k)} = \left[ P_{t,z}^{(k)} - P_{f,z}^{(k)} \right] \cdot \log_2 \frac{P_{t,z}^{(k)}}{P_{f,z}^{(k)}} =$$

$$= \{ p(\mu_z) [p(\gamma_1^{(k)} / \mu_z) - p(\gamma_2^{(k)} / \mu_z)] + p(\mu_c) [p(\gamma_2^{(k)} / \mu_c) - p(\gamma_1^{(k)} / \mu_c)] \} \times$$

$$\times \log_2 \frac{p(\mu_z) p(\gamma_1^{(k)} / \mu_z) + p(\mu_c) p(\gamma_2^{(k)} / \mu_c)}{p(\mu_z) p(\gamma_2^{(k)} / \mu_z) + p(\mu_c) p(\gamma_1^{(k)} / \mu_c)},$$

де  $P_{t,z}^{(k)}$  – повна ймовірність правильного прийняття рішень про належність реалізацій класу  $X_z^o$  та сусіднього йому класу  $X_c^o$  до контейнера класу  $X_z^o$ ,

побудованому на  $k$ -му кроці навчання діагностичної системи розпізнавати реалізації класу  $X_z^o$ ;

$P_{f,z}^{(k)}$  – повна ймовірність помилкового прийняття рішень;

$\gamma_1^{(k)}$  – апіорна основна гіпотеза про належність до контейнера класу  $X_z^o$  своїх реалізацій;

$\gamma_2^{(k)}$  – апіорна альтернативна гіпотези про належність до контейнера класу  $X_z^o$  реалізацій класу  $X_c^o$ ;

$\mu_z$  – апостеріорна гіпотеза про належність до контейнера класу  $X_z^o$  своєї реалізації;

$\mu_c$  – апостеріорна гіпотеза про належність до контейнера класу  $X_z^o$  реалізації класу  $X_c^o$ ;

$p(\mu_z)$  – безумовна ймовірність прийняття гіпотези  $\mu_z$ ;

$p(\mu_c)$  – безумовна ймовірність прийняття гіпотези  $\mu_c$ ;

$p(\gamma_1^{(k)} / \mu_z)$  – перша достовірність  $D_{1,z}^{(k)}$ , яка є апіорною умовною ймовірністю прийняття гіпотези  $\gamma_1^{(k)}$  за умови, що має місце апостеріорна гіпотеза  $\mu_z$ ;

$p(\gamma_2^{(k)} / \mu_c)$  – друга достовірність  $D_{2,z}^{(k)}$ ;

$p(\gamma_2^{(k)} / \mu_z)$  – помилка першого роду  $\alpha_z^{(k)}$ ;

$p(\gamma_1^{(k)} / \mu_c)$  – помилка другого роду  $\beta_z^{(k)}$ .

Згідно з принципом Бернуллі-Лапласа, виправдано прийняття рівно-ймовірних гіпотез  $p(\mu_z) = p(\mu_c) = 0,5$ , оскільки при оцінці ефективності функціонування системи розпізнавання, що навчається, апіорна інформація є неповною. Реалізація цього принципу вимагає прийняття рішень діагностичною системою за найгіршими у статистичному розумінні умовами її функціонування. Тому гарантується, що покращення умов функціонування системи не знизить її

функціональну ефективність, а навпаки підвищить. З урахуванням прийнятих допущень остаточною формулою модифікації критерію Кульбака для рівномірних двоальтернативних гіпотез має вигляд

$$\begin{aligned}
 E_z^{(k)} &= 0,5 \cdot [(D_{1,z}^{(k)} + D_{2,z}^{(k)}) - (\alpha_z^{(k)} + \beta_z^{(k)})] \cdot \log_2 \left( \frac{D_{1,z}^{(k)} + D_{2,z}^{(k)}}{\alpha_z^{(k)} + \beta_z^{(k)}} \right) = \\
 &= [D_{1,z}^{(k)} - \beta_z^{(k)}] \cdot \log_2 \left( \frac{1 - [D_{1,z}^{(k)} - \beta_z^{(k)}]}{1 + [D_{1,z}^{(k)} - \beta_z^{(k)}]} \right), \quad (1.6)
 \end{aligned}$$

де  $\alpha_z$ ,  $\beta_z$  – частота хибно негативних (False negative rate) та частота хибно позитивних (False positive rate) спрацювань вирішальних правил класу  $X_z^0$  відповідно;  $D_{1,z}$ ,  $D_{2,z}$  – перша достовірність (Sensitivity або Recall) та друга достовірність (Specificity) вирішальних правил класу  $X_z^0$  відповідно.

Нормовану модифікацію критерія (1.6) можна подати у вигляді [7]

$$E_z = \frac{1 - (\alpha_z + \beta_z)}{\log_2(2 + \varsigma) - \log_2 \varsigma} \cdot \log_2 \left[ \frac{2 - (\alpha_z + \beta_z) + \varsigma}{(\alpha_z + \beta_z) + \varsigma} \right], \quad (1.7)$$

де  $\varsigma$  – будь-яке мале знакододатне число, яке вводиться, для уникнення невизначеності при діленні на нуль.

Навчальна вибірка є обмеженою за обсягом, тому оцінка точнісних характеристик здійснюють за відповідними емпіричними частотами

$$D_{1,z}^{(k)} = \frac{K_{1,z}^{(k)}}{n_z}; \quad \beta_z^{(k)} = \frac{K_{2,z}^{(k)}}{n_z}, \quad (1.8)$$

де  $K_{1,z}^{(k)}$  – кількість подій, що характеризують належність реалізацій класу  $X_z^o$  до контейнера класу  $X_z^o$  на  $k$ -му кроці машинного навчання;

$K_{2,z}^{(k)}$  – кількість подій, що характеризують належність реалізацій класу  $X_c^o$  до контейнера класу  $X_z^o$  на  $k$ -му кроці машинного навчання;

$n_z$  – обсяг навчальної вибірки класу  $X_z^o$ .

При цьому  $n_z \geq n_{\min}$ , де  $n_{\min}$  – мінімальний обсяг репрезентативної навчальної вибірки, який визначається згідно з працею [31].

Робоча формула модифікації інформаційного критерію за Кульбаком після відповідної підстановки оцінок (1.8) у вираз (1.6) набуває вигляду

$$E_z^{(k)} = \frac{1}{n_z} [K_{1,z}^{(k)} - K_{2,z}^{(k)}] \cdot \log_2 \left[ \frac{\zeta + n_z + [K_{1,z}^{(k)} - K_{2,z}^{(k)}]}{\zeta + n_z - [K_{1,z}^{(k)} - K_{2,z}^{(k)}]} \right], \quad (1.9)$$

Критерій ефективності регресійного аналізу меж обмежувального прямокутника детектованого на зображенні об'єкта обчислюється за формулою

$$J_{Reg} = \frac{1}{\hat{n}} \sum_{i=1}^{\hat{n}} IoU_i, \quad (1.10)$$

де  $IoU_i$  – міра перетину (коефіцієнт Жаккарда) реальної рамки об'єкту інтересу з відповідною  $i$ -ю прогнозованою рамкою об'єкта інтересу.

Критерій ефективності регресійного аналізу в задачі візуальної одометрії визначається як відношення мінімально досягнутої (в тому числі іншими дослідниками) усередненої за відеопослідовністю середньоквадратичної помилки реконструкції траєкторії руху  $\varepsilon_{\min}$  до фактично виміряного її значення  $\varepsilon$ , тобто

$$J_{Reg} = \frac{\varepsilon_{\min}}{\varepsilon}. \quad (1.11)$$

Критерій обчислювальної ефективності розглядається як відношення мінімально можливої трудомісткості моделі (або її найбільш трудомісткої частини моделі)  $C_{\min}$  до фактичної трудомісткості  $C$ , тобто

$$J_{Complexity} = \frac{C_{\min}}{C}, \quad (1.12)$$

У нейромережових моделях трудомісткість  $C$  прийнято вимірювати кількістю операцій множення (Mul) та додавання (Add), що виконуються під час прямого проходження виходу. У однорівневих екстракторах ознак, основаних на побудові гістограм візуальних слів та фраз, трудомісткість може визначатися потужністю словника візуальних слів.

Таким чином, критерій функціональної ефективності моделі аналізу даних має як інформаційну і точнісну, так і вартісну природу. При цьому для одночасного врахування частинних критеріїв комплексний критерій формується на основі мультиплікативної згортки.

### **1.5 Модель екстракції ознакового опису для задачі детектування об'єктів**

Під час проектування однорівневого екстрактора ознакового опису є вибір алгоритмів пошуку ключових фрагментів, що дозволяє порівнювати об'єкти за їх окремими найбільш помітними частинам. Помітність ділянки зображення може бути визначена за результатами обчислення локальної інформаційно-технічної ентропії за Шеноном [32], кратномасштабних вейвлет-перетворень [31] або оператора Гессе [17]. Для вирішення задачі пошуку ключових фрагментів пропонується використати метод SURF (Speeded Up Robust Features), заснований на обчисленні матриці Гессе  $H$ , детермінант якої (Гессіан)  $\det(H)$  досягає екстремуму в точках максимальної зміни градієнту яскравості пікселів [17]:



$$H(f(x, y)) = \begin{vmatrix} \frac{\partial^2 f}{\partial x^2} & \frac{\partial^2 f}{\partial x \partial y} \\ \frac{\partial^2 f}{\partial x \partial y} & \frac{\partial^2 f}{\partial y^2} \end{vmatrix}, \quad \det(H) = \frac{\partial^2 f}{\partial x^2} \frac{\partial^2 f}{\partial y^2} - \left[ \frac{\partial^2 f}{\partial x \partial y} \right]^2,$$

де  $f(x, y)$  – функція зміни градієнту яскравості.

Гессіан інваріантний відносно повороту, але не інваріантний до зміни масштабу, тому розрахунок градієнтів для кожної точки здійснюється при різних масштабах фільтру Хаара. Розмір фільтру береться рівним  $4s$ , де  $s$  – масштаб ключового фрагменту. Вигляд фільтрів Хаара показано на рис.1.7.

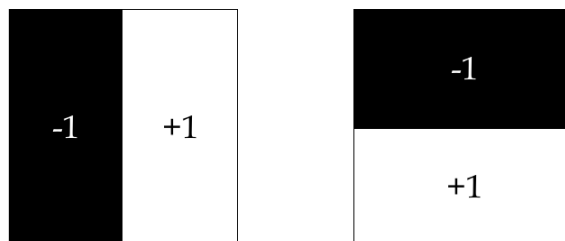


Рисунок 1.7 – Ядра фільтру Хаара

Для кожного знайденого ключового фрагменту метод SURF формує його дескриптор, що представляє собою набір з 64 (або 128 у розширеній версії SURF) чисел. Ці числа відображають флуктуації градієнту навколо центральної точки ключового фрагменту (рис. 1.8).

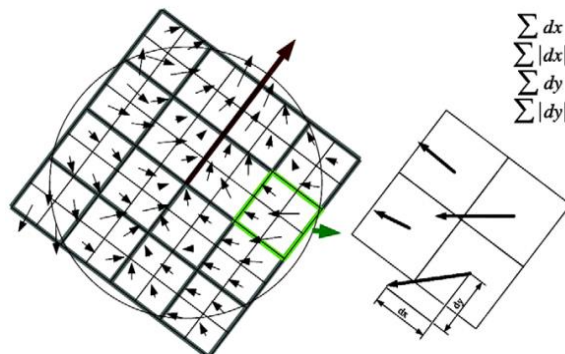


Рисунок 1.8 – Структура SURF-дескриптора

Оскільки в цій точці Гессіан досягає максимуму, то в її околі гарантується наявність ділянок з різним градієнтом, що забезпечує дисперсію дескрипторів для різних фрагментів і, відповідно, інваріантність дескрипторів відносно повороту. Розмір фрагменту для якого розраховується дескриптор, визначається масштабом матриці Гессе, що забезпечує інваріантність відносно масштабу.

Словник ключових фрагментів формується за результатами кластер-аналізу, в результаті якого схожі фрагменти об'єднуються в одну групу, яку представляє усереднений дескриптор. Побудова ознакового опису у вигляді глобального дескриптора зображення чи його ділянки здійснюється шляхом обчислення гістограми ключових слів. При цьому для врахування більшої кількості просторової інформації пропонується розширити його додатковими компонентами. Просторову інформацію можна отримати шляхом поділу вікна пошуку сіткою на квадратні комірки або полярною картою на комірки, що відновлюються в радіальному базисі, з наступним розрахунком гістограми частот появи пар сусідніх комірок вікна пошуку, у яких одночасно містяться ключові фрагменти, що відносяться до одного й того ж візуального слова (рис. 1.9).

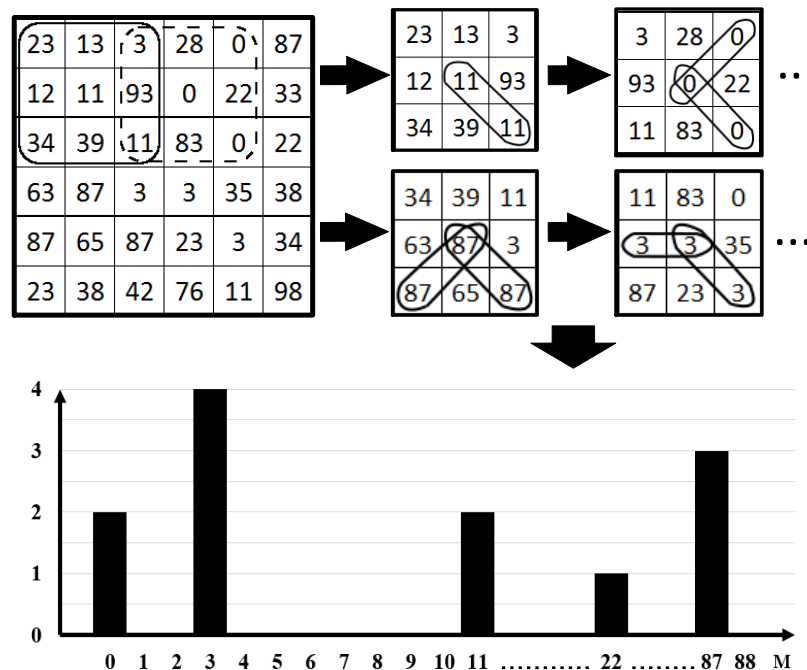


Рисунок 1.9 – Схема обчислення просторових ознак в однорівневому екстракторі

Побудова вирішальних правил в рамках так званої ІЕІ-технології [5] здійснюється згідно з принципом відкладених рішень в процесі оптимізації параметрів навчання за ієрархічною багатоциклічною ітераційною процедурою пошуку максимального граничного значення інформаційного критерію у вигляді (рис. 1.10)

$$g_{\xi}^* = \arg \left\langle \max_{G_{\Xi}} \left\{ \max_{G_{\Xi-1}} \left\{ \dots \left\{ \max_{G_1} \bar{E} \right\} \dots \right\} \right\} \right\rangle, \quad (1.13)$$

де  $g_{\xi}^*$  –  $\xi$ -й оптимальний в інформаційному розумінні параметр навчання системи ідентифікації об'єктів розпізнавати реалізації алфавіту класів  $\{ X_z^o \}$ ;

$G_{\Xi}$  – область допустимих значень  $\Xi$ -го параметру навчання;

$\bar{E}$  – усереднений за алфавітом класів інформаційний критерій.

Для детектування об'єкта інтересу з використанням однорівневого екстрактора ознак необхідно сформуванати список вікон, що є кандидатами на перевірку. Для кожного вікна здійснюється обчислення вектора ознак. Серед алгоритмів пошуку вікна-кандидата найбільшого поширення набули метод ковзного вікна з адаптивним кроком (англ. Run-Time Adaptive Sliding Window, RASW), метод ефективного пошуку підвікна (англ. Efficient sub-window search, ESS) та алгоритм селективного пошуку пропозицій вікна (Selective search).

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. Розмір кроку по двох осях може бути різний:  $dx$  і  $dy$ . Однак в простому випадку :  $d = dx = dy$ . Точність виявлення зменшується зі збільшенням кроку, але великий крок на «фоновій частині» зображення дозволяє зменшити кількість помилкових спрацьовувань. Зазвичай кроки дорівнюють одному або двом пікселям, оскільки більш високі значення призводять до погіршення якості виявлення. Алгоритм RASW має ще одну перевагу в порівнянні з класичним методом ковзного вікна зі статичним

кроком. При  $d > 1$  ковзне вікно може не потрапити в місце, де присутній шуканий об'єкт, а модифікований алгоритм зменшить крок до 1 пікселя і зупиниться в потрібному місці.

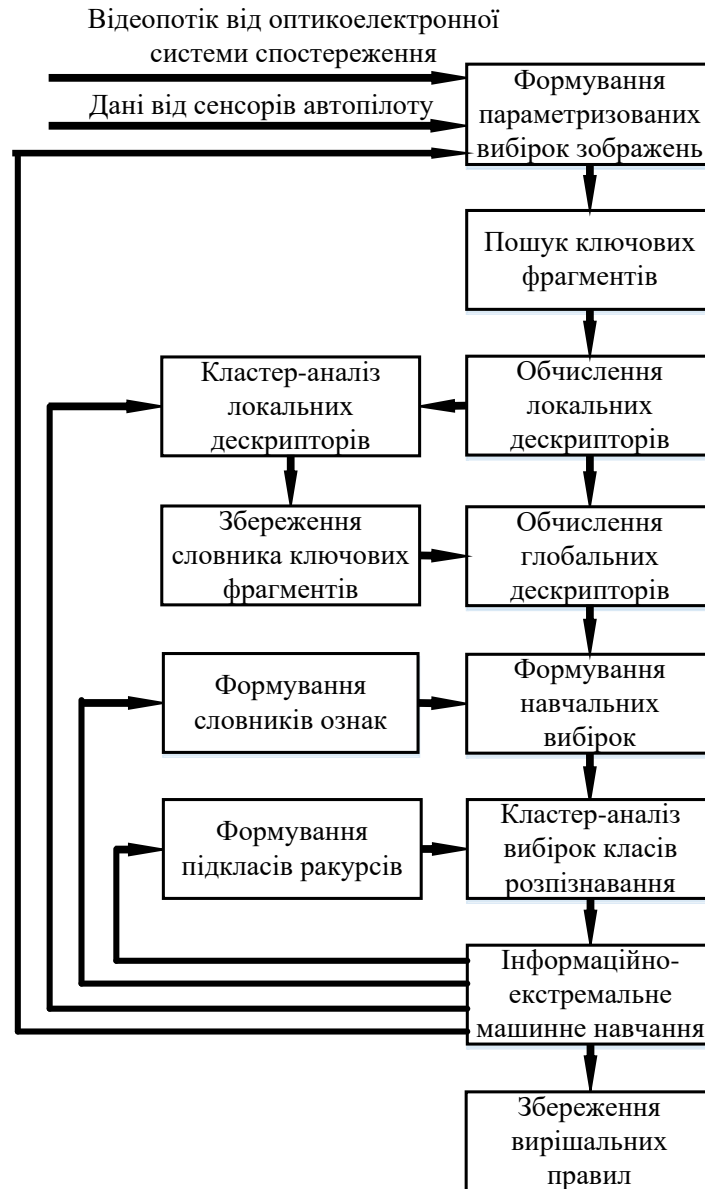


Рисунок 1.10 – Структурна схема навчання моделі ідентифікації об'єктів на місцевості

ESS – алгоритм визначення локалізації об'єкта інтересу на зображенні, що уникає повного перебору вікон/підвікон на основі методу гілок і границь (branch-and-bound search). При цьому простір пошуку вікон ділиться два підпростори, після чого здійснюється згортання підпросторів, якщо вони мають нижчу оцінку

ймовірності знаходження об'єкта інтересу порівно з іншими підпросторами (рис. 1.11) [40]. Оцінка ймовірності обчислюється на основі вирішальних правил класифікатора.

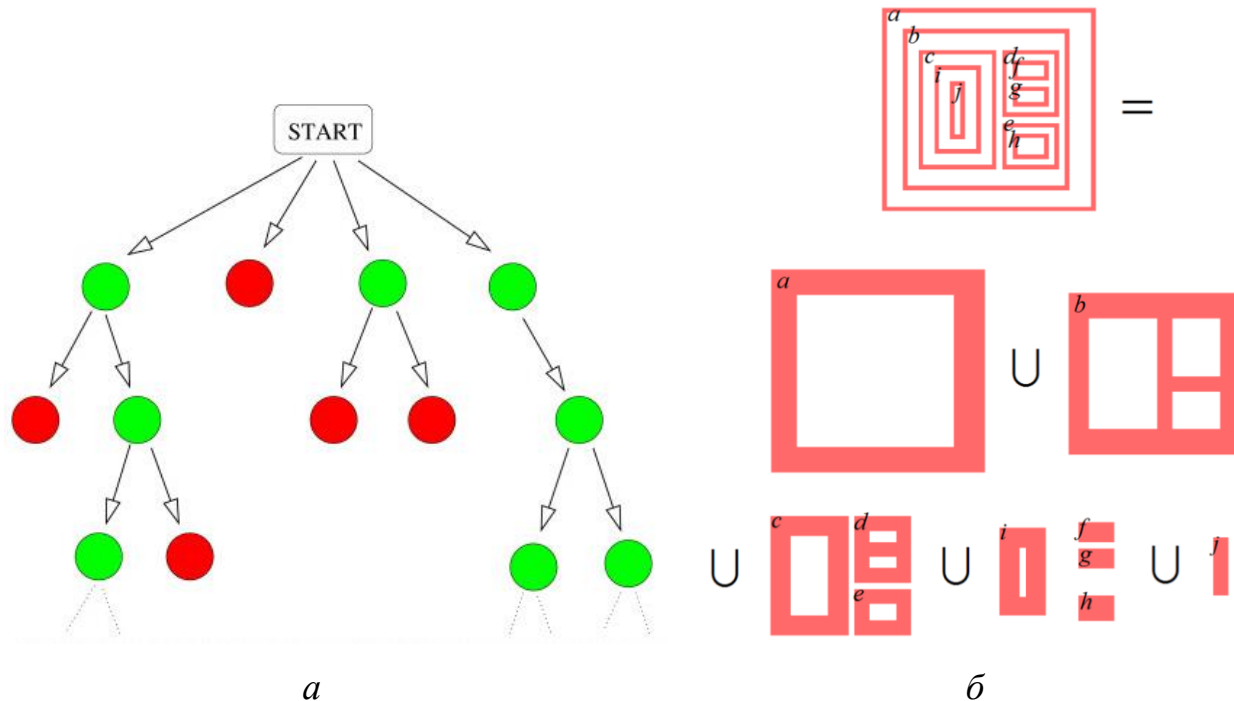


Рисунок 1.11 – Схема алгоритму ефективного пошуку підвікна :

*a* – дерево пошуку; *б* – підвікна

Селективний пошук – це алгоритм пропозиції ділянки (вікна), який використовується при детектуванні об'єкта інтересу на зображенні, що оснований на обчисленні ієрархічного групування схожих ділянок зображення на основі відповідності кольору, текстури, розміру і форми.

Селективний пошук починається з детальної сегментації зображення в залежності від інтенсивності пікселів, на основі графової сегментації і вибіркового пошуку [23]. При цьому алгоритм вибіркового пошуку приймає отримані сегменти як вхідні дані і виконує наступні кроки :

- 1) додавання всіх обмежувальних полів, що відповідають сегментованих частинам, у список пропозицій;
- 2) об'єднання суміжних сегментів групи на основі міри схожості;
- 3) перехід до першого кроку.

На кожній ітерації формується і додаються укрупнені сегменти в список пропозиції ділянки зображення (рис. 1.12). Тому пропозиції ділянок сформовані від менших сегментів до більших в порядку зростання. Селективний пошук використовує чотири міри подібності, що основані на кольорі, текстурі, розмірі і сумісності форми.

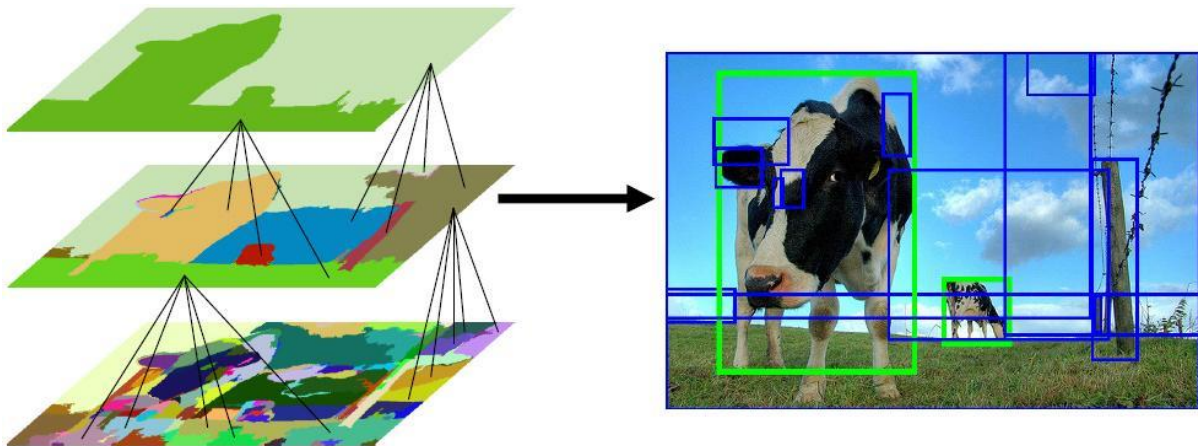


Рисунок 1.12 – Ілюстрація до алгоритму селективного пошуку

Ще одним перспективним підходом до сканування зображення є використання метаевристичних популяційних алгоритмів. Перша популяція вікон генерується випадковим шляхом, а потім параметри вікон корегуються для максимізації функцій належності до об'єктів інтересу, що обчислюються у вікнах пошуку.

У випадку високої варіативності спостережень ієрархічне ознакове подання є більш ефективним. При цьому в умовах ресурсних та інформаційних обмежень існує потреба у використанні всієї доступної апіорної інформації. Техніка передачі знань є одним з прикладів використання апіорної інформації, яка накопичена в попередньо навченій нейронній мережі [21], [22]. Такий підхід дозволяє запозичувати низькорівневі шари з глибокої мережі, попередньо навченої на великому обсязі даних. Крім того, під час навчання високорівневих шарів для адаптації до поточної доменної області використання пропонується використовувати крім розмічених також і нерозмічені навчальні зразки, що має

підвищити ефективність використання доступних даних та знизити когнітивне навантаження на оператора.

Однак, коли об'єкти інтересу мають малий розмір, то зона інтересу є недостатньо інформативною, тому важливим є використання додаткової інформації щодо контексту в якому знаходиться об'єкт розпізнавання. На рис. 1.13 зображено пропоновану архітектуру моделі детектування малорозмірних об'єктів інтересу на основі комбінації техніки передачі знань (Transfer learning) та інформації про контекст, яка отримана шляхом об'єднання карт ознак отриманих з різних шарів штучної нейронної мережі.

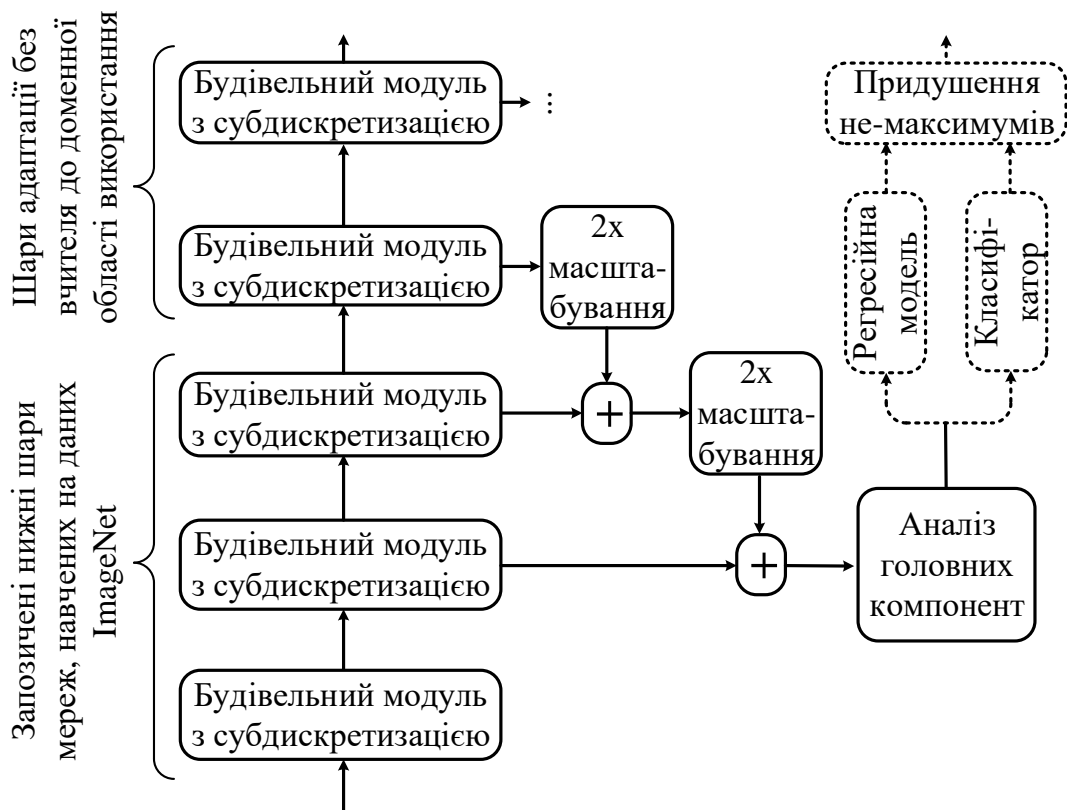


Рисунок 1.13 – Модель ієрархічного екстрактора ознак для детектування малорозмірних об'єктів на місцевості

Використання шару масштабування забезпечує однаковий розмір кожного каналу карти ознак. Конкатенація та масштабування розглядається як один шар “масштабування-конкатенації”.

Inception, Xception, VGG, Fire є найбільш популярними будівельними

модулями, які використовуються для побудови глибоких згорткових нейронних мереж [21], [23]. Ці модулі мають різну мікро-архітектуру, що обумовлює різну обчислювальну складність та ефективність навчання цих моделей. Ми пропонуємо адаптувати нижні шари наперед навченої моделі нейронної мережі Squeezenet, яка складається з Fire модулів і характеризуються високою обчислювальною ефективністю [23]. Верхні шари мережі в цьому випадку можуть бути побудовані за допомогою простих модулів архітектури VGG, які дозволяють отримати значну гнучкість при різних підходах навчання.

Глибина згорткових фільтрів кожного рівня рівна кількості каналів вхідного зображення для першого шару та кількості каналів карти активації – для решти. Активацію пікселя кожного каналу карти ознак пропонується обчислювати на основі алгоритму невід’ємного ОУП (Non-negative Orthogonal Matching Pursuit) [29]. Алгоритм невід’ємного ОУП здійснює пошук невід’ємних коефіцієнтів  $x_1, \dots, x_N$  лінійної комбінації навчених нейронів  $D$  з метою мінімізації норми помилки апроксимації вхідного патчу  $y$  (рис. 1.14).

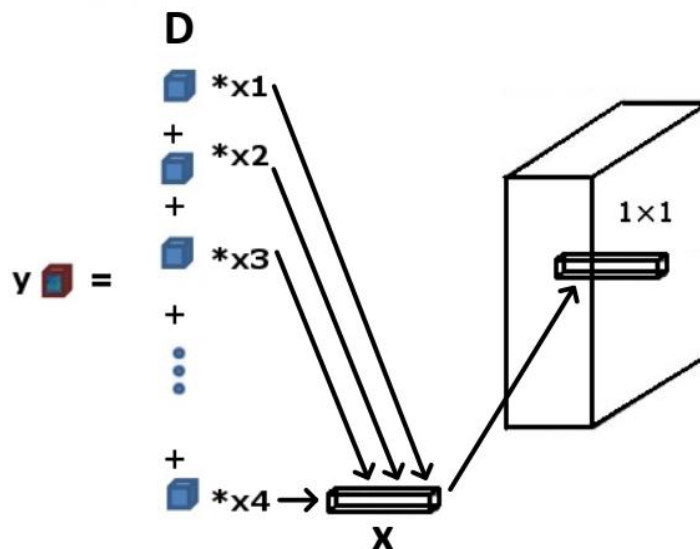


Рисунок 1.14 – Схема застосування алгоритму ортогонального узгодженого переслідування для обчислення пікселів карти ознак

Об’єднання карт ознак з різних шарів штучної нейронної мережі призводить до проблеми прокляття розмірності ознак. Для усунення даної проблеми



пропонується використовувати аналіз головних компонент. Його використання дозволяє знизити розмірність не беручи до розгляду низькорівневі ознаки, які є нечутливими до цільової доменної області застосування.

У нейромережевих детекторах пікселі з вихідної карти ознак або її масштабованих варіантів аналізуються класифікаційною та регресійною моделлю. Класифікаційний аналіз визначає наявність об'єкта інтересу на ділянці зображення, пов'язаній з пікселем карти ознак, а регресійний аналіз визначає межі обмежувального прямокутника детектованого об'єкта.

Класифікаційний аналіз карти ознак пропонується здійснювати в рамках ІЕІ-технології з двійковим кодуванням ознак на основі бустінгу. Ця технологія дозволяє виконати класифікаційний аналіз з низькою обчислювальною складністю та відносно високою точністю в умовах обмеженого обсягу навчальних даних [38].

Інформаційно-екстремальний класифікатор в режимі екзамену визначає належність вхідного вектору  $x$  з відповідним двійковим поданням  $b$  до одного з класів  $\{X_z^o \mid z = \overline{1, Z}\}$  відповідно до максимального значення функції належності  $\mu_z(b)$  на основі виразу  $\arg \max_z \{\mu_z(b)\}$ . При цьому функція належності  $\mu_z(b)$  до контейнера з опорним вектором  $b_z^*$  та радіусом  $d_z^*$ , обчислюється за формулою

$$\mu_z(b) = \exp\left(-\sum_{i=1}^{N_z} b_i \oplus b_{z,i}^* / d_z^*\right). \quad (1.14)$$

Бінарне кодування вектору  $x_j$  здійснюється шляхом конкатенації результатів прийняття рішень деревами ансамблю  $T_1, \dots, T_L$ . При цьому кожне дерево рішень формує бінарний код, де кожен ненульовий біт відповідає вузлу на шляху прийняття рішень від кореня дерева до термінального вузла (рис. 1.15). Древа рішень можуть бути побудовані за принципом багінгу або бустінгу.

В результаті вхідна навчальна вибірка кодується в бінарну навчальну

матрицю  $\{b_{z,s,i} | i = \overline{1, N_2}; s = \overline{1, n_z}; z = \overline{1, Z}\}$ , де  $N_2$  – кількість індукованих двійкових ознак;  $n_z$  – кількість навчальних зразків, що відповідають класу  $X_z^o$ .

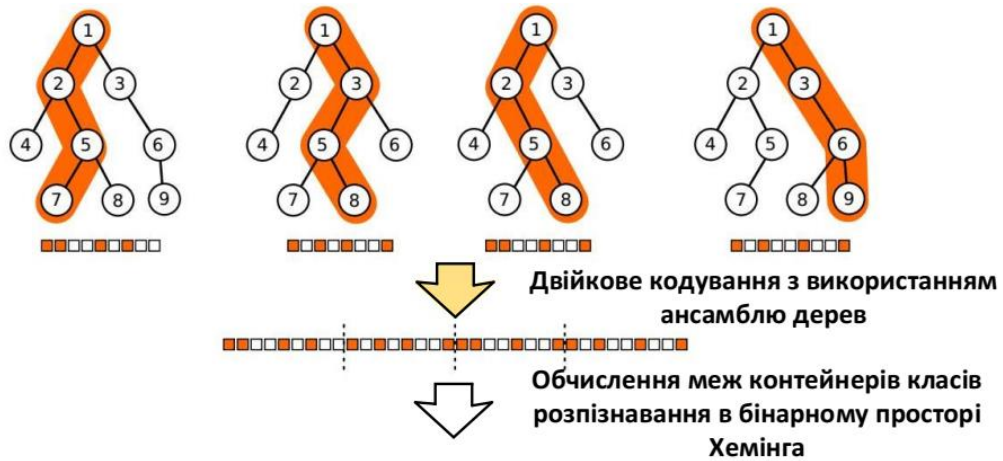


Рисунок 1.15 – Схема двійкового кодування ознак з використанням ансамблю дерева рішень

В результаті аналізу карт ознак буде сформовано множину прямокутників – попередніх претендентів. Для кожного прямокутника окрім координат обчислено функцію належності (1.14), тобто дано дві множини : множина прямокутників  $V = \{b_1, \dots, b_N\}$  та множина відповідних їм оцінок  $S = \{s_1, \dots, s_N\}$ . Однак, як правило, при цьому генерується багато дублюючих прямокутників для кожного об'єкту. Тому для формування результуючого списку прямокутників  $D$  детектованих об'єктів використовується алгоритм придушення не максимумів (Non-maximum suppression).

В алгоритмі придушення не максимумів з множини  $V$  прямокутники поступово переносять в множину  $D$ . В циклі прямокутник  $b_i \in V$  з максимальною оцінкою впевненості  $i^* = \arg \max\{S\}$  і переносимо його в список  $D$ , видаляючи зі списку  $V$ . При цьому також видаляються зі списку  $V$  всі прямокутники, що перетинаються з  $b_{i^*}$  більше ніж на заданий поріг  $T$ . При цьому, зменшення порогу  $T$  призводить до зростання шансу відкидання іншого об'єкту, що частково перекривається з заданим, замість відкидання дублікату. З

іншої сторони, збільшення порогу  $T$  призводить до зростання ймовірності отримати декілька прямокутників одного й того ж об'єкта на зображенні.

## 1.6 Модель екстракції ознакового опису для задачі автономної навігації

Визначення геокоординат детектованого об'єкта потребує інформації про геокоординати самого БПЛА. Тому в умовах недоступності радіозв'язку та радіонавігації БПЛА повинен вміти визначати власне положення і уникати перешкоди з використанням бортових відеокамер. Під час проектування екстрактора ознакового опису для задач візуальної навігації пропонується покласти в основу такі принципи :

- використання більше двох послідовних кадрів для оцінки переміщення між двома останніми кадрами з метою підвищення точності рішень в умовах розмивання рухом та наявності рухомих об'єктів в полі зору камери;

- використання ієрархічного ознакового подання спостережень, що дозволяє описувати складні ознаки меншою кількістю параметрів;

- уникнення використання шарів субдискретизації для попередження втрат інформації (дані про зсув, деформацію та масштаб), що важливо для регресійного аналізу переміщення в просторі і часу, особливо в умовах відсутності навчання з кінця-в-кінець;

- використання вирішальних правил, що мають низьку обчислювальну складність в режимах навчання та безпосереднього прийняття рішень, що дозволить скоротити час, необхідний для оцінки ефективності та настройки екстрактора ознак.

Спроектowana модель візуальної навігації повинна забезпечувати ухилення від перешкод та контроль власної позиції за допомогою одометрії. Запропонована модель візуальної навігації може бути навчена або як самостійна система, або як допоміжна система у вигляді паралельного коректора помилок для геометричної одометрії, основаної на використанні локальних ознак. У випадку допоміжного варіанту модель геометричної одометрії та коректор помилок повинні бути

синхронізовані. Схема інтелектуальної навігаційної системи для малогабаритного літального апарату показана на рис. 1.15.

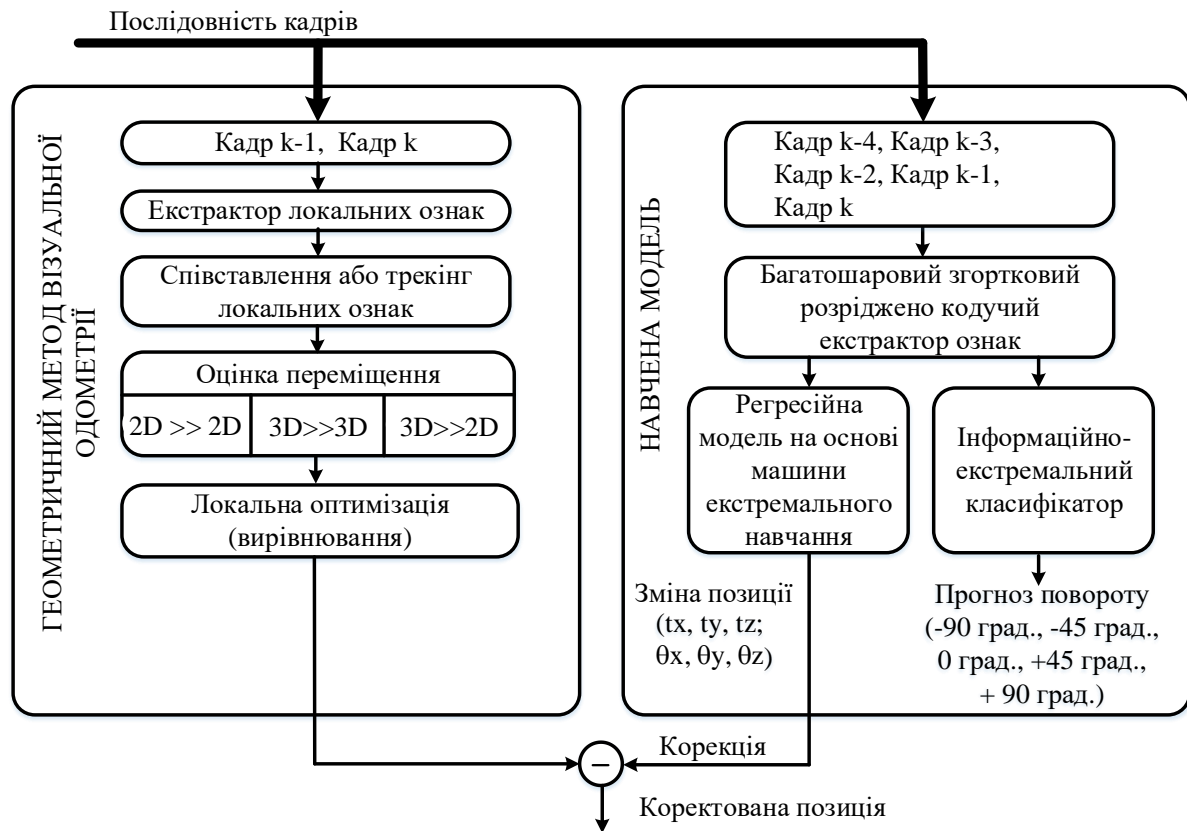


Рисунок 1.15 – Модель геометричної візуальної одометрії з пропонованою моделлю паралельного коректора помилок

Для того, щоб виділити ознаки візуальних спостережень, пропонується використовувати згорткову розріджено кодууючу нейронну мережу, на вхід якої надходить багатоканальне зображення, сформоване із серії послідовних відео кадрів у градації сірого. Згорткова розріджено кодууюча нейронна мережа має багатошарову структуру, на виході якої формується високорівневий ознаковий опис візуальних спостережень.

Згорткові фільтри пропонується попередньо навчати без вчителя, послідовно шар за шаром. Для виявлення перешкод та формування відповідної реакції використовується класифікатор, що навчається з учителем на навчальних зразках, закодованих відповідними високорівневими ознаками. Регресійна модель використовується для відображення візуальних ознак у відповідну оцінку

переміщення і положення відеокамери в просторі.

На рис. 1.16 показано 4-х шарову архітектуру згорткової розріджено-кодуючої нейронної мережі, в першому шарі якої знаходяться згорткові фільтри з різними ядрами :  $5 \times 5$ ,  $3 \times 3$  та  $1 \times 1$ . Кількість фільтрів регулюється параметром  $K_2$ . Для збереження одного і того ж розміру карт ознак, створених багатомірними фільтрами, використовується техніка заповнення нулями [10], [11].

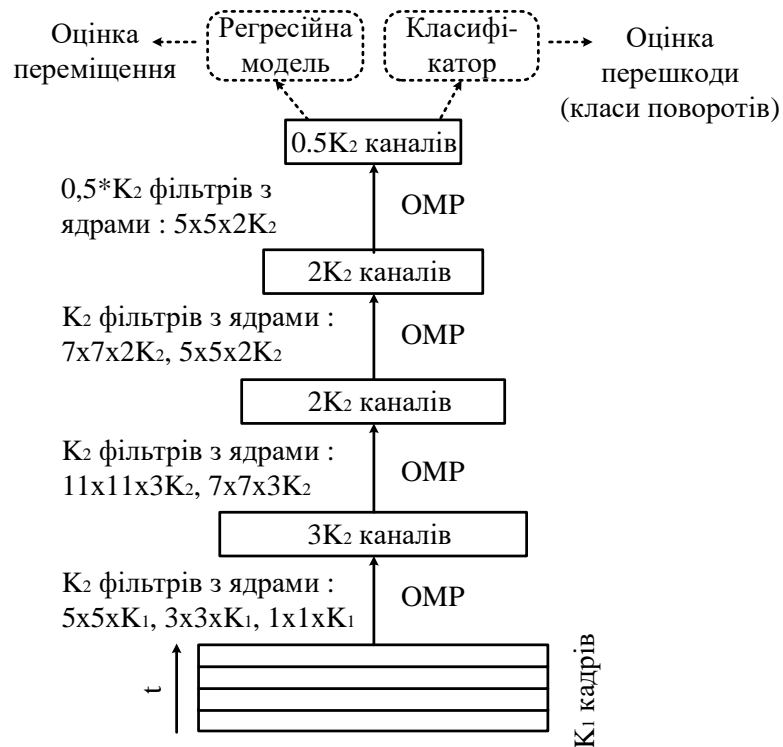


Рисунок 1.16 – Архітектура згорткової розріджено кодуючої моделі для екстракції візуальних ознак в навігаційній системі

У другому та третьому шарах параметр відступу для сканування карти ознак з багатомірними фільтрами становить 3 та 2, відповідно. При цьому пропонується використовувати алгоритм невід'ємного ОУП (рис. 1.14) для обчислення багатоканальних пікселів карти ознак [29].

Регресійну модель  $y = f(x)$  пропонується подати у вигляді мережі прямого поширення з одним прихованим шаром (МПОП). При цьому набір навчальних / тестових даних кодується візуальними ознаками  $\{(x_j, y_j) | x_j \in R^N$ ,

$y_j \in R^M, 1 \leq j \leq n$ , де  $x_j = (x_{j1}, x_{j2}, \dots, x_{jN})^T$ . Вихідна змінна  $y_j \in R^M$  відповідає вектору зміщення положення,  $t$ , та вектору повороту,  $\hat{R}$ , камери  $y = (y_1, y_2, \dots, y_n)^T = (t, \hat{R})^T$ . МПОП з  $R$  адитивними прихованими вузлами та функцією активації  $\varphi(x)$  може бути подана у вигляді системи рівнянь

$$\sum_{r=1}^R \beta_r \varphi(w_r^T x_j + b_r) = o_j, 1 \leq j \leq n$$

де  $w_r = (w_{r1}, w_{r2}, \dots, w_{rN})^T$  – вектор вагових коефіцієнтів, який зв'язує вхідний шар з  $r$ -м прихованим вузлом;

$b_r$  – зміщення  $r$ -го прихованого вузла;

$\beta_r = (\beta_{r1}, \beta_{r2}, \dots, \beta_{rM})$  – вектор вагових коефіцієнтів, що зв'язує вихідний шар з  $r$ -м прихованим вузлом;  $o_j$  – вихід мережі для вхідного вектору  $x_j$ ;

$\varphi(x)$  – функція активації.

МПОП з  $R$  прихованими вузлами може відтворити ці  $N$  зразків з нульовою помилкою, якщо всі параметри можуть вільно коригуватись, тобто існують,  $\beta_r$ ,  $w_r$  і  $b_r$ . Вище наведені вирази можуть бути компактно переписані як матрична рівність

$$H\beta = Y,$$

де

$$H = \begin{bmatrix} \varphi(w_1^T x_1 + b_1) & \dots & \varphi(w_R^T x_1 + b_R) \\ \dots & \dots & \dots \\ \varphi(w_1^T x_n + b_1) & \dots & \varphi(w_R^T x_n + b_R) \end{bmatrix}_{n \times R},$$

$$\beta = \begin{bmatrix} \beta_1^M \\ \dots \\ \beta_R^M \end{bmatrix}_{R \times M}, Y = \begin{bmatrix} y_1^M \\ \dots \\ y_n^M \end{bmatrix}_{n \times M}$$

де  $H$  – матриця виходу прихованого шару.

Таким чином, запропонований екстрактор ознак оснований на використанні багат шарової розріджено кодууючої згорткової нейронної мережі без використання шарів субдискретизації і функцій активації. Вихід мережі використовується для основної задачі регресійного аналізу переміщення та для допоміжної, регуляризуючої, задачі класифікаційного аналізу перешкод.

## 2 ОПТИМІЗАЦІЯ ПАРАМЕТРІВ ФУНКЦІОНУВАННЯ МОДЕЛІ АНАЛІЗУ ДАНИХ

### 2.1 Алгоритми пошукової оптимізації параметрів екстрактора ознак

Одним із шляхів підвищення оперативності пошуку параметрів функціонування моделі аналізу даних, максимально наближених до глобального оптимуму, є використання метаевристичних алгоритмів оптимізації. Ці алгоритми поділяються на популяційна та траєкторні.

Популяційні алгоритми є більш ресурсозатратними тому, як правило, використовуються для настройки невеликої кількості параметрів. Проте ці алгоритми не потребують початкових наближень і дозволяють знайти оптимальне рішення за відносно невелику кількість ітерацій. При цьому одна ітерація популяційного алгоритму потребує  $n_a$  обчислень цільової функції (критерію оптимізації), де  $n_a$  – кількість агентів популяції [31], [33].

Одним з найпростіших в реалізації популяційних алгоритмів пошукової оптимізації є алгоритм рою частинок (Particle Swarm Optimization). Розглянемо основні кроки реалізації алгоритму рою частинок для оптимізації вектора параметрів  $P_i, i = \overline{1, N}$ .

1. Ініціалізація рою частинок (агентів):

а) ініціалізація кількості частинок  $n_a$ ;

б) ініціалізація розмірності кожної частинки  $N$  та ініціалізація меж зміни  $i$ -ї координати  $j$ -ї частинки  $P_{j,i}$ ;

в) ініціалізація початкових позицій частинок  $P_j[0] := P_{\max} \cdot U(0,1)$ , де  $U(0,1)$  – генератор випадкових чисел з діапазону  $(0,1)$ , а  $P_{\max}$  – максимальне значення координат вектору параметрів;

г) ініціалізація початкових швидкостей частинок  $V_j(0) := 0$ ;

д) ініціалізація максимальної швидкості частинок в  $V_{\max,i}$ ;



е) ініціалізація вагових коефіцієнтів для формули швидкості, тобто ваги інерції  $w$  та констант прискорення  $c_1$  і  $c_2$ .

2. Інкремент номера ітерації:  $k := k + 1$ .

3. Інкремент номера частинки:  $j := j + 1$ .

4. Інкремент номера координати в позиції:  $i := i + 1$ .

5. Розрахунок нового стану частинки:

а) розрахунок  $i$ -ї компоненти швидкості для  $j$ -ї частинки за правилами

$$V_{j,i}[k+1] := wV_{j,i}[k] + c_1a_{1,i}[k] * (Pbest_{j,i}[k] - P_{j,i}[k]) + c_2a_2[k] * (Gbest_j - P_{j,i}[k]);$$

$$V_{j,i}[k+1] := \begin{cases} V_{j,i}[k+1] & \text{if } V_{j,i}[k+1] < V_{\max,i}, \\ V_{\max,i} & \text{if else,} \end{cases}$$

де  $a_1[k] = U(0,1)$ ,  $a_2[k] = U(0,1)$ ;

б) оновлення позиції частинки  $P_j[k+1] := P_j[k] + V_j[k+1]$ ;

в) обчислення цільової функції  $J_j[k+1]$ ;

г) оновлення значень найкращої персональної  $Pbest$  та глобальної  $Gbest$  позицій агентів пошуку

$$Pbest_j[k+1] := \begin{cases} Pbest_j[k], & \text{if } J(P_j[k+1]) \leq J(Pbest_j[k]); \\ P_j[k+1], & \text{if else;} \end{cases}$$

$$Gbest[k+1] := \arg \max_j \{J(Pbest_j[k+1])\}.$$

6. Перевірка умови зупину: якщо  $k < K_{\max}$ , де  $K_{\max}$  – максимальна кількість ітерацій пошуку, і  $J(Gbest[k+1]) < 1,0$ , то перехід до кроку 2, інакше – до кроку 7.

7. Зупин.

Для підвищення оперативності навчання також пропонується використовувати алгоритм оптимізації системою заряджених частинок (Charged System Search, CSS), який на відміну від інших алгоритмів еволюційного програмування, оснований на імітації механізмів природи, характеризується кращим співвідношенням між швидкістю збіжності та диверсифікацією пошуку [33].

Алгоритм оптимізації системою заряджених частинок заснований на використанні електростатичних законів Кулона та законів механіки Ньютона. Кожного з агентів популяції інтерпретують як заряджену частинку, заряд якої пропорційний значенню цільової функції в області пошуку, в якій знаходиться агент на поточній ітерації. Поточний заряд частинки популяції визначає сумарну силу, яка діє на дану частинку зі сторони інших частинок, а також напрямок і величину її переміщення на поточній ітерації. У відповідності до законів електростатики ця сила розраховується шляхом векторного складання сил притягання та відштовхування зі сторони усіх частинок популяції.

Перший крок роботи оптимізаційного алгоритму пошуку системою заряджених частинок полягає в ініціалізації популяції шляхом генерації  $|P|$  точок рівномірно розподілених в просторі рішень за правилом

$$p_i = P_{i,min} + U_{|P|}(0;1)(P_{i,max} - P_{i,min}), i \in [1:|P|],$$

та розрахунку значень цільової функції  $J(p_i) = J_i$  для вибору кращого рішення

$$J_i^{best} = \max_{i \in [1:|P|]} J_i.$$

На початку кожної ітерації виконується локальний пошук для кожного з поточних положень частинок  $p_i$  для збору інформації, щодо їх оточення за допомогою алгоритму лінійного стохастичного пошуку [31].

Наступним кроком ітерації пошукового алгоритму є розрахунок сумарних сил, які діють на кожну частинку популяції. Характер дії таких сил між

частинками  $p_i, p_j \in P$  може бути як притягуючий так і відштовхуючий і визначається константою

$$c_{i,j} = \begin{cases} -1, J_i < J_j; \\ 1, J_i \geq J_j. \end{cases}$$

Кожна з заряджених частинок  $p_j \in P$  є сферою, радіус якої дорівнює  $a > 0$  (вільний параметр роботи алгоритму).

Враховуючи прийняте значення константи  $c_{i,j}$  результуюча сила, яка діє на частинку зі сторони всіх інших частинок популяції, має вигляд

$$F_i = q_i \sum_{\substack{j=1 \\ j \neq i}}^{|P|} \left( \frac{q_i r_{i,j} \beta_1}{a^3} + \frac{q_i \beta_2}{r_{i,j}^2} \right) \times c_{i,j} (p_i - \gamma),$$

де значення  $\beta_1, \beta_2$  визначаються умовами :

$$\begin{aligned} \beta_1 &= 0, \beta_2 = 1, \text{ якщо } r_{i,j} \geq a, \\ \beta_1 &= 1, \beta_2 = 0, \text{ якщо } r_{i,j} < a. \end{aligned}$$

На поточній ітерації роботи алгоритму заряду кожної з частинок  $p_i \in P$  ставиться у відповідність величина  $q_i$ , рівна нормованому значенню цільової функції в поточному положенні  $p_i$  цієї частинки

$$q_i = \left| \frac{J_i - J^{best}}{J^{best} - J^{worst}} \right| i \in [1 : |P|],$$

де  $J^{best}, J^{worst}$  – краще та гірше значення цільової функції.

При цьому поточна відстань  $r_{i,j}$  між частинками  $p_i, p_j$  визначається за формулою

$$r_{i,j} = \frac{\|p_i - p_j\|_J}{\left\| \frac{p_i - p_j}{2} - p^{best} \right\|_J + \varepsilon}, i, j \in [1:|P|],$$

де  $\varepsilon$  – будь-яке мале знакододатнє число, яке дозволяє уникнути появи нуля в знаменнику дробу.

Міграція частинок в нові позиції здійснюється за правилом

$$p'_i = U(0;1)b_a \frac{F_l}{q} + U_l(0;1)b_v V_i + p_i, i \in [1:|P|],$$

$$V'_i = p'_i - p_i, i \in [1:|P|]$$

де  $b_a^t = 3(1 - t/\hat{t})$ ,  $b_v^t = 1 - t/\hat{t}$  – вагові коефіцієнти прискорення та швидкості відповідно, які зазвичай визначають баланс між диверсифікацією та інтенсифікацією пошуку, де  $t, \hat{t}$  – кількість пройдених та максимальна кількість ітерації роботи алгоритму.

У кінці кожної ітерації пошукового алгоритму відбувається перевірка виконання умови завершення роботи алгоритму, а саме досягнення заданого максимального числа ітерацій роботи алгоритму чи досягнення цільовою функцією свого максимального значення.

Тонке настроювання параметрів екстрактора ознакового опису вимагає використання зменшення кроку модифікації рішень, що обумовлює необхідність великої кількості ітерацій пошуку глобального оптимуму цільової функції. У випадку трудомісткого обчислення цільової функції і за великої кількості

параметрів доцільніше використовувати траєкторні метаевристичні алгоритми, що оперують лише одним рішенням на кожній ітерації пошуку. Серед траєкторних алгоритмів найбільш ефективними вважаються алгоритм симуляції відпалу (Simulated Annealing) та алгоритм сходження на пагорб (Hill Climbing) [36]. При цьому алгоритм симуляції відпалу є більш універсальним і при деяких настройках придатний як для глобальної, так і для локальної оптимізації параметрів моделі.

На рис. 2.1 показано псевдо код алгоритму симуляції відпалу, на кожній ітерації якого виконується обчислення цільового критерію  $f()$  шляхом пропускання розміченого навчального набору даних через модель аналізу даних та розрахунку комплексного критерію (1.3) [36], [37]. Початкове рішення утворюється за допомогою процедури *ініціалізація\_початкового\_рішення()*, що може бути реалізована на основі алгоритмів навчання без вчителя. При цьому ефективність алгоритму симуляції відпалу залежить від реалізації процедури *формування\_сусіднього\_рішення()* для формування нового рішення  $S_i$  на  $i$ -ій ітерації алгоритму.

```

Scurrent ← ініціалізація_початкового_рішення()
Sbest ← Scurrent
T ← T0
c ←  $\epsilon$ ,  $0 < \epsilon < 1$ 
для від i = 1 до epochs_max
    si ← формування_сусіднього_рішення(scurrent)
    якщо  $f(s_i) \geq f(s_{current})$ , то
        Scurrent ← si
        якщо  $f(s_i) \geq f(s_{best})$ , то sbest ← si

    інакше  $\exp\left(\frac{f(s_{current}) - f(s_i)}{T}\right) > \text{uniform\_random}(0,1)$  і Scurrent ← si
    T ← c × T
повернення Sbest

```

Рисунок 2.1 – Псевдокод алгоритму симуляції відпалу

Аналіз псевдокоду на рис. 2.1 показує, що поточне рішення  $S_{current}$ , відносно

якого відбувається пошук нових кращих рішень  $S_{best}$ , оновлюється у випадку знаходження нового рішення, що збільшує критерій (1.3) або випадковим чином з розподілу Гіббса. Для формування нового рішення пропонується використовувати найпростіший неадаптивний алгоритм, який можна представити у вигляді формули [37]

$$S_{current} = S_{current} + \text{uniform\_random}(-1,1) \cdot \text{step\_size}, \quad (2.1)$$

де  $\text{uniform\_random}$  – функція генерації випадкових чисел з рівномірного розподілу з заданого діапазону;

$\text{step\_size}$  – це розмір діапазону пошуку нових рішень, сусідніх з  $S_{current}$ .

Таким чином, за умов відсутності великих обсягів розмічених навчальних даних реалізація навчання з кінця-в-кінець на основі алгоритму зворотного поширення помилки є ускладненою і не може бути успішно реалізована, особливо при обмежених обчислювальних ресурсах. Тому запропоновано формувати квазіоптимальне рішення на основі алгоритмів навчання без вчителя, а їх корекцію здійснювати на основі популяційних чи траєкторних метаевристичних пошукових алгоритмів, що характеризуються низькою ймовірністю застрягання у “поганому” локальному оптимумі.

## **2.2 Алгоритм та результати машинного навчання однорівневого екстрактора ознакового опису для задачі детектування об’єктів на місцевості**

Побудова словника візуальних слів здійснюється за результатами навчання без вчителя. При цьому вибірка навчальних зразків формується з SURF-дескрипторів виділених ключових фрагментів розмічених та нерозмічених аеро-відео- зображень. Серед алгоритмів навчання без вчителя набув популярності метод кластер-аналізу  $k$ -середніх. Однак  $k$ -середніх характеризується повільною збіжністю та субоптимальністю результатів внаслідок жорсткої конкурентної схеми навчання і чутливості до початкової ініціалізації кластерів.

Перспективним підходом до реалізації навчання без вчителя є використання принципів нейронного газу та правила Ойа, що реалізовано в так званому алгоритмі розріджено кодуючого нейронного газу. Алгоритм нейронного газу характеризується м'якою конкурентною схемою навчання, що призводить до більш надійної збіжності алгоритму і оптимального розподілу кластерів на вибірці вхідних даних. При цьому використання правила Ойа дозволяє сформувати словник, що забезпечує мінімальну корельованість ознак і завадозахищеність кодування спостережень.

Вхідними даними для алгоритму розріджено кодуючого нейронного газу є потужність словника базисних векторів  $M$  (словника візуальних слів), розмірність простору ознак  $N$ ,  $\lambda_0, \lambda_{final}$  – початкове та кінцеве значення коефіцієнта розміру околу сусідів,  $\eta_0, \eta_{final}$  – початкове та кінцеве значення коефіцієнта швидкості навчання. Розглянемо основні кроки алгоритму.

- 1) ініціалізація словника базисних векторів  $D = (d_1, \dots, d_M)$  випадковими числами з рівномірного розподілу;
- 2) ініціалізація лічильника навчальних векторів  $t := 1$ .
- 3) вибір випадкового вектора  $x$  з множини навчальних векторів  $X$ .
- 4) L2-нормалізація векторів із словника  $D = (d_1, \dots, d_M)$  шляхом приведення до одиничної довжини;
- 5) обчислення поточних значень коефіцієнта розміру околу сусідів  $\lambda_t$  та швидкості навчання  $\eta_t$  :

$$\lambda_t := \lambda_0 (\lambda_{final} / \lambda_0)^{t/t_{max}}, \quad \eta_t := \eta_0 (\eta_{final} / \eta_0)^{t/t_{max}};$$

- б) обчислення міри схожості вхідного вектора  $x$  до базисних векторів  $d_{l_k} \in D$  для їх сортування

$$-(d_{l_0}^T x)^2 \leq \dots \leq -(d_{l_k}^T x)^2 \leq \dots \leq -(d_{l_{M-1}}^T x)^2;$$

7) оновлення координат базисних векторів  $d_{l_k} \in D$  за правилом Ойа [29] :

$$d_{l_k} := d_{l_k} + \eta_t \exp(-k / \lambda_t) y(x - y d_{l_k}), \quad y := d_{l_k}^T x, \quad k = \overline{0, M-1};$$

8) якщо  $t < t_{\max}$ , то інкремент лічильника  $t := t + 1$  та перехід до кроку 3, інакше – зупин виконання алгоритму.

Для кодування ключового фрагменту зображення одним візуальним словом можна обрати візуальне слово за мінімумом косинусної міри відстані. Для кодування ключового фрагменту зображення лінійною комбінацією візуальних слів можна скористатися алгоритмом невід’ємного ОУП.

Для реалізації ефективного розгортання і розпаралеленого виконання на великому різноманітті обчислювальних платформ розроблено граф обчислень з використанням операцій фреймворку Tensorflow (рис. 2.2) [24].

Опис операцій графу обчислень (рис. 2.2) приведено у табл. 2.1.

Таблиця 2.1 – Операції фреймворку Tensorflow для реалізації розріджено кодуючого нейронного газу

| Найменування операції | Короткий опис  |
|-----------------------|--|
| add                   | По елементне додавання двох значень тензора                  |
| argsort               | Сортування значень тензора за вказаним напрямком             |
| div                   | По елементне ділення двох значень тензора                    |
| dot                   | Скалярний добуток тензора                                    |
| exp                   | По елементний розрахунок значення експоненти тензора         |
| gather                | Зрізи за заданою віссю тензора відповідно до масиву індексів |
| L2 norm               | L2 нормалізація тензора                                      |
| mul                   | По елементне множення двох значень тензорів                  |



Продовження табл. 2.1

| Найменування операції | Короткий опис                                 |
|-----------------------|---|
| pow                   | По елементне приведення тензора до степеню    |
| sub                   | По елементне віднімання значень двох тензорів |

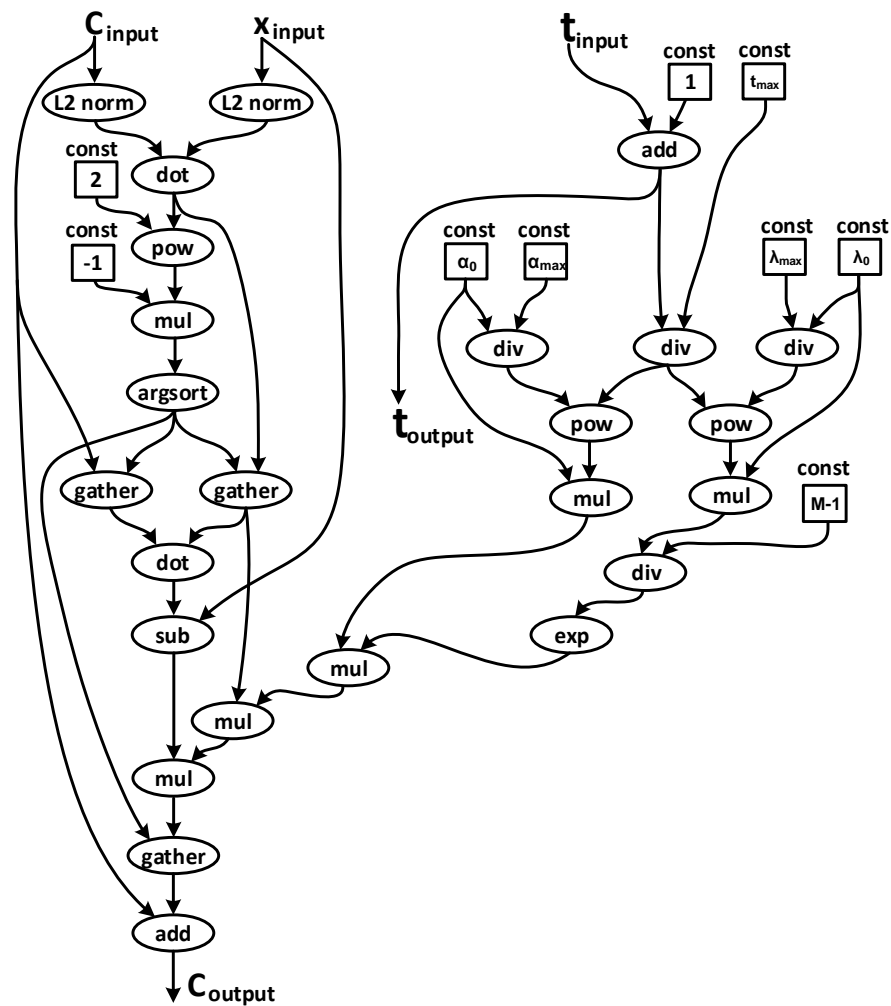


Рисунок 2.2 – Граф обчислень для реалізації однієї ітерації розріджено кодуючого нейронного газу

Розглянемо процедуру настроювання потужності алфавіту візуальних слів  $N_w$  та порогу відкидання локальних ознак  $T_H$  зображення як процедуру популяційного пошуку максимуму фітнес-функції в двовимірному просторі рішень. При цьому агент популяції містить параметри  $\Delta_w$  та  $\Delta_H$  для розрахунку

параметрів алгоритму екстракції ознак за правилами :

$$N_W = N_W^{min} + \frac{\Delta_W}{\Delta_{MAX}} (N_W^{max} - N_W^{min}) ,$$

$$T_H = T_H^{max} - \frac{\Delta_H}{\Delta_{MAX}} (T_H^{max} - T_H^{min}) ,$$

де  $N_W^{min}$  ,  $N_W^{max}$  – мінімальна та максимальна потужність алфавіту візуальних слів відповідно;

$P_j$  ,  $Pbest_j$  – поточна та найкраща персональна позиції  $j$ -го агента відповідно;

$\Delta_H(P_j)$  ,  $\Delta_H(Pbest_j)$  – параметр порогу відкидання локальних ознак поточної та найкращої персональної позицій  $j$ -го агента відповідно;

$\Delta_W(P_j)$  ,  $\Delta_W(Pbest_j)$  – параметр потужності словника візуальних слів поточної та найкращої персональної позицій  $j$ -го агента відповідно;

$T_H^{min}$  ,  $T_H^{max}$  – мінімальний та максимальний поріг відкидання локальних ознак відповідно.

Ефективність кожної частинки популяційного алгоритму, тобто її близькість до глобального оптимуму, вимірюється за допомогою наперед визначеної фітнес-функції, роль якої в даному випадку виконує функція критерію ефективності класифікаційної моделі (1.5) та (1.7). Кожна  $j$ -та частинка крім її позиції  $P_j$  зберігає наступну інформацію:  $V_j$  – поточна швидкість частинки,  $Pbest_j$  – краща персональна позиція частинки. Краща персональна позиція  $j$ -ї частинки – це позиція  $j$ -ї частинки, в якій значення фітнес функції для частинки було максимальним на поточний момент часу. Крім цього, з метою пошуку глобального екстремуму фітнес-функції найкраща частинка шукається в усьому рої, а її позиція позначається як  $Gbest$  .

Проте розглянутий вище ройовий алгоритм пошуку спрямований

переважно на видалення заважаючих ознак розпізнавання і відповідно на підвищення усередненого за алфавітом класів інформаційного критерію навчання (1.7). З метою додаткового скорочення потужності словника ознак шляхом видалення решти неінформативних ознак потрібно виконати деяку модифікацію ройового алгоритму пошуку. Для цього слід модифікувати процедуру оновлення значень найкращої персональної  $Pbest_j$  позиції агентів пошуку за правилом, в якому цільова функція  $E(\dots)$  є усередненим значенням функції критерію (1.7).

$$\begin{aligned} & \text{if } \left| E(P_j) - E(Pbest_j) \right| < \varepsilon \\ & \text{if } \sqrt{\Delta_H^2(P_j) + \Delta_W^2(P_j)} < \sqrt{\Delta_H^2(Pbest_j) + \Delta_W^2(Pbest_j)} \\ & \quad Pbest_j := P_j \end{aligned} \quad (2.2)$$

Аналогічно потрібно модифікувати процедуру оновлення значень найкращої глобальної  $Gbest_j$  позиції агентів пошуку

$$\begin{aligned} & \text{if } \left| E(Pbest_j) - E(Gbest) \right| < \varepsilon \\ & \text{if } \sqrt{\Delta_H^2(Pbest_j) + \Delta_W^2(Pbest_j)} < \sqrt{\Delta_H^2(Gbest) + \Delta_W^2(Gbest)} \\ & \quad Gbest := Pbest_j \end{aligned} \quad (2.3)$$

При заданих параметра пошукового алгоритму рою частинок ( $c_1 = c_2 = 1$ ,  $V_{max,j} = 2$ ,  $n_a = 5$ ,  $w = 0,95$  та  $K_{iter} = 5$ ) було виконано настроювання екстрактора ознак за вибірковими даними 4-х класів розпізнавання з обсягом вибірки  $n_m = 100$ . На рис. 2.3 показано кращі рішення, отримані за ройовим алгоритмом з урахуванням модифікацій (2.2) та (2.3) протягом 5-ти ітерацій настроювання екстрактора.

Аналіз рис. 2.3 показує, що кращі позиції частинок протягом 5-ти ітерацій спочатку розміщувалися в області з більшою кількістю кластерів і високим порогом відкидання локальних ознак, але остаточно алгоритм збігся до більш компромісного варіанту, який забезпечує граничне значення інформаційного

критерію і характеризується середніми значеннями параметрів настройки екстрактора. При цьому оптимальна кількість кластерів становить  $N_w^* = 1000$ , а оптимальний поріг відкидання локальних ознак рівний  $T_H^* = 0,002$ .

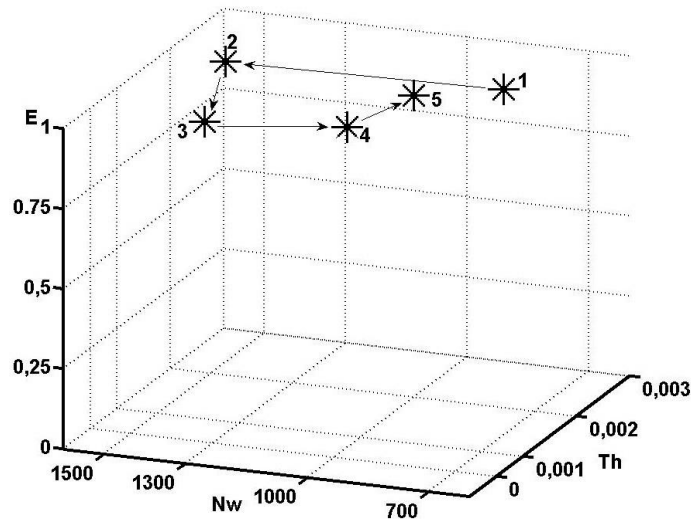


Рисунок 2.3 – Тривимірний графік залежності інформаційного критерію (1.5) від кращих позицій агентів рою протягом 5-ти ітерацій ройового алгоритму настроювання параметрів екстрактора ознак

Оцінка ефективності процедури настроювання в процесі машинного навчання потребує більш детального аналізу складових цієї ефективності. Для цього розглянемо область значень параметрів настройки екстрактора ознак та відповідні оцінки функціональної ефективності і часу навчання системи розпізнавання (рис. 2.4).

Аналіз рис. 2.3 та рис. 2.4 показує, що оптимальні значення параметрів екстрактора ознак, що визначаються на кожному кроці настроювання, знаходяться біля межі, вихід за яку супроводжується зниженням оперативності та функціональної ефективності машинного навчання системи розпізнавання об'єктів на місцевості. При цьому вирішальні правила мають геометричний зміст і відновлюються у бінарному просторі Хеммінга в процесі максимізації інформаційного критерію (1.7).

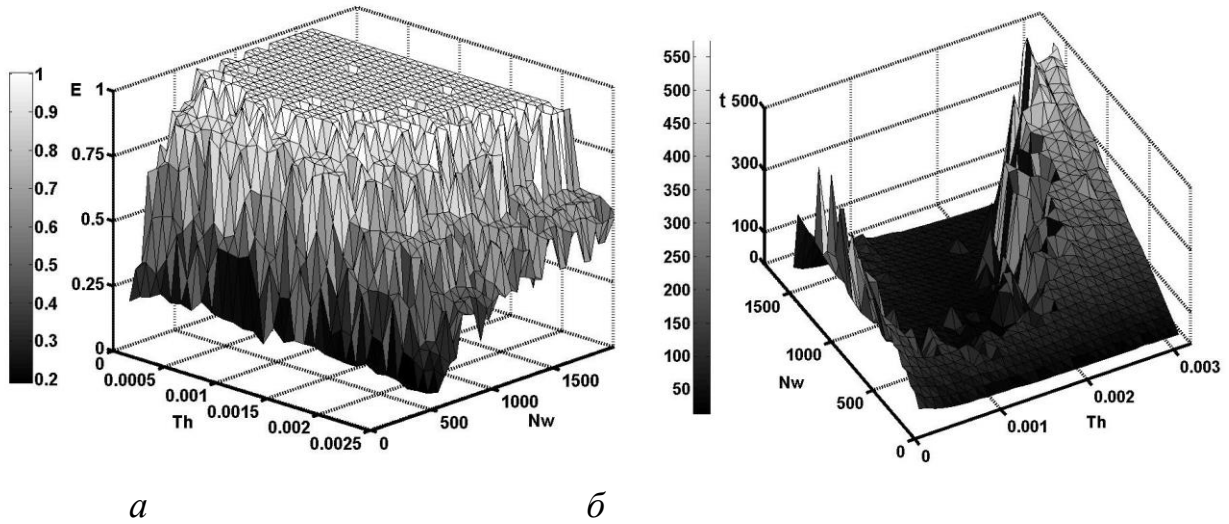


Рисунок 2.4 – Тривимірні графіки залежності мір ефективності навчання від параметрів настройки екстрактора ознак: *a* – інформаційний критерій ефективності вирішальних правил (1.7); *б* – час машинного навчання в секундах

У табл. 2.2 показано динаміку зміни інформаційного критерію та точнісних характеристик вирішальних правил при оптимальних настройках екстрактора ознак в процесі зміни максимальної кількості дерев, що здійснюють двійкове кодування спостережень..

Таблиця 2.2 – Результати популяційного алгоритму оптимізації кількості кодуєчих дерев класифікатора

| Кількість<br>дерев рішень | Значення усередненого за<br>алфавітом критерію (1.7) | Помилка<br>першого роду,<br>$\alpha_z$ | Помилка<br>другого роду,<br>$\beta_z$ |
|---------------------------|--|--|---------------------------------------|
| 50                        | 0,750  | 0                                      | 0,009                                 |
| 100                       | 0,898  | 0                                      | 0,002                                 |
| 150                       | 1,000  | 0,000                                  | 0,000                                 |
| 200                       | 1,000  | 0,000                                  | 0,000                                 |
| 250                       | 1,000  | 0,000                                  | 0,000                                 |

Аналіз табл. 2.2 показує, що практично з першої ітерації ройового пошуку

було отримано високодостовірні вирішальні правила, однак граничне значення інформаційного критерію було отримано на третій ітерації.

При заданих параметра пошукового алгоритму рою частинок ( $c_1 = c_2 = 1$ ,  $V_{max,j} = 2$ ,  $n_a = 5$ ,  $w = 0,95$ ,  $K_{iter} = 5$ ) було виконано сканування зображень. Як вхідні дані для алгоритму навчання розглядаються фотографії розміром 100x100 пікселів для 3-х об'єктів інтересу та фонові місцевості без об'єктів інтересу. Для алгоритму сканування використовуються фотографії місцевості формату Full HD (1920 x 1080 пікселів) із зображенням об'єктів інтересу, отримані на тій же висоті, на якій робилися навчальні зображення.

Оцінку ефективності алгоритму сканування зображення пропонується здійснювати шляхом вимірювання середнього часу пошуку об'єкта інтересу та підрахунку відсотку правильних спрацювань. Результати порівняння розробленого алгоритму сканування з відомими алгоритмами RAWs, ESS та Selective Search показано в табл. 2.3. При цьому ройовий та RAWs алгоритми використовують незмінне значення розміру скануючого вікна (100x100 пікселів), розраховане на основі матриці внутрішніх калібровочних параметрів камери та відомої висоти спостереження [31]. Відео- спостереження для навчання та тестування були сформовані на висоті  $\approx 50$  метрів з кутом огляду 90 градусів на одній і тій же місцевості ( $\approx 1000 \times 1000$  м) в один і той же день.

Таблиця 2.3 – Результати сканування тестових зображень

| Алгоритм сканування | Середній час обробки одного зображення, с | Відсоток хибних спрацювань, % | Відсоток пропусків, % |
|---------------------|---|-------------------------------|-----------------------|
| RAWs                | 0,687                                     | 0                             | 5                     |
| ESS                 | 0,811                                     | 0                             | 0                     |
| Selective Search    | 1,4                                       | 0                             | 2                     |
| Ройовий алгоритм    | 0,208                                     | 0                             | 1                     |

Аналіз табл. 2.3 показує, що використання апріорної інформації про умови спостереження за об'єктом інтересу та ройового алгоритму сканування

зображення дозволяє з прийнятною для практичного використання точністю детектувати об'єкти інтересу. При цьому ройовий алгоритм більше ніж втричі швидше за інші алгоритми здійснює пошук, забезпечуючи швидкість сканування Full HD відеопотоку до 5 кадрів/с.

Таким чином, застосування ройового сканування зображення дозволяє здійснювати обробку відео- потоку у форматі Full HD з прийнятною для практичного використання точністю та швидкістю. Однак фізичне моделювання відбувалося при обмеженій варіації спостережень і на невеликій висоті польоту, при якій розмір об'єктів на зображенні досить великий ( $\approx 100 \times 100$  пікселів). Для більш складних умов спостереження при великій варіації спостережень необхідно використовувати ієрархічний екстрактор ознакового опису.

### **2.3 Алгоритм та результати машинного навчання ієрархічного екстрактора ознакового опису для задачі детектування об'єктів на місцевості**

Першим етапом пропонованого алгоритму навчання детектора об'єктів на зображенні є реалізація навчання без вчителя високорівневих шарів мережі для максимального використання нерозмічених реалізацій навчальних даних. Навчання без вчителя високорівневих шарів мережі пропонується виконувати з використанням алгоритму зростаючого розріджено кодуючого нейронного газу, що оснований на принципах зростаючого нейронного газу та розрідженого кодування.

Набір даних для навчання згорткових фільтрів формується шляхом розбиття вхідних зображень або карт ознак на патчі. Ці патчі трансформуються до 1D-векторів та надходять на вхід зростаючого розріджено кодуючого нейронного газу, основні етапи якого наведені нижче :

- 1) ініціалізація лічильника навчальних векторів  $t := 0$ ;
- 2) два початкових вузли (нейрони)  $w_a$  і  $w_b$  ініціалізуються шляхом випадкового вибору векторів з навчальних даних. Вузли  $w_a$  і  $w_b$  з'єднуються ребром, вік якого встановлюється нульовим. Ці вузли вважаються нефіксованими;

3) обирається наступний вектор  $x$ , що нормалізується шляхом приведення до одиничної довжини (L2-нормування);

4) нормалізується кожний базисний вектор,  $w_k, k = \overline{1, M}$ , шляхом приведення до одиничної довжини (L2-нормалізація);

5) розраховується міра схожості вхідного вектора  $x$  до базисних векторів  $w_{s_k} \in W$  для сортування

$$-(w_{s_0}^T x)^2 \leq \dots \leq -(w_{s_k}^T x)^2 \leq \dots \leq -(w_{s_{M-1}}^T x)^2;$$

6) визначається найближчий вузол  $w_{s_0}$  та другий за близькістю вузол  $w_{s_1}$ ;

7) збільшується на одиницю вік всіх вузлів, інцидентних до  $w_{s_0}$ ;

8) якщо вузол  $w_{s_0}$  фіксований, то перехід до кроку 9, в інакше – крок 10;

9) якщо  $(w_{s_0}^T x)^2 \geq \nu$ , то перехід до кроку 12. В іншому випадку додаються новий нефіксований нейрон  $w_r = x$  та нове ребро, яке з'єднує  $w_r$  і  $w_{s_0}$ , потім перехід до кроку 13;

10) вузол  $w_{s_0}$  та його топологічні сусіди (вузли, зв'язані з ним ребрами) зміщуються у напрямку вхідного вектора  $x$  відповідно до правила Ойя [29] за формулами

$$\square w_{s_0} = \varepsilon_b \eta_t w_{s_0}^T x (x - w_{s_0}^T x w_{s_0}), \quad \square w_{sn} = \varepsilon_n \eta_t w_{sn}^T x (x - w_{sn}^T x w_{sn}),$$

$$0 < \varepsilon_b \leq 1, \quad 0 < \varepsilon_n \leq \varepsilon_b, \quad \eta_t := \eta_0 (\eta_{final} / \eta_0)^{t/t_{max}},$$

де  $\square w_{s_0}, \square w_{sn}$  – вектори корекції вагових коефіцієнтів нейрона-переможця та його топологічних сусідів відповідно;  $\varepsilon_b, \varepsilon_n$  – константи міри оновлення вагових коефіцієнтів нейрона-переможця та його топологічних сусідів відповідно;  $\eta_0, \eta_t, \eta_{final}$  – початкове, поточне та кінцеве значення швидкості навчання відповідно;

11) якщо  $(w_{s_0}^T x)^2 \geq \nu$ , то нейрон  $w_{s_0}$  позначається як фіксований;

12) якщо  $w_{s_0}$  і  $w_{s_1}$  з'єднані ребром, їхній вік зануляється, в іншому випадку – між  $w_{s_0}$  і  $w_{s_1}$  формується нове ребро з нульовим віком;



13) всі ребра графа з віком більше, ніж  $a_{\max}$  видаляються. У тому випадку, коли у деяких вузлах немає ребер (вузли стають ізольованими), ці вузли також видаляються;

14) якщо  $t < t_{\max}$ , то переходимо до кроку 15, інакше – збільшуємо лічильник  $t := t + 1$  і переходимо до кроку 3;

15) якщо всі нейрони фіксовані, то виконання алгоритму припиняється, інакше перехід до кроку 3 і починається нова епоха навчання (повторення набору даних навчання).

Для реалізації ефективного розгортання і розпаралеленого виконання алгоритму зростаючого розріджено кодуєного нейронного газу на великому різноманітті обчислювальних платформ розроблено граф обчислень з використанням операцій фреймворку Tensorflow [24]. Його структурні частини показано на рис. 2.5 та рис. 2.6.

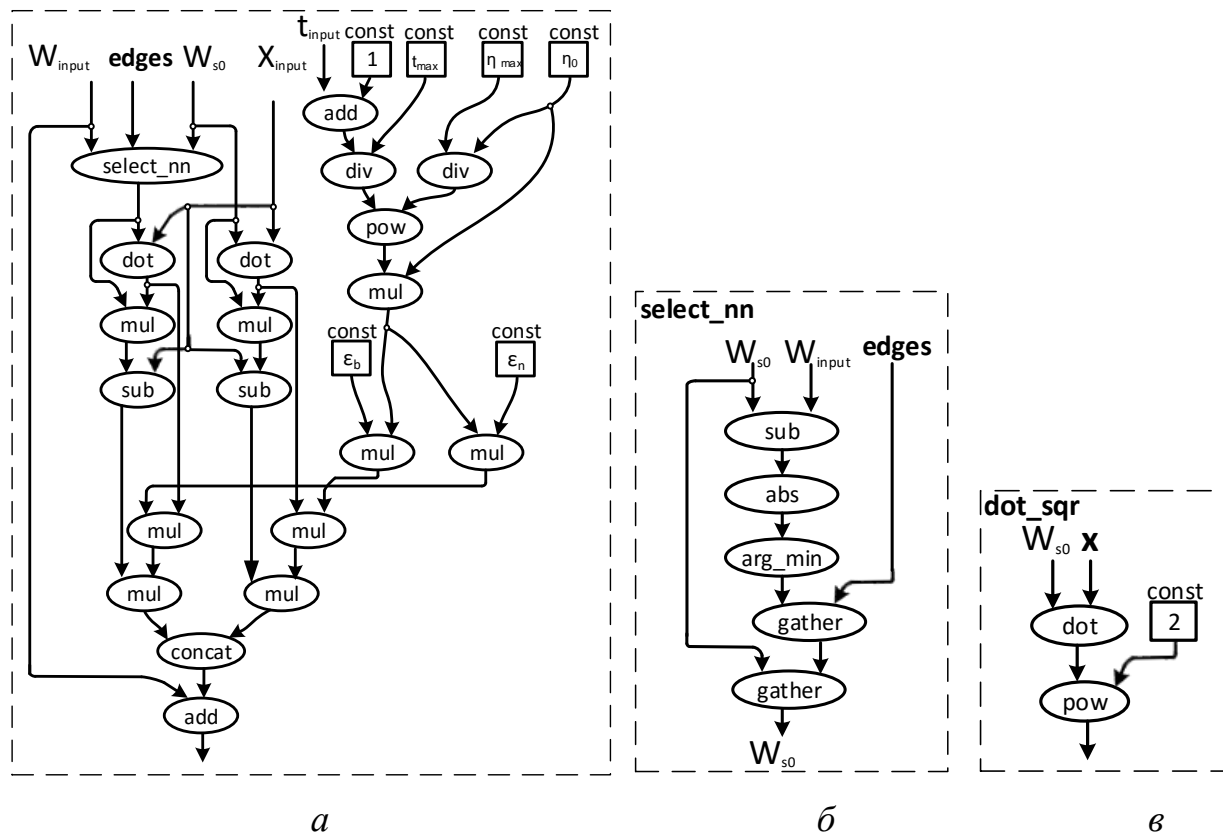


Рисунок 2.5 – Структурні модулі графу обчислень для реалізації однієї ітерації зростаючого розріджено кодуєного нейронного газу : *a* – процедура оновлення нейрона переможця та його топологічних сусідів; *b* – процедура вибору топологічних сусідів нейрона переможця; *c* – обчислення міри відгуку нейронів

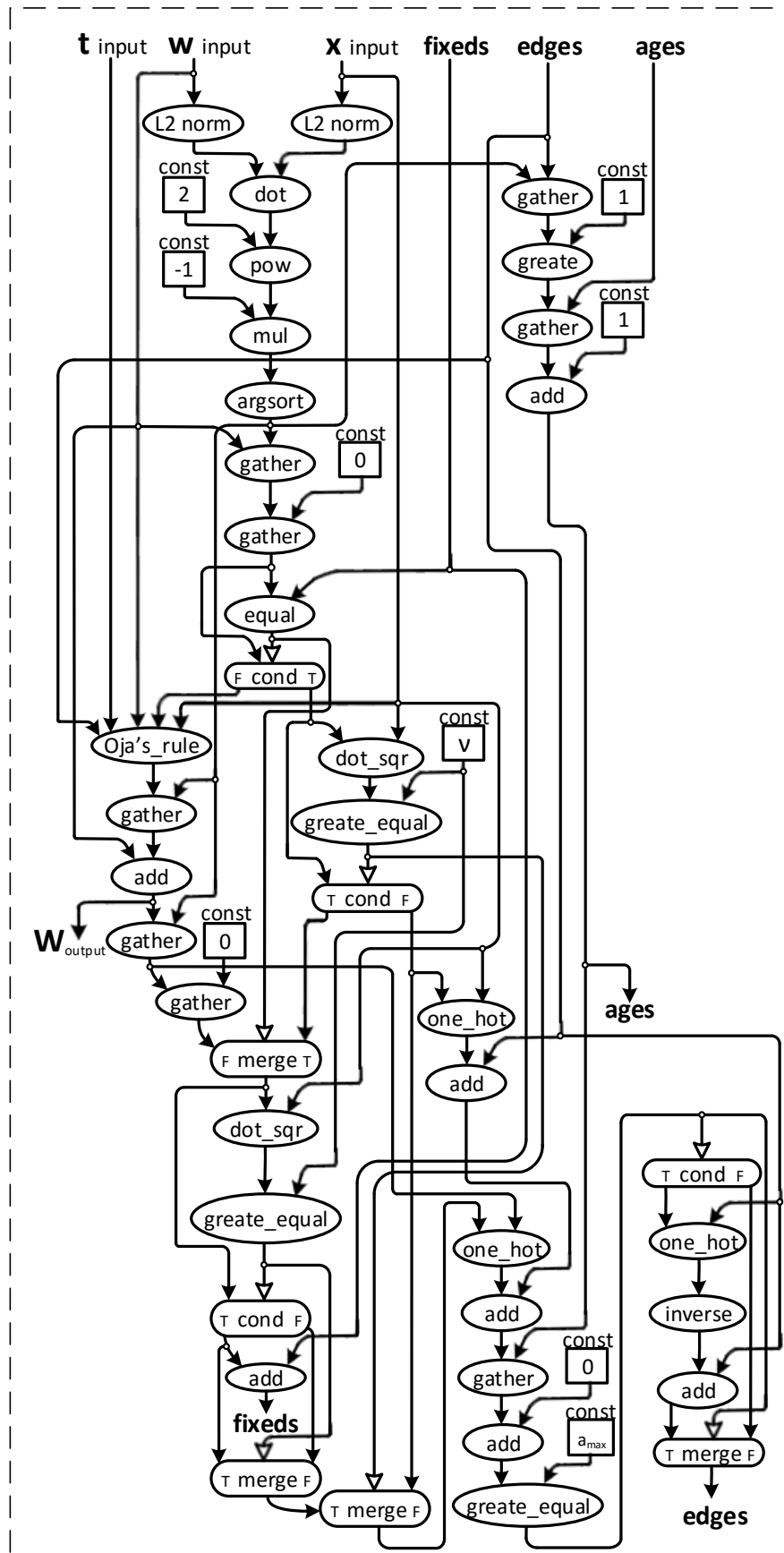


Рисунок 2.6 – Граф обчислень для реалізації однієї ітерації зростаючого розріджено кодучого нейронного газу

На вхід графу надходить лічильник кількості ітерацій  $t_{input}$ , масив (тензор) нейронів  $W_{input}$  з зарезервованою місткістю під максимально допустиму їх кількість, вхідний одномірний вектор  $x_{input}$ , масив прапорців фіксованості нейронів  $fixed$ , матриця суміжності  $edges$  для опису ребер неструктурованої сітки зростаючого нейронного газу, матриця віку ребер  $ages$ . Опис операцій графу обчислень (рис. 2.5 та рис.2.6), що не описані в табл. 2.1, приведено у табл. 2.4.

Таблиця 2.4 – Операції фреймворку Tensorflow для реалізації зростаючого розріджено кодуючого нейронного газу

| Найменування 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            |

Продовження табл. 2.4

| Найменування операції | Короткий опис                                |
|-----------------------|--|
| 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 штучної нейронної мережі призводить до проблеми прокляття розмірності ознак. Для усунення даної проблеми пропонується використовувати аналіз головних компонент. Його використання дозволяє знизити розмірність не беручи до розгляду низькорівневі 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 функції активації кожного пікселя карти ознак за допомогою алгоритму невід'ємного ОУП [15], [29].

Для реалізації ефективного розгортання і розпаралеленого виконання алгоритму невід'ємного ОУП на великому різноманітті обчислювальних платформ розроблено граф обчислень з використанням операцій фреймворку Tensorflow (рис. 2.7). Опис операцій відповідного графу обчислень, що не описані в табл. 2.1 та табл. 2.4, приведено у табл. 2.5.

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 пропонується здійснювати на основі алгоритму симуляції відпалу.

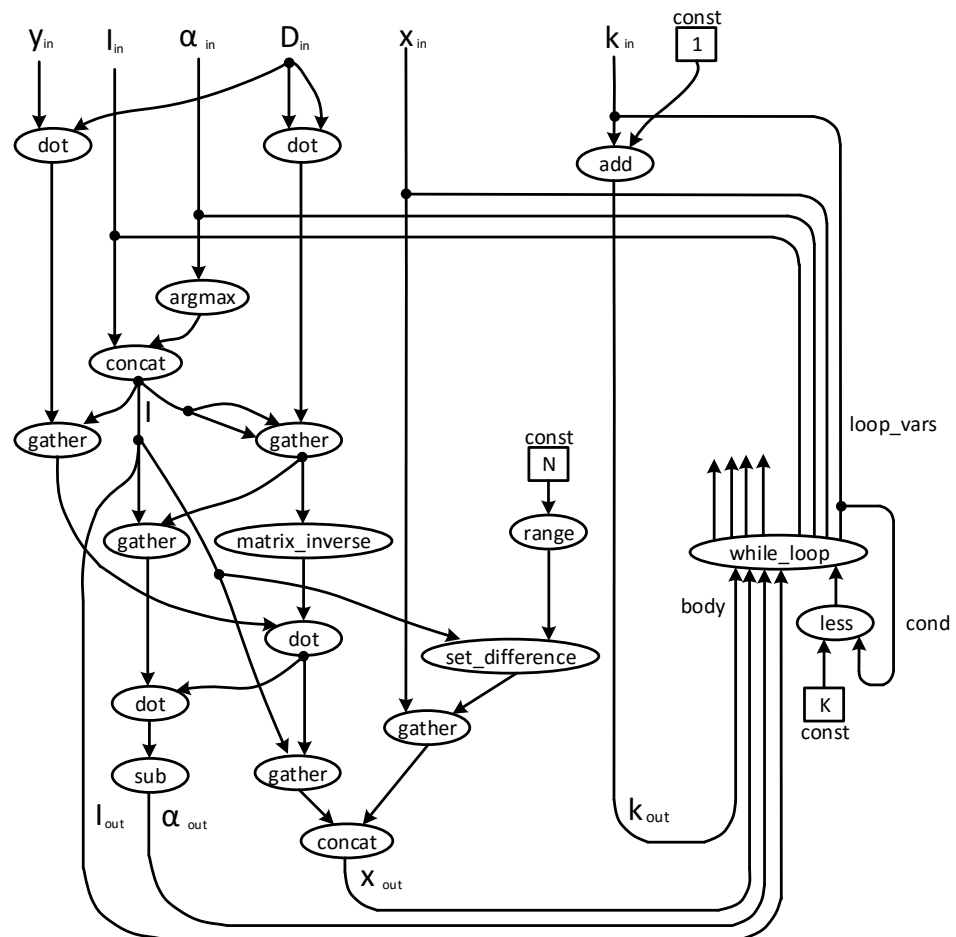


Рисунок 2.7 – Граф обчислень для реалізації однієї ітерації ОУП

Таблиця 2.5 – Операції фреймворку Tensorflow для реалізації ОУП

| 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                  | Поелементна 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)  |

Для навчання детектора об'єктів було використано 200 зображень з розміченого набору даних Inria Aerial Image Labeling Dataset [23]. Кожне зображення має роздільну здатність  $5000 \times 5000$ . При цьому 500 нерозмічених зображень з роздільною здатністю  $224 \times 224$  було згенеровано з використанням техніки випадкового вирізання з поворотом для навчання без вчителя. Також 200 розмічених зображень з роздільною здатністю  $224 \times 224$  було згенеровано для навчання з вчителем. Розмічений набір навчальних даних було аугментовано до 1000 зразків за допомогою додавання до зображень шуму, повороту та вирізання, зміни їхнього контрасту.

В наборі навчальних даних Inria Aerial Image Labeling Dataset представлено велику кількість транспортних засобів у міській місцевості. Транспортні засоби були обрані як об'єкти інтересу, при цьому міська зона розглядається як доменна область застосування. В цьому випадку алфавіт класів розпізнавання дорівнює  $Z=3$ , де перший клас розпізнавання відповідає автомобілям, другий клас розпізнавання відповідає вантажівкам і третій – фоновим зображенням. Розмір об'єкту інтересу на випадковому зображенні знаходиться в діапазоні  $[7 \times 7, \dots, 10 \times 10]$  пікселів.

Відповідно до техніки переносу знань перші 7 fire модулів попередньо навченої згорткової нейронної мережі Squeezenet були запозичені. В результаті кожне зображення було перекодовано в карту ознак з розміром  $13 \times 13 \times 384$ . Наступні шари навчаються без вчителя на нерозміченому наборі навчальних даних з цільової доменної області. При цьому ядра фільтрів рівні  $3 \times 3$ , крок сканування – 1. Вихідна карта ознак сформована шляхом об'єднання карт ознак, сформованих модулями Fire6 та Fire7, а також карти ознак, сформованої останнім згортковим шаром.

Спочатку пропонується виконувати навчання детектора з попередньо навченим останнім шаром без вчителя з використанням модифікації зростаючого нейронного газу без тонкої настройки. При цьому, гіперпараметр алгоритму навчання без вчителя рівний  $v=0.8$ . В алгоритмі навчання інформаційно-екстремального класифікатора пікселів карти ознак число вузлів в деревах рішень

обмежено до 16. При цьому глибина кожного дерева встановлюється на рівні 6.

З метою покращення результатів машинного навчання детектора, інформативність ознакового опису підвищена шляхом використання тонкої настройки високорівневих згорткових шарів. В цьому випадку поточні параметри алгоритму симуляції відпалу були встановлені таким чином:  $c=0.98$ ,  $T_0=10$ ,  $epochs\_max=6000$ ,  $step\_size=0.001$ . На кожному етапі тонкої настройки виконується перенавчання регресора та класифікатора. Для максимізації узагальнюючої здатності моделі та мінімізації обчислювальної складності, було здійснено послідовну оптимізацію гіперпараметра  $\nu$ , що відповідає за щільність розподілу нейронів, з кроком оптимізації 0,1.

Варто розглянути вплив параметрів алгоритму зростаючого розріджено кодуємого нейронного газу, використаного на етапі навчання без вчителя, на результати навчання з учителем. У табл. 2.6 наведено результати машинного навчання та кількість  $N_c$  згенерованих згорткових фільтрів (нейронів) як функцій від параметра  $\nu$ , що характеризує щільність покриття всього набору навчальних даних згортковими фільтрами.

Таблиця 2.6 – Результати навчання без вчителя детектора об'єкти з використанням різних значень параметра  $\nu$

| $\nu$ | $N_c$ | $J_{Cls}$ | $J_{Loc}$ | $J = J_{Cls} * J_{Loc}$ | Відсоток розпізнаних об'єктів на тестовій вибірці, % |
|-------|-------|-----------|-----------|-------------------------|--|
| 0,4   | 196   | 0,0912    | 0,450     | 0,04104                 | 0,67   |
| 0,5   | 337   | 0,1502    | 0,511     | 0,07675                 | 0,73   |
| 0,6   | 589   | 0,2508    | 0,621     | 0,15574                 | 0,87   |
| 0,7   | 889   | 0,4203    | 0,700     | 0,29421                 | 0,93   |
| 0,8   | 1519  | 1,0000    | 0,921     | 0,92100                 | 0,96   |
| 0,9   | 3058  | 1,0000    | 0,923     | 0,92300                 | 0,95   |

Аналіз табл. 2.6 показує, що зі зростанням значення гіперпараметра  $\nu$

зростає кількість нейронів та значення часткових і загального критеріїв оптимізації. При значенні  $v \leq 0.8$  точність моделі за тестовим набором даних зростає зі збільшенням параметра  $v$ , проте подальше збільшення цього параметра призводить до погіршення результатів через перенавчання моделі.

Рис. 2.8 показує залежність інформаційного критерію ефективності навчання (1.7) від радіусу контейнеру для кожного з класів. За даними графіками може бути оцінена точність та завадостійкість побудованих вирішальних правил.

Аналіз рис. 2.8 показує, що отримані вирішальні правила є безпомилковими за навчальним набором даних. Оптимальні радіуси контейнерів класів розпізнавання мають такі значення  $d_1^* = 30$ ,  $d_2^* = 50$  відповідно. Відстань між центрами контейнерів класів розпізнавання дорівнює  $d_{12}^* = 95$ .

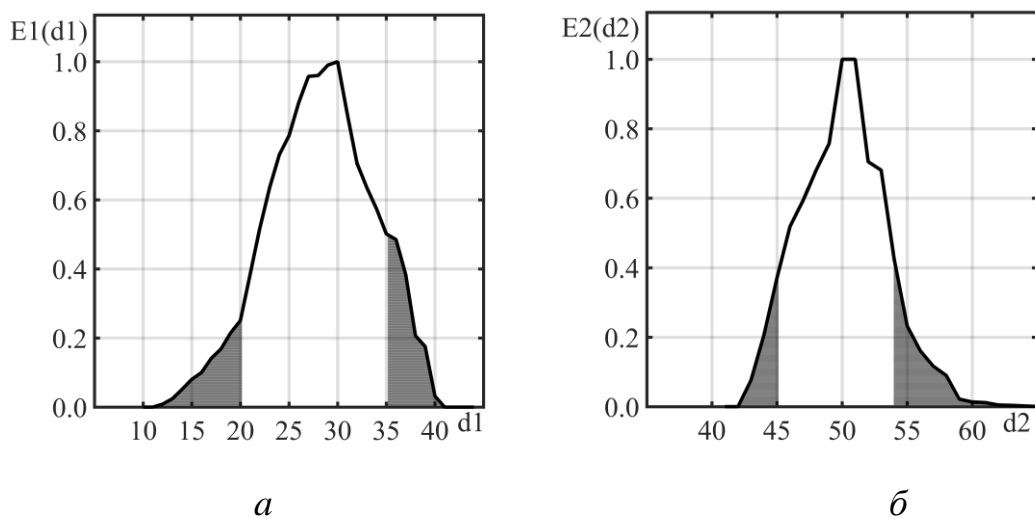


Рисунок 2.8 – Графіки залежності інформаційного критерію ефективності навчання (1.7) від радіусу контейнеру для кожного з класів:

$a$  – клас  $X_1^0$ ;  $b$  – клас  $X_2^0$

Таким чином, було отримано високодостовірні вирішальні правила в умовах обмеженого обсягу навчальних даних, які забезпечують отримання компактного розподілу в просторі ознак та значного міжкласового зазору.

Для оцінки переваг у використанні попереднього навчання без вчителя,



розглянемо результати машинного навчання з використанням симуляції відпалу до і після навчання за алгоритмом зростаючого розріджено кодуючого нейронного газу. На рис. 2.9 показано залежність критерію ефективності навчання детектора  $J = J_{Cls} * J_{Loc}$  за алгоритмом симуляції відпалу з параметрами  $c=0.998$ ,  $T_0=10$ ,  $epochs\_max=5000$ ,  $step\_size=0.001$ .

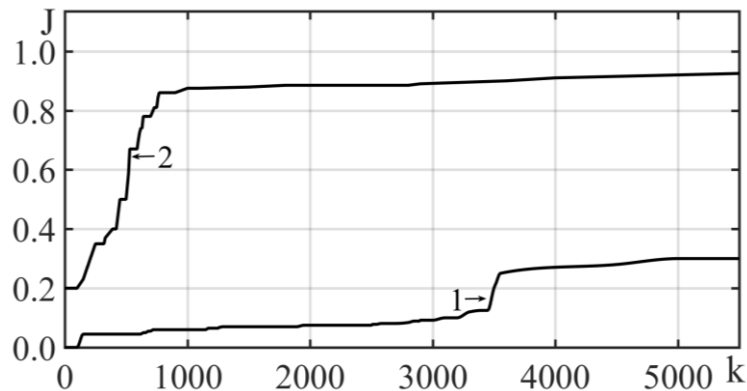


Рисунок 2.9 – Графіки залежності критерію оптимізації від кількості епох навчання : 1 – до застосування попереднього навчання без вчителя; 2 – після застосування попереднього навчання без вчителя

Аналіз рис. 2.9 показує, що попереднє навчання без вчителя на основі зростаючого розріджено кодуючого нейронного газу дозволяє покращити кінцевий результат навчання з учителем за допомогою алгоритму симуляції відпалу. При цьому, якщо попередньо відбувалося навчання без учителя, то досягнення глобального максимуму критерію відбувається більше ніж в 10 разів швидше. Крім того, перевірка отриманих моделей за тестовою вибіркою показує, що використання попереднього навчання без учителя дозволяє зменшити ефект перенавчання в умовах обмеженого обсягу розмічених навчальних даних. При використанні попереднього навчання без вчителя критерій розрахований для навчальної вибірки рівний  $J_{train}=0,921$ , забезпечуючи частку правильно детектованих об'єктів на тестовій вибірці рівну 96%. Без використання попереднього навчання відповідні критерії значно відрізняються,  $J_{train}=0.3011$  з часткою правильно детектованих об'єктів на тестовій вибірці рівній 85%.

Моделювання здійснювалося на одноплатному міні комп'ютері Raspberry Pi 3+. При цьому обробка одного зображення займала всього 0,1 секунду.

Таким чином, запропонований алгоритм попереднього навчання високорівневих шарів без вчителя дозволяє підвищити значення критерію навчання і частки виявлених об'єктів на тестових зображеннях. Крім того вдалося зменшити ефект перенавчання і підвищити швидкість знаходження глобального максимуму при навчанні з учителем на розміченій навчальній вибірці обмеженого обсягу. Однак в рамках даного дослідження не було розглянуто залежність критерію ефективності навчання від параметрів алгоритму симуляції відпалу. Тому наступні дослідження будуть спрямовані на удосконалення моделі детектора та розробку алгоритмів настройки параметрів пошукового алгоритму оптимізації в процесі машинного навчання.

## **2.4 Алгоритм машинного навчання ієрархічного екстрактора ознакового опису для задачі автономної навігації**

Реалізація автономної навігації передбачається на одному чи двох одноплатних 64-х бітних комп'ютерах Raspberry Pi 3+. Це комп'ютер має процесор CPU Broadcom BCM2837B0, Cortex-A53 (ARMv8) з частотою 1,4 ГГц і обсягом оперативної пам'яті рівним 1 Гбайту. 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 BCM2837 Cortex-53 chip.

Програмна реалізація режимів навчання і екзамену здійснюється з використанням бібліотек OpenCV 3,4 та Tensorflow HRT з додатковими модулями (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 ОУП,

дерева рішень, машина екстремального навчання та зростаючий розріджено кодуючий нейронний газ реалізуються з використанням фреймворку Tensorflow як в режимі навчання, так і екзамену.

З метою тестування моделі візуальної навігації в режимі паралельного коректора помилок здійснюється послідовно попередня обробка даних та їх аналіз на основі геометричного методу одометрії ORB-SLAM з обмеженням на максимальну кількість детектованих точок рівній 100. Результати зберігаються в текстовий файл. Після цього запропонована модель (рис. 1.15 та рис. 1.16) навчаються на половині послідовностей з KITTI набору даних [24]. Останнім кроком є запуск моделі в режимі екзамену для створення кінцевого результату з урахуванням результатів ORB-SLAM.

Проектування алгоритму навчання моделі візуальної навігації здійснюється на основі таких основних принципів :

- поєднання принципів конкурентного навчання та розрідженого кодування для зменшення вимог до розміру вибірки навчальних даних та поліпшення конвергенції алгоритму попереднього навчання екстрактора ознак;
- використання вирішальних правил з низькою обчислювальною складністю навчання в режимах навчання та екзамену на основі рандомізованих методів рішення задачі найменших квадратів (методи Грама-Шмідта, псевдоінверсія Мура-Пенроуза) або методи, основані на деревах рішень (Random Forest, Random Subspace, Adaptive boost);
- використання метаевристичної оптимізації для узгодження компонентів гібридної моделі аналізу даних та покращення збіжності до глобального оптимуму критерію ефективності моделі;
- використання комплексного критерію ефективності моделі на основі мультиплікативної агрегації частинних критеріїв ефективності з метою отримання конфігурації екстрактора ознак, що одночасно покращує всі частинні критерії.

Процес машинного навчання навігаційної системи орієнтований на визначення оптимального вектора параметрів  $g$ , що забезпечує максимум комплексного критерію

$$J = \frac{\bar{E}}{E_{\max}} \cdot \frac{\varepsilon_{\min}}{\varepsilon} \cdot \frac{C_{\min}}{C}, \quad (2.4)$$

$$g^* = \arg \max_G \{J(g)\}, \quad (2.5)$$

де  $\bar{E}$  – інформаційний критерій ефективності навчання для розпізнавання перешкод, усереднений за набором класів;

$\varepsilon$  – значення середньо квадратичної похибки регресії при визначенні зміни координат камери у просторі;

$C$  – критерій обчислювальної трудомісткості моделі екстракції ознак;

$E_{\max}$ ,  $\varepsilon_{\min}$ ,  $C_{\min}$  – максимальне можливе значення інформаційного критерію ефективності навчання класифікатора, мінімально допустиме значення похибки регресійної моделі та критерію обчислювальної трудомісткості екстрактора ознак, відповідно;

$G$  – допустима область значень параметрів, які впливають на екстракцію ознак і прийняття рішень.

Пропонується провести навчання екстрактора ознак без вчителя, з використанням розглянутого в попередньої підрозділі модифікації алгоритму зростаючого нейронного газу. Класифікаційний аналіз вихідної карти ознак пропонується здійснювати з використанням розглянутого у попередньому розділі інформаційно-екстремального класифікатора. Так само регресійну модель пропонується реалізовувати на основі ортогональної інкрементальної машини екстремального навчання. Таким чином обрано регресійну модель і класифікатор, які характеризуються високою оперативністю навчання і оснований на послідовному нарощуванні моделі аналізу даних до моменту досягнення необхідної достовірності рішень. Для апроксимації глобального оптимуму комплексного критерію під час навчання вирішальних правил та тонкої настройки екстрактора ознак ми пропонуємо використовувати метаевристичний алгоритм імітації відпалу [36], [37].

Різні значення параметрів  $K_1$  та  $K_2$  впливають як на інформативність ознак,

так і на обчислювальну трудомісткість. Ми пропонуємо вимірювати трудомісткість за кількістю операцій множення (Mul) та додавання (Add), що виконуються під час згорткових операцій з зображенням або картою ознак. Для архітектури мережі, показаної на рис. 1.16, трудомісткість може бути розрахована за формулою

$$C = K_2(2706472K_1 + 4438784K_2) \quad (2.6)$$

Для класифікатора та регресійної моделі оптимальна конфігурація згорткового екстрактора може бути різною, оскільки вона відповідає за різні завдання. Тому комплексний критерій (2.4) пропонує компроміс з точки зору точності правил прийняття рішень та обчислювальної складності екстрактора візуальних ознак.

Множина класів розпізнавання  $\{X_z^o\}$  описує характерні перешкоди та відповідні команди реакції і має потужність  $Z = 5$ . Перший клас розпізнавання  $X_1^o$  характеризує рух вперед без повороту. Класи  $X_2^o$  і  $X_3^o$  відповідають лівому повороту 45 і 90 градусів відповідно. Класи  $X_4^o$  і  $X_5^o$  відповідають правому повороту 45 і 90 градусів відповідно. Обсяг навчальних та тестових даних кожного класу  $n_z = 300$ .

Спочатку пропонується навчати вирішальні правила моделі, використовуючи попередньо навчений без вчителя екстрактор ознак за допомогою зростаючого розріджено кодуючого нейронного газу без тонкої настройки. При цьому під час навчання без вчителя використовуються три фіксованих значення параметра реконструкції набору навчальних даних  $v$ . Цей параметр безпосередньо впливає на кількість каналів карт ознак  $K_2$ .

Для покращення результатів машинного навчання візуальної навігаційної системи підвищують інформативність ознакового опису шляхом тонкої настройки навчених без вчителя згорткових фільтрів. У цьому випадку використовувались наступні параметри алгоритму симуляції відпалу:  $c = 0.98$ ,  $T_0 = 10$ ,  $\text{epochs\_max} =$

5000, step\_size = 0,001. На кожному кроці для тонкої настройки передбачається перепідготовка моделі регресії та класифікатора.

Для виявлення тенденції зміни середніх значень часткових та комплексного критеріїв при зростанні параметрів  $K_1$  та  $K_2$ , що впливають на розмір екстрактора ознак (рис. 1.16), було виконано симуляцію для трьох фіксованих значень кожного з цих параметрів (табл. 2.2). При цьому оптимальні значення цих параметрів визначаються для відкритого набору даних КІТТІ-07.

Таблиця 2.7 – Залежність часткових та комплексного критерію від параметрів екстрактора ознакового опису  $K_1$  і  $K_2$

| $\nu$ | $K_1$ | $K_2$ | $\bar{E} / E_{\max}$ | $\varepsilon_{\min} / \varepsilon$ | $C_{\min} / C$ | $J$      |
|-------|-------|-------|----------------------|------------------------------------|----------------|----------|
| 0.6   | 3     | 180   | 0,083                | 0,112                              | 1,000          | 0,009296 |
| 0.6   | 5     | 180   | 0,101                | 0,188                              | 0,827          | 0,015703 |
| 0.6   | 7     | 180   | 0,098                | 0,200                              | 0,705          | 0,013818 |
| 0.7   | 3     | 210   | 0,28                 | 0,688                              | 0,297          | 0,057214 |
| 0.7   | 5     | 210   | 0,29                 | 0,756                              | 0,264          | 0,057879 |
| 0.7   | 7     | 210   | 0,29                 | 0,775                              | 0,238          | 0,053491 |
| 0.8   | 3     | 460   | 0,39                 | 0,968                              | 0,082          | 0,030957 |
| 0.8   | 5     | 460   | 0,55                 | 1,000                              | 0,077          | 0,04235  |
| 0.8   | 7     | 460   | 0,51                 | 1,000                              | 0,072          | 0,03672  |

Аналіз табл. 2.7 показує, що збільшення значень параметрів  $K_1$  і  $K_2$  загалом призводить до збільшення достовірності та обчислювальної складності (2.6) вирішальних правил класифікатора та регресійної моделі. У той же час, збільшення параметра  $K_1$  мало впливає на ефективність класифікатора внаслідок зниження ефективності пошукового алгоритму зі значним збільшенням розміру простору пошуку, тоді як помилка регресії однаково чутлива до значення параметрів  $K_1$  і  $K_2$ .

Однак, з огляду на те, що при зростанні  $K_1$  та  $K_2$  достовірність правил

прийняття рішень зростає повільніше, ніж обчислювальна складність, то використання комплексного критерію  $J$  забезпечує компромісне рішення. Тобто ми вважаємо наступні значення параметрів  $K_1^* = 5$  та  $K_2^* = 21$  оптимальними.

У оптимальній конфігурації екстрактора ознак середнє значення інформаційного критерію функціональної ефективності рівне  $\bar{E} = 0,29$ . Це відповідає точності 95,2% для навчального набору та 94% для тестового набору.

На рис. 2.10 показано графік зміни середнього значення інформаційного критерію ефективності (1.5) відносно кількості ітерацій алгоритму імітації відпалу.

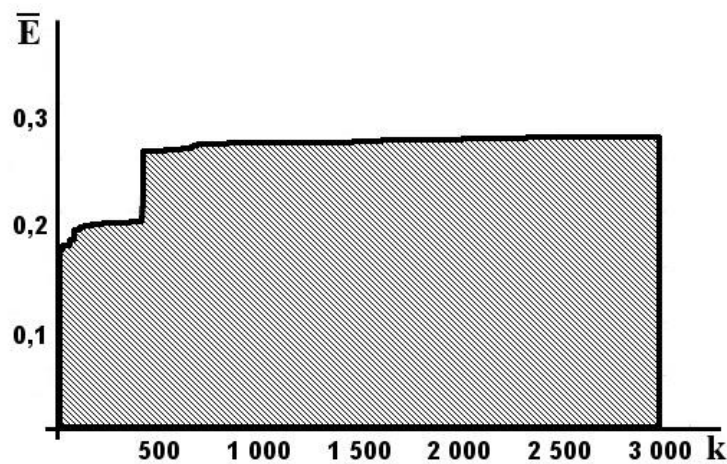


Рисунок 2.10 – Графік зміни усередненого значення інформаційного критерію ефективності (1.5) залежно від кількості ітерацій алгоритму імітації відпалу

Аналіз рис. 2.10 показує, що після 1000-ї ітерації зростання усередненого за алфавітом класів інформаційного критерію (1.5) почало сповільнюватися, а після 2500-ї ітерації практично не змінилося. Така зміна критерію вказує на те, що подальше збільшення інформаційного критерію досягне лише за рахунок збільшення інформативності ознакового опису або шляхом збільшення значень  $K_1$  та  $K_2$ , або шляхом вдосконалення структури екстрактора (рис. 1.16).

Для візуальної оцінки ефективності машинного навчання навігаційної системи, реальна траєкторія, виміряна за допомогою GPS та LiDaR, може бути

порівняна з реконструйованою траєкторією, отриманою з використанням навченої моделі. На рис. 2.11а показано результат використання одометрії на основі локальних ознак з системи ORB-SLAM [24], [25]. Максимальна кількість ключових точок обмежена до 100, щоб забезпечити необхідну частоту кадрів у випадку бортової системи без застосування обчислювальних можливостей графічного процесора.

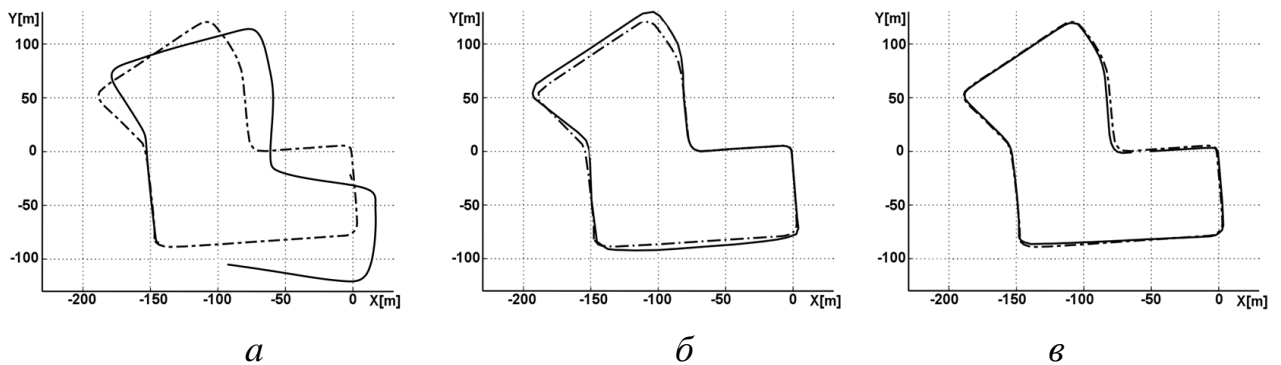


Рисунок 2.11 – Реальна траєкторія (пунктирна лінія) та реконструйована (суцільна лінія) для KITTI-07 [20]: *а* – монокулярна одометрія на основі локальних ознак; *б* – запропонована модель; *в* – паралельна корекція помилок для монокулярної одометрії на основі локальних ознак з використанням запропонованої моделі

Аналіз рис. 2.11а показує, що реконструйована траєкторія має значну похибку. На рис. 2.11б показано результат використання запропонованої моделі, навченої та перевіреної на тих самих наборах даних. У цьому випадку помилка реконструкції набагато менша. На рис. 2.11в показано результат використання одометрії на основі локальних ознак від ORB-SLAM з паралельною корекцією на основі запропонованої моделі. Як бачите, точність реконструкції висока.

В табл. 2.8 показано, що запропонований підхід дозволяє отримати кращі під час результати реконструкції траєкторії порівняно з добре відомим ORB-SLAM. Зокрема, для послідовності Sequence 7 з KITTI набору даних вдалося скоротити помилку оцінки переміщення приблизно на 65.6% при тій же частоті кадрів (10 кадрів за секунду). Тестування відбувалося на одноплатному комп'ютері Raspberry Pi 3+.



Таблиця 2.8 – Усереднені результати на KITTI наборі даних

| Послідовність | Запропонована модель |                         | ORB-SLAM з обмеженням до 100 ключових точок на кадр |                         |
|---------------|----------------------|-------------------------|---|-------------------------|
|               | $t$ , %              | $\theta$ , градус/100 м | $t$ , %   | $\theta$ , градус/100 м |
| KITTI -07     | 1.57                 | 2.60                    | 2.24  | 7.56                    |
| KITTI -09     | 1.49                 | 2.12                    | 1.64  | 6.89                    |

На рис. 2.12а показано, що навчена модель на наборі даних KITTI-07 має низьку продуктивність при тестуванні на іншому наборі даних. На рис. 2.12б показано, що повторне навчання моделі на новому наборі даних значно підвищило точність реконструкції траєкторії з набору даних.

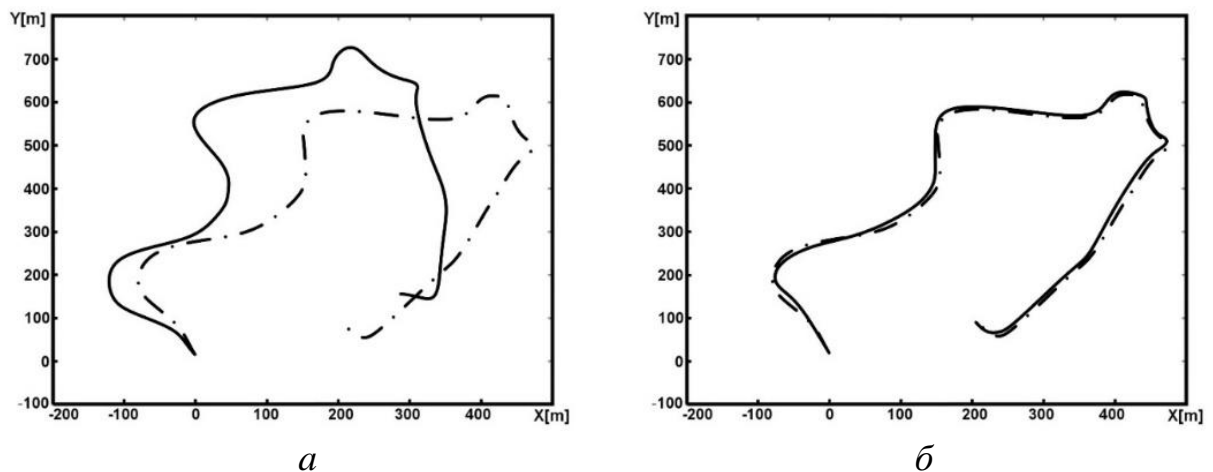


Рисунок 2.12 – Реальна траєкторія (пунктирна лінія) і реконструйована (суцільна лінія) траєкторії для KITTI-09 [20]: а – навчена модель на KITTI-07; б – після перенавчання на KITTI-09

Таким чином, результати, отримані на відкритих наборах даних, свідчать про придатність моделі для практичного використання. Щоб уникнути перенавчання, оптимізована складність моделі. Але обмежена ємність моделі викликає потребу повторного навчання при зміні доменної області використання.

Для незалежного тестування навченої моделі було використано послідовності набору TUM, відзняті на відкритому повітрі. Для порівняння ORB-SLAM тестувався на тих же послідовностях з обмеженням на кількість ключових

точок рівню 100 (табл. 2.9).

Таблиця 2.9 – Усереднені результати на TUM наборі даних

| Послідовність | Запропонована модель |                         | ORB-SLAM з обмеженням до 100 ключових точок на кадр |                         |
|---------------|----------------------|-------------------------|---|-------------------------|
|               | $t$ , %              | $\theta$ , градус/100 м | $t$ , %   | $\theta$ , градус/100 м |
| TUM-outdoors5 | 4.31                 | 3.60                    | 6.24  | 7.56                    |
| TUM-outdoors6 | 5.42                 | 3.68                    | 7.64  | 8.11                    |
| TUM-outdoors7 | 4.33                 | 3.59                    | 6.44  | 7.93                    |
| TUM-outdoors8 | 5.01                 | 3.61                    | 7.12  | 7.99                    |

Аналіз табл. 2.9 показує, що помилка оцінки переміщення в ORB-SLAM для послідовностей TUM перевищує похибку оцінки переміщення в запропонованій моделі більш ніж в 1,4 рази. Помилка оцінки повороту в ORB-SLAM перевищує відповідну похибку оцінки повороту в запропонованій моделі більш ніж у 2 рази. Однак, точність запропонованої моделі за результатами тестування на незалежній послідовності нижче, ніж на послідовностях KITTI наборі даних, який брав участь під час навчання. Це могло бути викликано ще тим, що на початку TUM послідовності відеоматеріал відзняється в приміщенні, що дуже відрізняється від навчальних даних.

Таким чином, запропонована модель і алгоритм навчання автономної навігаційної системи БПЛА були перевірені на відкритих наборах даних. Випробування на незалежній послідовності TUM, що була відзнята на вулиці, дає похибку оцінки переміщення, що не перевищує 6%, а похибка оцінки повороту не перевищує 3,68 градусів на 100 м. Точність може бути додатково покращена додатковим навчанням на зразках подібних послідовностей.

## ВИСНОВКИ

У науково-дослідній роботі розв'язано важливу науково-технічну прикладну задачу аналізу і синтезу екстрактора ознакового опису аеро- відео-спостережень для задач автономної навігації та детектування об'єктів на місцевості. Головні наукові та практичні результати роботи полягають в наступному.

1. Розроблено нові моделі однорівневої та ієрархічної екстракції ознакового опису середовища для задачі навігації та детектування об'єктів на місцевості. Їх архітектура дозволяє підвищити точність вирішальних правил за рахунок врахування більшої кількості контекстної інформації порівняно з аналогами.

2. Запропоновано модель екстрактора ознак для задачі візуальної навігації, що складається з 4-х згорткових розріджено кодуєчих шарів, кожен з яких містить різномасштабні згорткові фільтри. На вхід модель приймає серію послідовних кадрів, що обробляються одночасно. Отримана модель здатна здійснювати оцінку переміщення з частотою 10Гц на одноплатному міні-комп'ютері Raspberry Pi 3+. Навчена запропонована модель дозволяє підвищити на 65% точність реконструкції траєкторії порівняно з моделлю ORB-SLAM, побудованої в рамках геометричного підходу. Крім того випробування алгоритмів на незалежній послідовності TUM, відзнятій на відкритому повітрі, показує, що похибка оцінки переміщення не перевищує 6%, а похибка оцінки повороту не перевищує 3,68 градусів на 100 м.

3. Запропоновано модель однорівневого екстрактора ознак для задачі детектування об'єктів на місцевості у вигляді гістограм частоти появи візуальних слів та їх пар. Також запропоновано кодування ключових фрагментів зображення, екстракція яких здійснюється на основі методу SURF, як за мінімумом косинусної міри відстані до відповідного візуального слова із навченого словника, так і у вигляді комбінації візуальних слів, отриманих в результаті розрідженого кодування. Для побудови гістограми пропонується використовувати Average-pooling та L2-нормалізацію. При цьому утворювати пари можуть лише ті слова,

що знаходяться у сусідніх комірках, на які рівномірно розбивається вікно сканування зображення. Фізичне моделювання на одноплатному міні комп'ютері Raspberry Pi 3+ показало, що для детектування об'єктів, які займають приблизно 100x100 пікселів на зображенні з роздільною здатністю Full-HD, можна досягти нуля хибних спрацювань при менше ніж 1% пропусків. При цьому частота кадрів становить 5 кадрів за секунду.

4. Запропоновано модель ієрархічного екстрактора ознак для задачі детектування малорозмірних об'єктів на місцевості з високою варіативністю спостережень. Модель складається з 7-ми перших модулів обчислювально ефективної згорткової мережі Squeezenet, одного згорткового розріджено кодуючого шару, шару об'єднання з масштабуванням, перетворювача головних компонентів. Такий підхід забезпечує використання більшої кількості контекстної інформації порівняно з традиційною структурою згорткового екстрактора ознак, що дозволяє досягти 96% точності розпізнавання на тестових зображеннях набору Inria Aerial Image Labeling dataset. При цьому розроблена модель забезпечує прийнятну для практичного застосування точність ідентифікації об'єктів, розмір яких в 7-14 разів менший за розмір найменшої сторони аерофотознімку. При цьому модель може обробляти зображення з частотою 10 кадрів за секунду на одноплатному міні комп'ютері Raspberry Pi 3+.

5. Розроблено категорійно-функтурні та функціональні діаграми процесів навчання однорівневого та ієрархічного екстрактора ознак для задач візуальної навігації та детектування об'єктів на місцевості. Розроблені діаграми дозволяють на етапі системного аналізу не тільки встановлювати відношення між елементами інформаційного забезпечення та інформаційними потоками оброблення інформації, але і суттєво полегшують реінжиніринг, рефакторинг та інші етапи розробки інформаційного забезпечення БПЛА.

6. Розроблено алгоритм зростаючого розріджено кодуючого нейронного газу, оснований на схемі м'якої конкуренції, правилі Ойа та вставці нових нейронів за порогом досяжності, для навчання верхніх шарів ієрархічного екстрактора ознакового опису. Це дозволяє ефективно використовувати

нерозмічені навчальні дані, автоматично визначати необхідну кількість нейронів та забезпечити близький до оптимального розподіл нейронів по даним. При цьому показано, що використання зростаючого розріджено кодуючого нейронного газу дозволяє пришвидшити більше ніж в 10 разів пошук оптимуму критерія ефективності навчання, близького до глобального.

7. Удосконалено алгоритм тонкої настройки параметрів моделі шляхом використання симуляції відпалу для пошуку глобального оптимуму комплексного критерію, що враховує ефективність вирішальних правил та обчислювальну складність екстрактора ознак. На відміну від алгоритму зворотного поширення помилки, у запропонованому алгоритмі використовується мультиплікативна згортка прямих критеріїв та метаевристичний пошук, що покращує конвергенцію до глобального оптимуму. Для прискорення обчислень регресійні вирішальні правила будуються на основі машини екстремального навчання, а класифікаційні вирішальні правила будуються на основі інформаційно-екстремальної моделі. При цьому інформаційно-екстремальну модель запропоновано будувати на основі ансамблю дерев рішень для двійкового кодування спостережень та радіально-базисних функцій для розділення класів у двійковому просторі Хеммінга.

8. Практичне значення отриманих результатів для безпілотної авіації полягає у формуванні сучасної наукової та методологічної основи для проектування здатних до навчання систем автономної навігації та детектування об'єктів на місцевості для малогабаритних літальних апаратів, що мають ресурсні та інформаційні обмеження. Запропоновані моделі і алгоритми машинного навчання дозволяють знизити вимоги до обсягу навчальних даних та обчислювальних ресурсів, що дозволяє знизити габарити апарату і підвищити оперативність адаптації до нових умов функціонування.

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Moskalenko V. Improving the effectiveness of training the on-board object detection system for a compact unmanned aerial vehicle / V. Moskalenko, S. Dovbysh, I. Naumenko, A. Moskalenko, A. Korobov // Eastern-European Journal of Enterprise Technologies. – 2018. – No 4/9(94). – P. 19–26. – DOI: 10.15587/1729-4061.2018.139923.

2. Москаленко В. В. Моделі і методи інтелектуальної інформаційної технології автономної навігації для малогабаритних безпілотних апаратів / В. В. Москаленко, А. С. Москаленко, А. Г. Коробов // Радіоелектроніка, інформатика, управління. – 2018. – № 3. – С. 68–77. – DOI: 10.15588/1607-3274-2018-3-8.

3. Moskalenko V. V. A model and training method of small-sized object detection system for a compact aerial drone / V. V. Moskalenko, A. S. Moskalenko, Korobov A. G., Zaretsky N. O. // Radio Electronics, Computer Science, Control Journal. – 2019. – Vol. 1.

4. Moskalenko V. V. The Model and Training Algorithm of Compact Drone Autonomous Visual Navigation System / V. V. Moskalenko, A. S. Moskalenko, A. G. Korobov, V.A. Semashko // Data Journal. – 2019.

5. Москаленко В. В. Модель та алгоритм навчання системи детектування малорозмірних об'єктів для малогабаритних безпілотних літальних апаратів / В. В. Москаленко, А. С. Москаленко, А. Г. Коробов, М. О. Зарецький, В. А. Семашко // Радіоелектронні і комп'ютерні системи. – 2018. – № 4. – С. 15–23.

6. Москаленко В. В. Модель і алгоритм навчання детектора шкідливого трафіку на основі модифікації зростаючого нейронного газу. / В. В. Москаленко, А. С. Москаленко, М. О. Зарецький // Радіоелектронні і комп'ютерні системи. – 2018. – № 3. – С. 260–271

7. Moskalenko V. Model and Training Methods of Autonomous Navigation System for Compact Drones. / V. Moskalenko, A. Moskalenko, A. Korobov, O. Boiko,

S. Martynenko, O. Borovenskyi // Proceedings of the IEEE Second International Conference on Data Stream Mining & Processing (DSMP) (Lviv, Ukraine, 21–25 August 2018). – Lviv Polytechnic Publishing House: Lviv, Ukraine, 2018. – P. 503–508. – DOI:10.1109/DSMP.2018.8478521.

8. Korobov A. G. Parameters optimization method of the information-extreme object recognition system on the terrain / A. G. Korobov, A. S. Moskalenko, V. V. Nahornyi, J. Zavgorodnya, O. B. Berest, V. O. Zhurba // Proceeding of the IEEE First International Conference on System Analysis & Intelligent Computing (SAIC), 08–12 October, 2018 Kyiv, Ukraine. – P 231–235.

9. Moskalenko V. V. Growing Convolutional Neural Network For Malware Traffic Detection / V. V. Moskalenko, A. S. Moskalenko, // Information and Telecommunication Technologies and Radio Electronics: proceedings of the Third IEEE International Conference (UkrMiCo), 10–14 Sept., 2018 Odessa, Ukraine. – P. 206–211.

10. V. Moskalenko V. V. Object Detection Based On Growing Convolutional Neural Network for Autonomous Systems / V. V. Moskalenko, A. S. Moskalenko, A. G. Korobov, B. Lypivets // Advanced Information Systems and Technologies: proceedings of the VI-th International Conference (AIST), 16–18 May 2018 Sumy, Ukraine. – P. 110–114.

11. Москаленко В. В. Метод самонавчання згорткового екстрактора ознак для розпізнавання багатоканальних діагностичних зображень / В. В. Москаленко, А. С. Москаленко, А. М. Кудрявцев // Інтелектуальні системи прийняття рішень та проблеми обчислювального інтелекту: матеріали міжнародної наукової конференції, 21–27 трав. 2018 м. Залізний Порт, Україна. – 2018. – С. 258–260.

12. Коробов А. Г. Автономна бортова система детектування об'єктів на місцевості. / А. Г. Коробов // Інтелектуальні системи прийняття рішень та проблеми обчислювального інтелекту: матеріали міжнародної наукової конференції, 21–27 трав. 2018 м. Залізний Порт, Україна. – 2018. – С. 236–238.

13. Cadena C. Past, Present, and Future of Simultaneous Localization And Mapping: Towards the Robust-Perception Age / C. Cadena, L. Carlone, H. Carrillo,

Y. Latif, D. Scaramuzza, et al. // IEEE Transactions on Robotics. – 2016. – No. 32(6). – P. 1309–1332. – DOI: 10.1109/TRO.2016.2624754.

14. Smolyanskiy N. Toward Low-Flying Autonomous MAV Trail Navigation using Deep Neural Networks for Environmental Awareness / N. Smolyanskiy, A. Kamenev, J. Smith, S. T. Birchfield // CoRR. – 2017. – V. abs/1705.02550. – Available at: <https://arxiv.org/pdf/1705.02550.pdf>.

15. Bengio Y. Representation Learning: A Review and New Perspectives. / Y. Bengio, A. Courville, P. Vincent // CoRR. – 2014. – V. abs/1206.5538. – Available at: <https://arxiv.org/pdf/1206.5538.pdf>.

16. Schmidhuber J. Deep Learning in Neural Networks: an Overview / J. Schmidhuber // Neural Networks. – 2015. – Vol. 61. – P. 85–117. – DOI:10.1016/j.neunet.2014.09.003.

17. Antipov G. Learned vs. Hand-Crafted Features for Pedestrian Gender Recognition. / G. Antipov, S. Berrani, N. Ruchaud, J. Dugelay // Proceedings of the 23rd ACM International Conference on Multimedia, Brisbane, 26–30 Oct., 2015 Australia. – P. 1263–1266. – DOI: 10.1145/2733373.2806332.

18. Yongtao Y. Bag-of-visual-phrases and hierarchical deep models for traffic sign detection and recognition in mobile laser scanning data / Y. Yongtao, L. Jonathan, W. Chenglu, G. Haiyan, L. Huan, W. Cheng // ISPRS Journal of Photogrammetry and Remote Sensing. – 2016. – Vol. 113. – P. 106–123. – DOI: 10.1016/j.isprsjprs.2016.01.005.

19. Folkesson J. A Feature Based Navigation System for an Autonomous Underwater Robot / J. Folkesson, J. Leederkerken, R. Williams, A. Patrikalakis, J. A. Leonard // Proceedings of the sixth edition Field and Service Robotics, 9–12 July, 2007 Chamonix, France. –2007. – P. 105–114. – DOI:10.1007/978-3-540-75404-6\_10.

20. Wu A. D. Autonomous Flight in GPS-Denied Environments Using Monocular Vision and Inertial Sensors / A. D. Wu, E. N. Johnson, M. Kaess, F. Dellaert // Journal of Aerospace Information Systems. – 2013. – Vol. 10. – No 4. – P. 172–186. – DOI:10.2514/1.I010023.

21. Nguyen A., Yosinski J., Clune J. Deep neural networks are easily fooled:



High confidence predictions for unrecognizable images / A. Nguyen, J. Yosinski, J. Clune // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, (CVPR), Boston, MA, 7–12 June, 2015. – P. 427–436. – DOI: 10.1109/CVPR.2015.7298640.

22. Mhaskar H. N. Deep vs. Shallow Networks: an Approximation Theory Perspective / H. N. Mhaskar, T. Poggio // CoRR. – 2016. – V. abs1608.03287. Available at: <https://arxiv.org/pdf/1608.03287.pdf>.

23. Ammour N. Deep Learning Approach for Car Detection in UAV Imagery / N. Ammour, H. Alhichri, Y. Bazi, B. Benjdira, et al. // Remote Sens. – 2017. – V. 9. – I.4. – P. 2–15.

24. Mohanty V. DeepVO: A Deep Learning approach for Monocular Visual Odometry / V. Mohanty, S. Agrawal, S. Datta, A. Ghosh et al. // CoRR. – 2016. – V. abs/1611.06069. – Available at: <https://arxiv.org/pdf/1611.06069.pdf>.

25. Kim D. K. Deep Neural Network for Real-Time Autonomous Indoor Navigation / D. K. Kim, T. Chen // CoRR. – 2015. – V. abs/1511.04668v2. – Available at: <https://arxiv.org/pdf/1511.04668v2.pdf> (date of request 16.12.2018).

26. Ayoul Th., Buckley T., Crevier F. UAV Navigation above Roads Using Convolutional Neural Networks. – Available at: <http://cs231n.stanford.edu/reports/2017/pdfs/553.pdf>.

27. Subbotin S. The special deep neural network for stationary signal spectra classification / S. Subbotin // Proceedings of the 14th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET), 20–24 Feb. 2018 Lviv-Slavske, Ukraine. – 2018. – P. 123–128.

28. Mrazova I. Image Classification with Growing Neural Networks / I. Mrazova, M. Kukacka // International Journal of Computer Theory and Engineering. – 2013. – Vol. 5. – N. 3. – P. 422–427. – DOI:10.7763/IJCTE.2013.V5.722.

29. Labusch K. Sparse coding neural gas: learning of overcomplete data representations / K. Labusch, E. Barth, T. Martinetz // Neurocomputing. – 2009. – Vol. 72, I. 7–9. – P. 1547–1555. – DOI:10.1016/j.neucom.2008.11.027.

30. Palomo J. The Growing Hierarchical Neural Gas Self-Organizing Neural

Network / J. Palomo , E. López-Rubio // IEEE Transactions on Neural Networks and Learning Systems. – 2017. – Vol. 28, No. 9. – P. 2000–2009. – DOI:10.1109/TNNLS.2016.2570124.

31. Москаленко В. В. Інформаційно-екстремальний алгоритм функціонування системи розпізнавання об'єктів на місцевості з оптимізацією параметрів екстрактора ознак / В. В. Москаленко, А. Г. Коробов // Радіоелектроніка, інформатика, управління. – 2017. – № 2. – С. 38–45. – DOI: 10.15588/1607-3274-2017-2-7.

32. Москаленко В. В. Метод навчання без вчителя ієрархічного екстрактора візуальних ознак на основі модифікації нейронного газу / В. В. Москаленко, А. С. Москаленко, А. Г. Коробов // Радіоелектронні та комп'ютерні системи. – 2017. – № 3 (83). – С. 56–61.

33. Moskalenko V. V. Information-extreme algorithm of the system for recognition of objects on the terrain with optimization parameter feature extraction / V. V. Moskalenko, A. G. Korobov // Radio Electronics, Computer Science, Control. – 2017. – № 2. – С. 38–45. – DOI 10.15588/1607-3274-2017-2-7.

34. Vens C. Random Forest Based Feature Induction. / C. Vens, F. Costa // Proceedings of the IEEE 11th International Conference on Data Mining, Vancouver Canada, 11–14 Dec, 2011. Piscataway, NJ, 2011. – P. 744–753. – DOI: 10.1109/ICDM.2011.121.

35. Zou W. Fault Diagnosis of Tennessee-Eastman Process Using Orthogonal Incremental Extreme Learning Machine Based on Driving Amount / W. Zou, Y. Xia, H. Li // IEEE Transactions on Cybernetics. – 2018. – P. 1–8. – DOI: 10.1109/TCYB.2018.2830338.

36. Ayumi V. Optimization of convolutional neural network using microcanonical annealing algorithm / V. Ayumi, L. M. R. Rere, M. I. Fanany, A. M. Arymurthy // Proceedings of the International on Conference Advanced Computer Science and Information Systems (ICACISIS), Malang, Indonesia, 15–16 Oct. 2016. – Institute of Electrical and Electronics Engineers, 2016. – P. 506–511. – DOI: 10.1109/ICACISIS.2016.7872787.

37. Rere R. L. M. Metaheuristic Algorithms for Convolution Neural Network / R. L. M. Rere, M. I. Fanany, A. M. Arymurthy // Computational Intelligence and Neuroscience. – 2017. – Vol. 2016. – P. 1–13. – DOI:10.1155/2016/1537325
38. Moskalenko V. V. Development of the method of unsupervised training of convolutional neural networks based on neural gas modification. / V. V. Moskalenko // Proc. of the International scientific and practical conference «Technology Transfer: fundamental principles and innovative technical solutions: Physical Sciences and Engineering» Tallinn, Estonia, 23.11.2017. – 2017. – P. 21–27.
39. Коробов А. Г. Автономна система навігації безпілотного літального апарату / Коробов А. Г. // Матеріал доповіді на IV Міжнародній науково-технічній конференції “Інформатика, управління і штучний інтелект”, м. Харків, 21 – 23 листопада 2017. – 2017. – С. 10.
40. Comaschi F RASW: A run-time adaptive sliding window to improve Viola-Jones object detection / F. Comaschi, S Stuijk, T Basten, H Corporaal // Proceedings of the Seventh International Conference on Distributed Smart Cameras, Palm Springs, California, USA, October 29. – Nov. 1. 2013. – IEEE Washington DC, USA, 2013.