Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Комп‘ютерних наук
Кафедра Програмної інженерії

АТЕСТАЦІЙНА РОБОТА (ПРОЕКТ)

Пояснювальна записка

другий (магістерський)

«Дослідження ефективності моделей класифікації зображень»

Виконав: студент 2 курсу, групи ПЗСм-18-1
спеціальності
121 – Інженерія програмного забезпечення
освітньо-професійної програми
Програмне забезпечення систем
Грідін І. В.

Керівник: проф. Смеляков К. С.

Допускається до захисту
Зав. кафедри, проф. _____ Дудар З. В.
(підпис)

2019 р.

Харківський національний університет радіоелектроніки

Факультет Комп‘ютерних наук

Кафедра Програмної інженерії

Рівень вищої освіти другий (магістерський)

Спеціальність 121 – Інженерія програмного забезпечення

Тип програми освітньо-професійна програма

Освітня програма Програмне забезпечення систем

ЗАТВЕРДЖУЮ:

Зав. кафедри _____

(підпис)

«___»_____ 2019 р.

ЗАВДАННЯ

НА АТЕСТАЦІЙНУ РОБОТУ

студентові Грідіну Ігорю Володимировичу

1. Тема роботи проекту «Дослідження ефективності моделей класифікації зображень» затверджена наказом по університету від «__._» _____ 2019р. № _____

2. Термін подання студентом роботи (проекту): _____ 2019р.

3. Вихідні дані до роботи: пояснювальна записка.

4. Перелік питань, що потрібно опрацювати в роботі: мета роботи, аналітичний огляд методів розпізнавання зображень, аналіз нейронних мереж для класифікації зображень, експериментальне дослідження ефективності роботи нейронних мереж для класифікації зображень.

5. Консультанти розділів роботи

Найменування розділу	Консультант (посада, прізвище, ім’я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата
Аналіз предметної галузі	проф. Смеляков К. С.		

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи (проекту)	Термін виконання етапів проекту (роботи)	Примітка
1.	Аналіз предметної галузі		
2.	Огляд існуючих методів та моделей		
3.	Експериментальне дослідження ефективності існуючих моделей		
4.	Підготовка пояснювальної записки		
5.	Підготовка презентації та доповіді		
6.	Попередній захист		
7.	Нормоконтроль, рецензування		
8.	Занесення диплома в електронний архів		
9.	Допуск до захисту у зав. кафедри		

Дата видачі завдання: «____»_____ 2019 р.

Студент _____ Грідін І. В.

(підпис)

Керівник роботи _____ проф. Смеляков К. С.

РЕФЕРАТ / ABSTRACT

Пояснювальна записка до атестаційної магістерської роботи містить 51 сторінку, 21 рисунок, 2 таблиці.

OPENCV, КЛАСИФІКАЦІЯ ЗОБРАЖЕНЬ, МАШИННЕ НАВЧАННЯ, НЕЙРОННА МЕРЕЖА, МОДЕЛЬ, РОЗПІЗНАВАННЯ ЗОБРАЖЕНЬ.

Об’єкт дослідження – системи комп’ютерного зору, які здійснюють класифікацію та ідентифікацію об’єктів на зображенні.

Мета дослідження – аналіз існуючих моделей класифікації зображень, переваг та перспектив їх поширення за допомогою дослідження ефективності їх застосування.

В результаті роботи проведеного дослідження була виконана оцінка ефективності застосування існуючих згорткових нейронних мереж на базі бібліотеки OpenCV та сформовано рекомендації щодо застосування існуючих фреймворків глибинного навчання на базі цієї бібліотеки.

OPENCV, IMAGE CLASSIFICATION, IMAGE RECOGNITION, MACHINE TRAINING, MODEL, NEURAL NETWORK.

Object of research is a computer vision systems that classify and identify objects in an image.

The purpose of the research is to analyse existing models of image classification, benefits and prospects of their dissemination by examining the effectiveness of their application.

Result of the work is an efficiency evaluation of the use of existing convolutional neural networks based on the OpenCV library with the conclusions about the use of existing deep learning frameworks based on this library.

ЗМІСТ

ВСТУП

Розпізнавання візуальних образів є одним з найважливіших компонентів систем управління та обробки інформації, автоматизованих систем і систем прийняття рішень. Завдання, пов'язані з класифікацією і ідентифікацією предметів, явищ і сигналів, що характеризуються кінцевим набором деяких властивостей і ознак, виникають в таких галузях як робототехніка, інформаційний пошук, моніторинг та аналіз візуальних даних, дослідження штучного інтелекту. Алгоритмічна обробка і класифікація зображень застосовуються в системах безпеки, контролю і управління доступом, в системах відеоспостереження, системах віртуальної реальності та інформаційних пошукових системах. На даний момент в виробництві широко використовуються системи розпізнавання рукописного тексту, автомобільних номерів, відбитків пальців або людських осіб, що знаходять застосування в інтерфейсах програмних продуктів, системах безпеки та ідентифікації особистості, а також в інших прикладних цілях.

В сучасних умовах прискореного темпу розвитку технологій та їх автоматизації, поширенням робототехніки, розвитком інтернет-речей та систем штучного інтелекту одними із найбільш актуальних галузей залишаються сфери машинного навчання, комп’ютерного зору та розпізнавання об’єктів на зображеннях, а також їх застосуванням в рамках стартапів або невеличких проектів, в рамках яких особливо гостро стоїть питання можливості їх застосування, швидкості навчання та точністю результатів.

Актуальність даної проблеми особливо висока в галузях, де розпізнавання образів застосовується в природному середовищі (відеоспостереження, аналіз даних камер моніторингу, робото-технічні зорові системи), де зоровий сенсор може мати довільний обмежений кут огляду по відношенню до шуканого об'єкту.

Метою роботи є аналіз існуючих моделей класифікації зображень, переваг та перспектив їх поширення за допомогою дослідження ефективності їх застосування. Галузь застосування моделей класифікації зображень поширюється на системи

управління та обробки інформації, автоматизовані системи і системи прийняття рішень.

Об’єктом дослідження атестаційної роботи є системи комп’ютерного зору, які здійснюють класифікацію та ідентифікацію об’єктів на зображенні.

Предметом дослідження атестаційної роботи є моделі класифікації зображень.

В ході атестаційної роботи магістра було:

– проведено аналіз методів розпізнавання зображень;

– проведено аналіз існуючих згорткових нейронних мереж для класифікації зображень;

– проведено експериментальне дослідження ефективності існуючих згорткових нейронних мереж на базі бібліотеки OpenCV;

– проведено експериментальне дослідження ефективності фреймворків глибинного навчання на базі бібліотеки OpenCV;

– за результатами проведеного дослідження була виконана оцінка ефективності застосування існуючих згорткових нейронних мереж на базі бібліотеки OpenCV та сформовано рекомендації щодо застосування існуючих фреймворків глибинного навчання на базі цієї бібліотеки.

За результатами атестаційної роботи магістра було розроблено презентацію (див. додаток А).

1 АНАЛІТИЧНИЙ ОГЛЯД СУЧАСНИХ МЕТОДІВ РОЗПІЗАВАННЯ ЗОБРАЖЕНЬ

1.1 Моделі-класифікатори

Один з основних підходів, що найбільш широко використовувався в області розпізнавання зображень являє собою застосування класичних моделей-класифікаторів, що навчаються з учителем. Для навчання таких моделей використовуються маркована вибірка даних, що складається з масиву зображень і відповідного їм масиву міток, що визначають категорію, до якої відноситься зображення. В процесі навчання масив даних розділяється на дві нерівні частини – навчальну вибірку і тестову вибірку, потім за допомогою специфічного для конкретного алгоритму правила навчання параметри моделі налаштовуються з використанням навчальної вибірки таким чином, щоб отримавши в якості вхідних даних зображення, модель на виході виробляла б мітку відповідного класу. Цей підхід представлений безліччю моделей, серед яких найбільш широко використовуваними є регресійна модель, штучна нейронна мережа (багатошаровий перцептрон), метод опорних векторів, а також дерева прийняття рішень і моделі-ансамблі, що представляють собою поєднання деяких перерахованих моделей [1].

Багатошарові перцептрони, які навчаються методом зворотного поширення помилки, широко використовуються для розпізнавання різних категорій зображень, таких як рукописні цифри, почерк, людські обличчя і дані зорових сенсорів робото-технічних систем [2]. Модель багатошарового перцептрона являє собою сукупність штучних нейронів – обчислювальної одиниці моделі – об'єднаних в рівні (шари), задані в ієрархічному порядку.

Штучний нейрон являє собою модель біологічного нейрона (нервової клітини), представлену одним або декількома входами, одним виходом і функцією активації [3]. Крім цього, кожен вхід штучного нейрона має асоційований коефіцієнт або вага, а вихідне значення нейрона є значення функції активації від зваженої суми його вхідних значень.

Як функції активації може виступати функція, що володіє властивостями нелінійності, нормалізації вхідних даних, і деякими іншими.

При об'єднанні штучних нейронів в мережу вхідні значення нейрона шару представляють собою вихідні значення нейронів попереднього шару. При цьому нейрони першого (вхідного) шару отримують в якості вхідних значення безпосередньо дані, що підлягають розпізнаванню, які в разі розпізнавання зображення представляють собою значення інтенсивності складових його пікселів (точкових елементів). Вихідний шар мережі може варіюватися в залежності від завдання, але класична архітектура має на увазі формування його числом нейронів, рівній кількості класів розпізнавання, при цьому вихідне значення кожного нейрона нормується по інтервалу {0,1}, і являє собою ймовірність приналежності вхідного зображення до відповідного класу [4].

Оскільки сформулювати аналітично правило класифікації зображень за категоріями розпізнавання часто є скрутним, здатність навчатися на базі вибірки робить нейронні мережі і споріднені з ними моделі придатними для розпізнавання природних зображень навколишнього світу, що відрізняються нечіткою структурою і безліччю варіацій в межах класу.

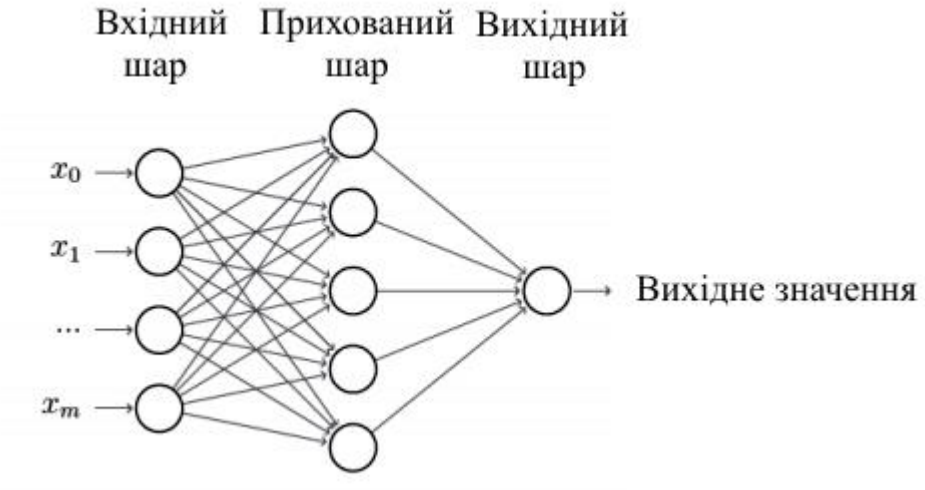

Рис. 1.1 – Схема штучної нейронної мережі з трьома шарами

Навчання мережі методом зворотного поширення помилки включає в себе три етапи: подачу на вхід даних, з подальшим поширенням даних в напрямку

виходів, обчислення і зворотне поширення відповідної помилки і коригування ваг. Після навчання передбачається лише подача на вхід мережі даних та поширення їх в напрямку виходів. При цьому, якщо навчання мережі може бути досить тривалим процесом, то безпосереднє обчислення результатів навченої мережею відбувається дуже швидко. Крім того, існують численні варіації методу зворотного поширення помилки, розроблені з метою збільшення швидкості протікання процесу навчання.

Також варто відзначити, що одношарова нейронна мережа істотно обмежена в тому, навчання яким шаблонами вхідних даних вона підлягає, в той час, як багатошарова мережа (з одним або більше прихованим шаром) не має такого недоліку.

Багатошарові перцептрони демонструють успішні результати при використанні їх для розпізнавання зображень деяких окремих обмежених категорій, таких як символи природної мови, рукописні цифри і почерк [4]. В даний час в більшості додатків, використовують пряме навчання з учителем для розпізнавання зображень, нейронні мережі витіснені методом опорних векторів, що пропонують більш ефективне з точки зору обсягу обчислювальних ресурсів рішення.

Метод опорних векторів розглядає кожен екземпляр даних (зображення) як точку в n-вимірному просторі, де n відповідає розмірності даних або загальної кількості пікселів зображення. Кожна з точок належить до певного класу (категорії). При цьому завдання розпізнавання представляється у вигляді завдання по знаходженню такої гіперплощини в n-вимірному просторі, яка відділяла б усі крапки, відповідні зображенням даного класу від інших, які не належать йому. Припускаючи, що таких гіперплощин може існувати безліч, метод опорних векторів ставить метою відшукання оптимальної площини, відстань до якої від найближчої точки мінімізується в межах безлічі можливих варіантів.

Навчання моделі, що використовує метод опорних векторів виробляється методами квадратичного програмування, такими як послідовна мінімальна оптимізація [5].

Метод опорних векторів має деякі переваги і недоліки по відношенню до використання багатошарових перцептронів:

– багатошаровий перцептрон є моделлю з безліччю прихованих параметрів, що залежать від числа нейронів мережі. Параметризована модель потенційно здатна до інкапсуляції більш складних, високорівневих функцій, але при цьому вимагає більше часу і обчислювальних ресурсів для навчання та налаштування параметрів. Метод опорних векторів використовує вектори, відібрані з навчальної вибірки, при цьому кількість параметрів обмежена зверху розміром вибірки, а на практиці може бути проріджені за рахунок використання інженерії ознак [3];

– на відміну від навчання нейронної мережі, яке здійснюється з допомогою методу градієнтного спуску (і його варіацій) і оцінки помилки мережі, навчання моделі опорних векторів включає в себе не тільки оцінку помилки, але і метрику складності отриманої гіперплощини. Пошук оптимального значення нейронної мережі вразливий до наявності локального мінімуму, здатного зупинити процес градієнтного спуску, при цьому метод опорних векторів при коректному виборі мета-параметрів гарантує знаходження глобального рішення [6];

– навчена нейронна мережа вимагає мінімальних обчислювальних ресурсів для роботи в режимі розпізнавання (передбачення категорій). Метод опорних векторів в деяких випадках, коли число векторів велике в порівнянні з розміром вибірки, будує передбачення істотно повільніше [3];

– у порівнянні з нелінійним (використовують ядра) методом опорних векторів, нейронна мережа демонструє розширені можливості до онлайн-навчання, коли розмір вибірки не фіксований і поповнюється за рахунок надходження нових даних.

1.2 Метод головних компонент

Одна з особливостей розпізнавання зображень в порівнянні з іншими додатками теорії розпізнавання образів полягає в тому, що зображення в растровому вигляді (у вигляді двовимірної матриці пікселів, кожен з яких має деякий значення яскравості або кольору), мають високу розмірність – середньостатистична фотографія може бути представлена вектором довжини ~ 10^6.Дані, представлені розмірністю таких порядків, вимагають виняткових обчислювальних ресурсів, і практично не піддаються обробці на сучасних персональних комп'ютерах (ситуація, відома як «прокляття розмірності» [1]). При цьому, однак, лише невелика частина цих параметрів критична для завдання розпізнавання, що дозволяє зображень демонструвати низьку чутливість до випадкового шуму і глобальним спотворень. ця особливість успішно використовується в алгоритмах стиснення з втратами – за допомогою алгоритму JPEG зображення може бути стисло аж до до 10%, при цьому зміни залишаються непомітні для людського ока. З огляду на цю особливість, стає можливим застосування до природних зображень статистичних методів зниження розмірності, таких як метод головних компонент [6]. Суть методу полягає в тому, щоб представити вхідні дані у вигляді лінійної суми компонент з деякими коефіцієнтами. Класичний метод головних компонент, однак, непридатний для більшості зображень через обчислювальної складності побудови ковариационной матриці. Турк і Пентланд [7] в 1991-ому р. запропонували алгоритм розпізнавання Eigenfaces, де використовували альтернативний, прийнятний для сучасних комп'ютер метод розрахунку власних векторів. В їхньому прикладі метод використовувався на фронтальних фотографіях людських обличь. Підтверджуючи припущення про те, що розмірність зображення може бути значно знижена, зберігаючи при цьому досить інформації для успішного розпізнавання людиною, вони показали, що кожна з осіб вибірки можна уявити при допомоги обмеженого (<10) набору головних компонент.

Рис. 1.2 – Приклади головних компонент алгоритму Eigenfaces [91]

Для розпізнавання тестові зображення проектувалися на базис обраних головних компонент, тобто представлялися у вигляді лінійної суми доданків. Потім на представлених таким чином даних тренували модель, використовує навчання з учителем (багатошаровий перцептрон або SVM), і таким чином, завдання зводилася до класичної. Використання Eigenfaces дозволяло ефективно розпізнавати обличчя при різному освітленні і давало деяку стійкості до орієнтації; проте, алгоритм погано працював на обличчях різного розміру (варіації масштабу). Крім того, алгоритм був розрахований на те, що вхідні дані будуть являти собою особи, зорієнтовані відповідним чином, не пропонуючи методу відшукання цікавить фрагмента особи серед зображення композитної сцени.

Крім перерахованих, метод головних компонент мав і інші обмеження, які сприяли появі нових методів подання зображень. Б. Ольшозен в своїй роботі [8] показав, що алгоритм, названим їм розрідженим кодуванням здатний ефективніше представляти внутрішню структуру зображення і об'єктів в ньому, при цьому демонструючи деякі властивості, вражаюче схожі з властивостями клітин зорової кори головного мозку (так званих «простих клітин» зони V1). Цей алгоритм, проте, в протилежність PCA, представляв дані у вигляді надповного базису векторів, кожен з яких, таким чином, не був лінійно незалежним від інших.

Розрідженість отриманого уявлення забезпечується тим, що для окремо взятого зображення, більшість компонентів будуть мати коефіцієнт, що дорівнює нулю. Ця умова мотивовано тим фактом, що природні зображення, як правило, можуть бути представлені за допомогою комбінації невеликого числа ненульових компонент-примітивів, таких як краю або кордону (в області алгоритмів

розрідженого кодування відповідні компоненти звуться «атомів» або «кодових слів»). Таким чином, розріджений кодування забезпечує великий набір компонентів, які можуть значно відрізнятися один від одного, при цьому гарантуючи, що окремо взяте зображення буде представлено за допомогою суми всього лише деяких з них. Існують різні алгоритми пошуку розрідженого коду для вибірки зображень, таких як ортогональне узгоджене переслідування, регресія найменшого кута [6] і використання специфічних нейронних мереж – розріджених автоенкодерів. Перевага цього методу в порівнянні з методом головних компонент виражається в тому, що компоненти, отримані за допомогою другого способу, завжди представляють собою лінійні перетворення вхідних даних, тоді як у разі розрідженого коду (і деяких інших уявлення) компоненти можуть бути нелінійними, приховуючи, таким чином, більш складні функції представлення даних. Інший широко використовується клас алгоритмів, здатний формувати цілісні уявлення об'єктів – обмежені машини Больцмана [2]. Обмежена машина Больцмана являє собою породжує стохастичну нейронну мережу, яка навчається формування деякого імовірному розподілу своїх входів. Вони являють собою модифікацію класичних машин Больцмана, які, в свою чергу, є варіаціями мереж Хопфілда.

Зв'язки між нейронами такої мережі являють собою двочастковий граф, де одна частина відповідає вхідному прошарку мережі, а друга – прихованому шару. Кожен вхідний нейрон з'єднаний з усіма прихованими нейронами при допомоги симетричних зв'язків, при цьому нейрони в межах кожної частини біграфа не мають зв'язків один з одним (на відміну від класичних, «необмежених» машин Больцмана, де такі зв'язки можливі). Це обмеження дозволяє ефективно навчати мережу, використовуючи алгоритм зіставлення розбіжність [9]. Істотною перевагою уявлень, які формуються за допомогою розрідженого кодування і обмеженою машини Больцмана в порівнянні з методом головних компонент є їх нелінійність, що дозволяє розглядати методи нарощування таких репрезентативних моделей. цей підхід відомий під назвою глибокого навчання, і відзначений різким підвищенням точності розпізнавання в безлічі сфер машинного навчання, в тому числі в

розпізнаванні зображень. В його основі лежить припущення про те, що уявлення, яким навчаються репрезентативні моделі, мають ієрархічну природу, і таким чином, існує можливість навчання каскаду моделей, кожен з яких приймає в якості вхідних даних уявлення, що виробляються вищестоящою моделлю. Метод головних компонент, таким чином, не здатний формувати глибокі ієрархії уявлень, оскільки будь-яка, необмежено велика комбінація лінійних перетворень тотожна одному лінійному перетворенню [3].

Для розпізнавання зображень успішно застосовувалися глибокі моделі, що складаються з обмежених машин Больцмана – так звані глибокі мережі довіри [9]. Використання ієрархічних уявлень дозволяє такий моделі навчатися складним, масштабних об'єктах, забезпечуючи додаткові рівні стійкості до інваріантним перетворенням на кожному шарі уявлення. Так, глибока модель, навчена на базі людських облич, здатна розпізнавати значно більш суттєві викривлення, ніж модель Eigenfaces, що включають в себе обертання об'єкта в межах обмежених кутів.

Глибокі моделі також можуть будуватися і на базі методів розрідженого кодування – одним з найбільш відомих є глибокий автоенкодер [9], ті, яких навчають пошарово, жадібним чином. В цілому глибокі моделі забезпечують більш гнучкі і багаті уявлення, які підходять для об'єктів зі складною структурою. Зворотною стороною цієї переваги є ускладнений процес навчання, в окремих випадках (для глибоких мереж довіри) вимагає розробки окремих алгоритмів, і в загальному випадку – споживає більше обчислювальних ресурсів.

Компактні цілісні уявлення дозволяють позбутися від «прокляття розмірності », перетворюючи складні в обробці, об'ємні зображення в компактний вид, забезпечуючи при цьому деяку стійкість до варіативності.

Методи, які здійснюють нелінійні перетворення, такі як розріджений кодування, можуть використовуватися для отримання багаторівневих уявлень, використовуючи глибоке навчання і властивість стаціонарності природних зображень (той факт, що статистичні характеристики локальних ділянок зображень, як правило, розподілені рівномірно). При цьому підході відшукання

компактних цілісних уявлень демонструє високі результати для об'єктів, що мають в цілому схожу форму (як людські обличчя), але не здатний справлятися з об'єктами, що мають значні візуальні відмінності (наприклад, відносити до одного класу автомобілі різних моделей) [3].

Більш того, оскільки розпізнаються об'єкти зазвичай мають тривимірну природу, вони здатні істотно змінювати форму під впливом геометричних трансформацій (так, зображення особи в профіль не може бути представлено сумою компонентів, отриманих декомпозицією зображення особи анфас). В силу умови цілісності отримані уявлення уразливі до проблеми неповних даних – ситуацій, коли частина об'єкта загороджена або нерозрізнена через шум. Для отримання компактних цілісних уявлень, таким чином, необхідна суворо підібрана вибірка об'єктів, вирівняних по загальній орієнтації. Складання подібної вибірки має на увазі участь експериментатора і обробки вихідних зображень людиною.

Ці особливості і обмеження методу зниження розмірності привели до розвитку альтернативного підходу до розпізнавання, специфічного для сфери розпізнавання зображень і використовує виявлення локальних ознак, що є стійкими компоненти (частини) зображеного об'єкта.

1.3 Згорткові нейронні мережі

Проблеми, що виникли в процесі використання моделей, які формують цілісні репрезентації, сприяли розвитку нової групи алгоритмів, що використовують локальні ознаки зображень. Необхідність такого підходу була продиктована властивістю стаціонарності природних зображень – об'єкти, присутні на зображенні, могли вільно переміщатися в межах поля зору, при цьому бажаним результатом розпізнає алгоритму залишалося співвіднесення безлічі таких інваріантних репрезентацій об'єкта до одного класу.

Крім іншого, використання локальних ознак при розпізнаванні зображень було підкріплено свідченнями з області нейробіології. В класичної роботі Д. Хьюбела і Т. Візела [10], що візуальна кора головного мозку являє собою складний комплекс клітин, кожна з яких чутлива тільки до обмеженого ділянці поля зору. Такі ділянки, інакше звані рецептивних полями, стикуються разом, забезпечуючи перекриття всього поля зору. Відповідні клітини при цьому виконують роль локальних фільтрів вхідних даних, реагуючи на присутність у власному рецептивної поле деяких примітивних структур, таких як краю і кордони. Було виявлено також існування так званих «складних клітин», мають ширші рецептивні поля, і демонстрували інваріантність по відношенню до точного розташування об'єкта в полі зору.

З урахуванням того, що візуальна кора головного мозку являє собою найбільш потужну і гнучку зорову систему з існуючих на даний момент, поява моделей [4], симулює її поведінку, виглядало природним кроком. Однією з найбільш успішних моделей, що вважається визнаним лідером [11] в області розпізнавання зображень є згорткова нейронна мережа.

Згорткові мережі являють собою варіацію архітектури багатошарового перцептрону, і включають в себе згорткові шари, шари підвибірки і повнозв'язні шари. Архітектура сверточное мережі використовує переваги двовимірної структури вхідних даних – зображень з допомогою методу локальної зв'язності, обмежуючи кількість зв'язків між нейронами прихованого згорткового шару і вхідними даними. Конкретно, кожен нейрон прихованого шару пов'язаний тільки з обмеженим локальним (які не мають розривів) ділянкою зображення.

Крім цього, нейронна мережа використовує загальні, або розділяються ваги, накладаючи штучне обмеження на алгоритм навчання зворотним поширенням помилки, так, щоб кожен нейрон прихованого шару мав набір ваг, спільний з іншими нейронами цього шару. При прямому поширенні така мережу здійснює математичну операцію згортки вхідного зображення набором фільтрів, які подаються вагами нейронів прихованого шару [8].

Проміжними результатами мережі є так звані «карти ознак» – двовимірні матриці, що представляють собою результат згортки окремим фільтром.

Шар підвибірки виконує операцію угруповання карт ознак, розглядаючи регіони окремих розмірів та агрегує значення, отримані в результаті згортки. Основне призначення шарів підвибірки – знизити варіативність даних, забезпечуючи стійкість до трансляцій локального ознаки в межах окремого регіону.

Таким чином, за рахунок використання декількох попереміннних шарів згортки і шарів підвибірки згорткова мережа дозволяє отримувати уявлення, незалежні від конкретного розташування локального ознаки в зображенні, і однаковим чином реагувати на об'єкти, що цікавлять (наприклад, людські обличчя), присутні на будь-якій ділянці фотографії.

Згорткові мережі – один з найбільш успішних існуючих на сьогоднішній день алгоритмів розпізнавання зображень. моделям, реалізують відповідну архітектуру, належать перші місця в змаганнях алгоритмів розпізнавання, таких як ImageNet. Серед недоліків виділяють труднощі при обробці маленьких об'єктів, і нездатність справлятися з спотвореннями, такими як розмиває фільтр або сильний шум (такі спотворення присутні в навколишньому світі, наприклад, при погляді через товсте скло). При цьому згорткові мережі порівняно легко справляються з проблемами високоточного розпізнавання, які викликають труднощі у людей – наприклад, розпізнавання окремих моделей машин або порід собак, і інші завдання, що вимагають виділення вузько-специфічних ознак.

Однією з основних особливостей згорткових мереж є той факт, що така модель не володіє інформацією про те, як саме локалізований зображений шуканий об'єкт – його конкретне місцезнаходження та орієнтація в просторі. При цьому в рішенні прикладних задач управління і обробки інформації, знання параметрів локалізації є необхідною умовою – в залежності від розташування або пози об'єкта система обробки інформації може класифікувати зображення по-різному відповідно до покладених на неї завдань.

Альтернативні підходи до виділення локальних ознак включають в себе методи класичного комп'ютерного зору, які не використовують навчальні моделі.

Ці методи здійснюють пошук на зображенні характерних ділянок, відповідають алгоритмічно явно заданим умовам. Серед них виділяється низка умов.

Виявлення країв/кордонів. Краєм називається ділянку зображення, представляє собою кордон між двома контрастними регіонами, помітну людським оком. Математично точки, складові такі ділянки, визначаються як точки, де градієнт зображення має локальний максимум.

Дослідження як в області функціонування біологічних зорових систем, так і теорії інформації [12], показують, що репрезентація об'єктів в полі зору за допомогою кордонів є ефективний з точки зору мінімізації ентропії спосіб зберігання і обробки інформації і може використовуватися для компактної репрезентації зображення. Окрім цього, виділення меж дозволяє знизити вплив деяких факторів, які не впливають на розпізнавання, таких як освітлення і тіні. Виділення меж (за допомогою фільтрів Кенні, Собеля, або згортки вейвлетами Габора) часто використовується як попередній етап обробки зображень в інших алгоритмах розпізнавання, в тому числі – згорткових мережах [9].

Виявлення кутів або «точок інтересу». До цієї групи належать алгоритми, які виділяють локальні ділянки зображення, максимально чутливі до змін. Традиційно ця група алгоритмів (що включає в себе детектор Харріса, детектор Ши-Томасі і інші) використовувалася для відшукання кутів між прямими лініями, але в даний момент за її допомогою розглядаються також будь-які точки з високим значенням кривизни [13].

Виявлення ділянок неоднорідності. Під ділянками неоднорідності, на відміну від кутів, розуміються деякі безперервні регіони зображення, які відрізняються за значеннями кольору або інтенсивності від навколишнього фону, і при цьому схожі між собою. Як правило, такі ділянки відповідають локальним екстремумам зображення [8].

Перераховані локальні ознаки широко використовуються в задачах візуального трекінгу і стеження за об'єктом, але в чистому вигляді непридатні для завдання розпізнавання в силу своєї недискримінатівной природи – такі методи не надає можливості відрізнити один кут (або ділянку неоднорідності) від іншого і

висловити відмінність або схожість в числовому еквіваленті. Цим вимогам, проте, задовольняють підходи, які використовують ідею відшукання точок інтересу з використанням локальних дескрипторів, і представлені такими алгоритмами як SIFT, SURF і ORB [14].

Дескриптор є композицією ділянок зображення, локалізованих спільно, де для кожної ділянки або блоку розраховуються параметри орієнтації, масштабу, і деякі інші, що дозволяють імовірнісно ідентифікувати місце розташування ознаки, яке відповідає даному дескриптору. З урахуванням використання в дескрипторах параметрів орієнтації та масштабу, такі ознаки виявляються інваріантними по відношенню до обертання зображення, зміни масштабу і яскравості або контрасту. Такі алгоритми як SIFT і ORB, крім того, забезпечують можливість зіставлення зображень, співвідносячи однакові локальні ознаки один з одним.

Рис. 1.3 – Пошук локальних ознак SIFT

Ознаки, що використовують локальні дескриптори, можуть ефективно використовуватися для розпізнавання зображень одного і того ж об'єкта під впливом афінних перетворень в тривимірному просторі. Локальність ознак дає можливість справлятися з проблемою оклюзії, забезпечуючи можливість зіставляти об'єкти по частинах. Основні проблеми методів сімейства SIFT – слабка стійкість до варіативності, що не дозволяє алгоритму відносити до одного класу об'єкти, візуально відрізняються формою або текстурою [14].

Методи виділення локальних ознак дозволяють справлятися з деякими класами проблем розпізнавання зображень, забезпечуючи стійкість до оклюзії, знижуючи обчислювальну навантаження при обробці зображень високої розмірності і дозволяючи формувати інваріантні ознаки для виявлення об'єктів під дією інваріантних перетворень.

1.4 Методи оцінки ефективності розпізнавання

Актуальним питанням при розпізнаванні зображень є оцінка ефективності роботи методу розпізнавання. Для отримання чисельного значення оцінки широко використовуються як загальні методи математичної статистики, так і специфічні показники, що застосовуються для оцінки алгоритмів машинного навчання.

Однією з найбільш простих метрик оцінки ефективності є процентна частка коректно розпізнаних зображень. Коректним розпізнаванням вважається отримання на виході алгоритму класу, відповідного попередньо заданому класу. Для оцінки використовується вибірка, що спроектована аналогічно навчальній вибірці, але містить зображення, до яких алгоритм не міг мати доступ в процесі навчання. Для цього, як правило, вихідна загальна вибірка поділяється на дві нерівні частини (розмір тестової вибірки при цьому може відрізнятися, і складати 20-30% розміру загальної вибірки [5]).

Розглянутий показник є найбільш узагальненим і підходить для безлічі завдань розпізнавання з обмеженою кількістю класів. У завданнях, де кількість класів не фіксоване, і завдання розпізнавання являє собою завдання ідентифікації об'єкта певної категорії серед безлічі інших, потенційно необмежених категорій, замість неї застосовуються такі показники як точність і повнота оцінки. Їх використання дозволяє розрізняти помилково-позитивні (класифікатор прийняв позитивне рішення по зображенню, що не містить шуканого об'єкта) і помилково-негативні (класифікатор не впізнав об'єкт на зображенні, де він був присутній)

помилки розпізнавання, або помилки першого і другого роду. Таким чином, точність оцінки в межах класу представляє собою частку зображень дійсно належать даному класу щодо всіх зображень які система віднесла до цього класу. Повнота системи – це частка знайдених класифікатором зображень, що належать класу щодо всіх зображень цього класу в тестовій вибірці [2].

Показники точності і повноти широко використовуються в області обробки інформації, і як правило, розраховуються спільно. При цьому існує кілька методів зіставлення двох показників:

– кожен показник враховується індивідуально;

– для випадків, коли між показниками може спостерігатися спостерігається залежність, проводиться оцінка одного з показників при фіксації іншого;

– обидва показники можуть бути скомбіновані в один.

З точки зору теорії ймовірності ці показники можуть інтерпретуватися наступним чином: точність відповідає ймовірності того, що випадково обраний з безлічі позитивно упізнаних зображень дійсно розпізнаний коректно, при цьому повнота є ймовірністю того, що випадково обраний з загальної зображення буде коректно класифікований алгоритмом. Так, в залежності від програми завдання, до продуктивності методу розпізнавання можуть бути пред'явлені вимоги, що стосуються як максимізації повноти (для випадків, коли певна кількість помилково-позитивних рішень допускається), так і збалансованого значення двох показників.

2 АНАЛІЗ ІСНУЮЧИХ НЕЙРОННИХ МЕРЕЖ

2.1 Загальний аналіз архітектур нейронних мереж

Алгоритми глибоких нейромереж сьогодні здобули велику популярність, яка багато в чому забезпечується продуманістю їх архітектур. З моменту свого зародження технології штучних нейронних мереж розвивалися досить відокремлено від класичних методів, нерідко докорінно змінюючи уявлення про предмет і проблематику теорії машинного навчання і розпізнавання об’єктів, залишаючи значний вплив на теоретичний, термінологічний і методологічний апарати цих дисциплін. Через деякий час після розвитку базових моделей штучних нейронних мереж, відбувся значний поділ науки про нейромережі на види топологій архітектури мереж і методи навчання мереж.

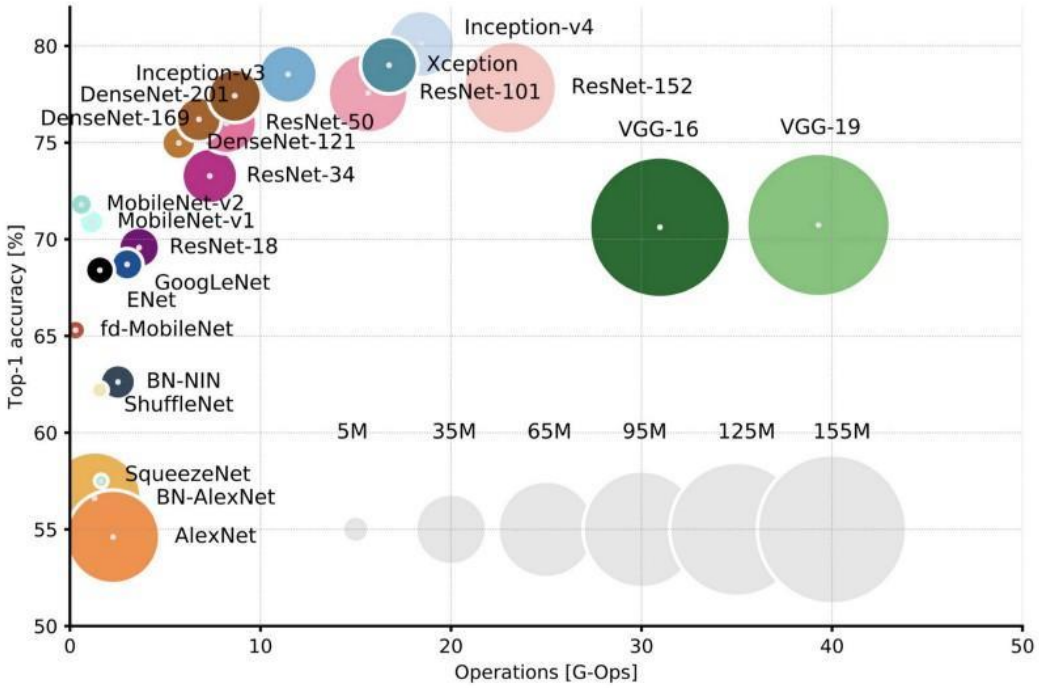

Рис. 2.1 – Порівняння популярних архітектур по one-crop-точності

На рис. 2.1 представлено порівняння популярних архітектур нейронних мереж по one-crop-точності і кількості операцій, необхідних для одного прямого проходу. Розмір крапок пропорційний кількості мережевих параметрів; в

нижньому правому куті представлена легенда, яка охоплює межі параметрів в рамках від 5×106 до 155×106. І ті, й інші фігури мають таку ж вісь ординат. Сірі точки виділяють центр крапок [15].

2.2 LeNet5

У 1994-му була розроблена одна з перших згорткових нейронних мереж, що поклала початок глибокому навчанню. Ця піонерська робота Яна Лекуна (Yann LeCun) після багатьох успішних ітерацій починаючи з 1988-го отримала назву LeNet5 [16].

Архітектура LeNet5 стала фундаментальною для глибокого навчання, особливо з точки зору розподілу властивостей зображення по всій картинці. Згортки з учнями параметрами дозволяли за допомогою декількох параметрів ефективно витягати однакові властивості з різних місць. У ті роки ще не було відеокарт, здатних прискорити процес навчання, і навіть центральні процесори були повільними. Тому ключовим перевагою архітектури виявилася можливість зберігати параметри і результати обчислень, на відміну від використання кожного пікселя в якості окремих вхідних даних для великої багатошарової нейромережі.

У LeNet5 в першому шарі пікселі не використовуються, тому що зображення сильно корельовані просторово, так що використання окремих пікселів в якості вхідних властивостей не дозволить скористатися перевагами цих кореляцій.

Особливості LeNet5:

– використання послідовності з трьох шарів – згортки (convolution), групування (pooling) і нелінійності (non-linearity);

– використання згортки для вилучення просторових властивостей;

– підвибірки з використанням просторового усереднення карт;

– нелінійність у вигляді гіперболічного тангенса;

– фінальний класифікатор у вигляді багатошарової нейромережі;

– розріджена матриця зв'язності між шарами, яка дозволяє зменшити обсяг обчислень.

Ця нейронна мережа лягла в основу багатьох наступних архітектур і надихнула безліч дослідників.

З 1998-го по 2010-й нейронні мережі перебували в стані інкубації. Більшість людей не помічали їх зростаючих можливостей, хоча багато розробників поступово відточували алгоритми. Завдяки розквіту камер мобільних телефонів і здешевлення цифрових фотоапаратів ставало доступно все більше даних для навчання. Заодно росли і обчислювальні можливості, процесори ставали могутніше, а відеокарти перетворилися в основний обчислювальний інструмент. Всі ці процеси дозволяли розвиватися і нейронним мережам, нехай і досить повільно.

2.3 AlexNet

У 2012-му Олексій Крижевський опублікував AlexNet, поглиблену і розширену версію LeNet, яка з великим відривом перемогла в складному змаганні ImageNet.

ImageNet – це набір з 15 мільйонів помічених зображень з високою роздільною здатністю, розділених на 22 000 категорій. Зображення зібрані в інтернеті та позначені вручну за допомогою краудсорсинга Amazon's Mechanical Turk. Починаючи з 2010 року проводиться щорічний конкурс ImageNet Large-Scale Visual Recognition Challenge (ILSVRC), який є частиною Pascal Visual Object Challenge. У змаганні використовується частина датасета ImageNet від 1000 зображень в кожній з 1000 категорій. Всього виходить 1,2 мільйона зображень для навчання, 50 000 зображень для перевірки і 150 000 – для тестування. ImageNet складається із зображень з різною роздільною здатністю. Тому для конкурсу їх масштабують до фіксованого розміру 256×256. Якщо спочатку зображення було прямокутним, то його обрізають до квадрата в центрі зображення [17].

AlexNet – цезгорткова нейронна мережа, яка справила великий вплив на розвиток машинного навчання та особливо – на алгоритми комп'ютерного зору. Мережа з великим відривом виграла конкурс по розпізнаванню зображень ImageNet LSVRC-2012 в 2012 році (з кількістю помилок 15,3% проти 26,2% в другого місця).

Архітектура AlexNet схожа з створеної мережею LeNet, однак в AlexNet більше фільтрів на шарі і вкладених згортальних шарів. Мережа включає в себе згортки, максимальне об'єднання, дропаути, поширення даних, функції активацій і стохастичний градієнтний спуск.

Особливості AlexNet:

– використання функцій активації лінійної ректифікації ReLU замість арктангенса для додавання нелінійності в модель, за рахунок чого швидкість методу виростає в 6 разів;

– використання дропаутів замість регуляризації як вирішення проблеми перенавчання;

– перекриття об'єднань для зменшення розміру мережі, за рахунок чого рівень помилок першого і п'ятого рівнів знижуються до 0,4% і 0,3%, відповідно.

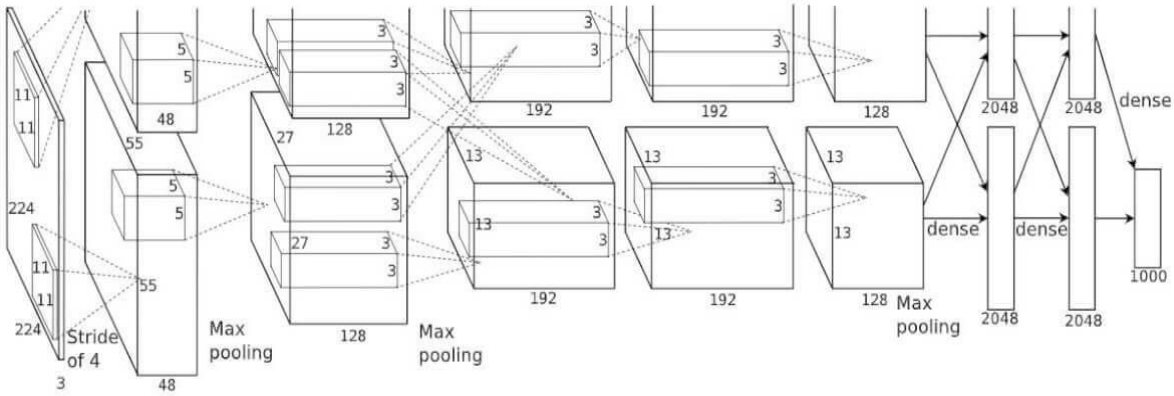

Рис. 2.2 – Архітектура мережі AlexNet

Архітектура мережі приведена на рис. 2.2. AlexNet містить вісім шарів з ваговими коефіцієнтами. Перші п'ять з них згорткові, а решта три – повнозв'язні. Вихідні дані пропускаються через функцію втрат softmax, яка формує розподіл

однієї тисячі міток класів. Мережа максимізує багатолінійну логістичну регресію, що еквівалентно максимізації середнього по всім навчальним випадків логарифма ймовірності правильного маркування з розподілу очікування. Ядра другого, четвертого і п'ятого згортальних шарів пов'язані тільки з тими картами ядра в попередньому шарі, які знаходяться на одному і тому ж графічному процесорі. Ядра третього сверточного шару пов'язані з усіма картами ядер другого шару. Нейрони в повнозв'язних шарах пов'язані з усіма нейронами попереднього шару [18].

Таким чином, AlexNet містить 5 згорткових шарів і 3 повнозв'язних шари. ReLU застосовується після кожного згорткового і повнозв'язного шару. Дропаут застосовується перед першим і другим повнозв'язними шарами. Мережа містить 62,3 мільйона параметрів і витрачає 1,1 мільярда обчислень при прямому проході. Згорткові шари, на які припадає 6% всіх параметрів, виробляють 95% обчислень.

Особливості цього рішення:

– використання блоків лінійної ректифікації (ReLU) в якості нелінійностей;

– використання методики відкидання для вибіркового ігнорування окремих нейронів в ході навчання, що дозволяє уникнути перенавчання моделі;

– використання функцій активації лінійної ректифікації ReLU замість арктангенса для додавання нелінійності в модель, за рахунок чого швидкість методу виростає в 6 разів;

– перекриття max pooling, що дозволяє уникнути ефектів усереднення average pooling;

– використання NVIDIA GTX 580 для прискорення навчання.

Таким чином, мережа досягла наступного рівня помилок першого і п'ятого рівнів: 37,5% і 17,0% відповідно.

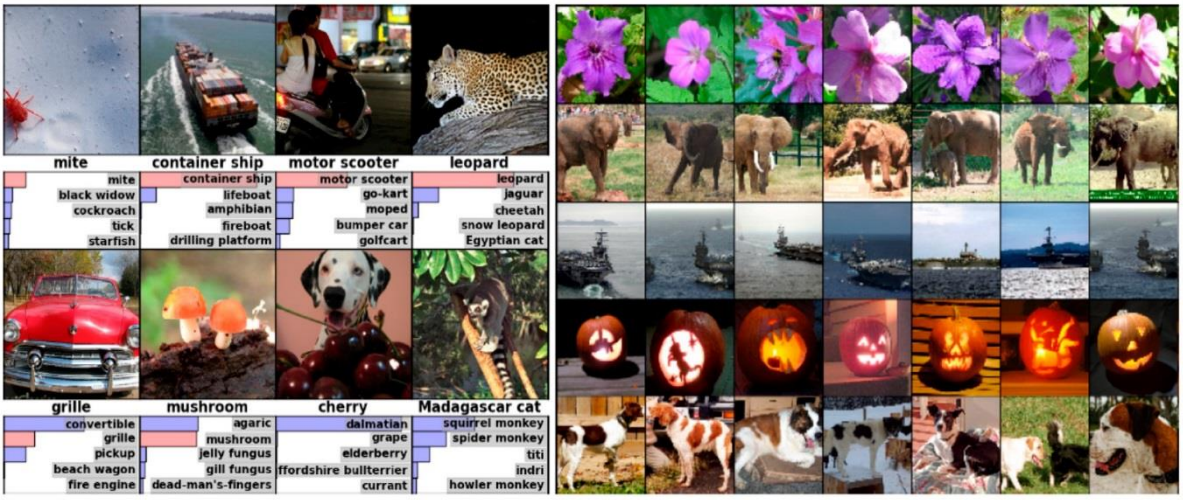

Рис. 2.3 – Приклади роботи нейромережі AlexNet с датасетом ImageNet

На рис. 2.3 зображено приклад роботи AlexNet в рамках змагання ILSVRC-2010. Зліва – вісім тестових зображень ILSVRC-2010 і п'ять ярликів, які найбільш вірогідні на думку моделі. Правильна мітка записується під кожним зображенням, а ймовірність показана червоною смугою, якщо вона знаходиться у верхній п'ятірці. Справа: п'ять тестових зображень ILSVRC-2010 в першому стовпці. В інших стовпцях показано шість навчальних зображень.

Результати показали, що глибока згорткова нейронна мережа здатна досягати рекордних результатів на дуже складних датасетах, використовуючи тільки навчання з учителем. Через рік після публікації AlexNet всі учасники конкурсу ImageNet стали використовувати згорткові нейронні мережі для вирішення задачі класифікації. AlexNet була першою реалізацією згорткових нейронних мереж і відкрила нову еру досліджень.

2.4 GoogLeNet та архітектура Inception

На хвилі успіху AlexNet, Крістіан Жегеді з Google почав перейматися зниженням обсягу обчислень в глибоких нейронних мережах та в результаті створив GoogLeNet – першу архітектуру Inception.

До осені 2014-го моделі глибокого навчання стали дуже корисні в категоризції змісту зображень і кадрів з відео, внаслідок чого багато скептиків визнали користь глибокого навчання і нейронних мереж, а інтернет-гіганти, в тому числі Google, сильно зацікавилися розгортанням на своїх серверних потужностях ефективних і великих мереж.

Крістіан шукав шляхи зменшення обчислювального навантаження в нейронних мережах, домагаючись високої продуктивності або зберігаючи обсяг обчислень, але все одно при цьому підвищуючи продуктивність.

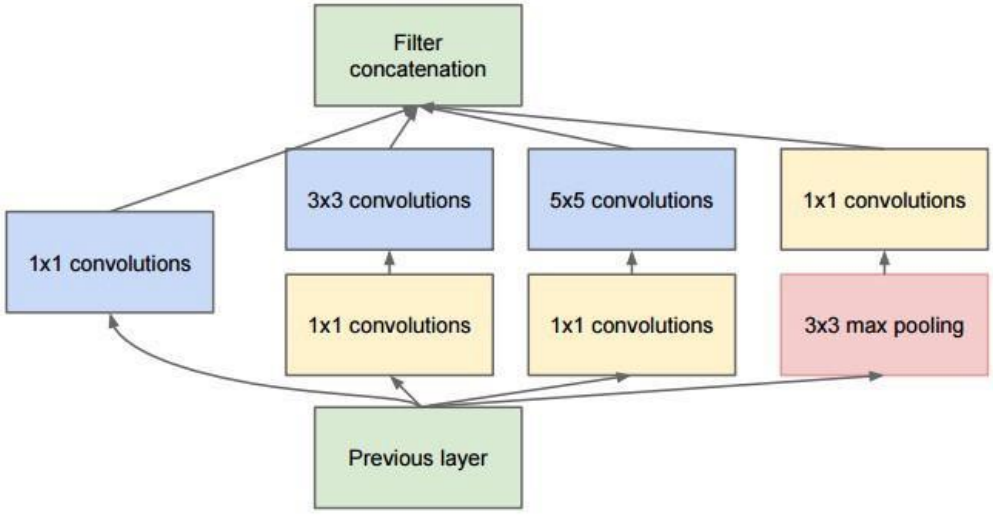

Рис. 2.4 – Архітектура модуля Inception

На рис. 2.4. зображена архітектура модуля Inception. На перший погляд, це паралельна комбінація згорткових фільтрів 1x1, 3x3 і 5x5. Але перевага полягала у використанні згорткових блоків 1x1 для зменшення кількості властивостей перед подачею в складні паралельні блоки. Зазвичай цю частину називають bottleneck [19].

Bottleneck-шар зменшує кількість властивостей (а внаслідок цього і операцій) в кожному шарі, так що швидкість отримання результату можна зберегти на високому рівні. Перш ніж передавати дані в згорткові модулі, кількість властивостей зменшується, скажімо, в 4 рази. Це сильно скоротило обсяг обчислень, що і забезпечило архітектурі популярність.

Нехай у нас є 256 властивостей на вході і 256 на виході, і нехай Inception-шар виконує тільки згортки 3x3. Отримуємо 256x256x3x3 згорток (всього 589 000 операцій множення з накопиченням). Це може виходити за рамки вимог до швидкості обчислень, припустимо, щоб шар оброблявся за 0,5 мілісекунди на Google Server. А при зменшенні кількості властивостей для згортання до 64 (256/4), загальна кількість операцій становить близько 70 000, що зменшує навантаження майже в десять разів. Таким чином, Bottleneck-шари продемонстрували чудову продуктивність на датасетах ImageNet, і стали застосовуватися в більш пізніх архітектурах, таких як ResNet [20].

Та на цьому Крістіан і його команда не зупинилися та у лютому 2015-го року в якості другої версії Inception була представлена архітектура Batch-normalized Inception. В її рамках пакетна нормалізація обчислює середнє і середньоквадратичне відхилення всіх карт розподілу властивостей у вихідному шарі, і нормалізує їх відгуки з цими значеннями. За допомогою цього відгуки всіх нейронних карт лежать в одному діапазоні і з нульовим середнім. Такий підхід полегшує навчання, тому що наступний шар не зобов'язаний запам'ятовувати зміщення вхідних даних і може займатися тільки пошуком кращих комбінацій властивостей.

Наступна версія Inception була представлена у грудні 2015-го року та мала такі ідеї:

– максимізація потоку інформації в мережі за рахунок акуратного балансу між її глибиною і шириною. Перед кожним pooling-му збільшуються карти властивостей;

– зі збільшенням глибини також систематично збільшується кількість властивостей або ширина шару;

– ширина кожного шару збільшується заради збільшення комбінації властивостей перед наступним шаром;

– у міру можливості використовуються тільки згортки 3x3. З оглядом на те, що фільтри 5x5 і 7x7 можна декомпозувати за допомогою кількох 3x3.

2.5 ResNet

У грудні 2015-го року, приблизно в той же час, як була представлена третя версія архітектури Inception, відбулася революція – опублікували ResNet. У ній закладені проста ідея – обхід вхідних даних для наступного шару відбувається за рахунок подачі вихідних даних двох успішних попередніх згорткових шарів.

Такі ідеї вже пропонувалися раніше, але в даному випадку автори обходять два шари одразу і застосовують підхід у великих масштабах. Обхід одного шару не дає особливої вигоди, а обхід двох – ключова знахідка. Це можна розглядати як маленький класифікатор або мережа-в-мережі. Також це був перший в історії приклад навчання мережі з кількох сотень, навіть тисяч шарів.

У багатошаровій ResNet застосували bottleneck-шар, аналогічний тому, що застосовується в Inception – він зменшує кількість властивостей в кожному шарі, спочатку використовуючи згортку 1x1 з меншим виходом (зазвичай чверть від входу), потім йде шар 3x3, а потім знову згортка 1x1 в більшу кількість властивостей. Як і у випадку з Inception-модулями, це дозволяє економити обчислювальні ресурси, зберігаючи багатство комбінацій властивостей. В якості фінального класифікатора в ResNet використовується pooling-шар з softmax [21].

Таким чином, архітектуру ResNet можна розглядати як систему одночасно паралельних і послідовних модулів – у багатьох модулях in/out-сигнал проходить паралельно, а вихідні сигнали кожного модуля з'єднуються послідовно.

Оскільки вихідний сигнал повертається назад і подається як вхідний, ResNet можна вважати поліпшеною правдоподібною моделлю кори головного мозку

3 ДОСЛІДЖЕННЯ ЕФЕКТИВНОСТІ МЕТОДІВ РОЗПІЗНАВАННЯ

3.1 Планування експерименту

При виборі засобів для розпізнавання зображень потрібно враховувати їх актуальність, легкість підтримки та доречність використання в межах даної галузі. Мабуть, багато хто погодиться, що OpenCV є найбільш відомою бібліотекою комп'ютерного зору. За довгий час свого існування вона придбала велику аудиторію користувачів і стала, де-факто, стандартом в області комп'ютерного зору. Безліч алгоритмів, що працюють «з коробки», відкритість вихідного коду, чудова підтримка та можливість користуватися бібліотекою на мовах C, C++, Python, а також Matlab, C# та Java під різними операційними системами дозволяє OpenCV залишатися затребуваною. Окрім того, OpenCV не стоїть на місці – в неї постійно додається новий функціонал. Не стала винятком і область глибокого навчання – OpenCV дозволяє завантажувати та отримувати результати (передбачення) за допомогою моделей, створених в будь-якому з трьох популярних фреймворків – Caffe, TensorFlow або Torch. Також бібліотека забезпечує швидку роботу на CPU та підтримку основних верств нейронних мереж [22].

Бібліотека OpenCV являє собою набір модулів, кожен з яких пов'язаний з певною областю комп'ютерного зору. Існує стандартний набір модулів – так би мовити, «must have» для будь-якого завдання комп'ютерного зору. Реалізуючи відомі алгоритми, дані модулі добре опрацьовані і протестовані. Всі вони представлені в основному репозиторії OpenCV. Також існує репозиторій з додатковими модулями, що реалізують експериментальну або нову функціональність. Вимоги до експериментальних модулів, зі зрозумілих причин, м'якше. І, як правило, коли якийсь із таких модулів стає досить розвиненим, сформованим і затребуваним, він може бути перенесений в основний репозиторій.

Одним з таких модулей, що не так давно зайняв почесне місце у основному репозиторії є модуль глубокого навчання dnn, за допомогою якого і був запланований експеримент.

В останні роки в багатьох областях глибоке навчання показує результати, що значно перевершують аналогічні у класичних алгоритмів. Це стосується і галузі комп'ютерного зору, де маса завдань вирішується із застосуванням нейронних мереж. У світлі цього факту було логічним дати користувачам OpenCV можливість роботи з нейронними мережами.

Замість використання вже існуючих реалізацій розробниками було прийнято рішення написання свого і на те є кілька причин.

По-перше, це дозволяє досягти легковажності рішення – залишаючи тільки можливість виконання прямого проходу по мережі, можна спростити код, прискорити процес установки і зборки.

По-друге, своя реалізація дозволяє звести зовнішні залежності до мінімуму. Це спрощує поширення додатків, що використовують dnn. І, якщо раніше в проекті використовувалася бібліотека OpenCV, не важко буде додати в такий проект підтримку глибоких мереж.

Так само, своє рішення дозволяє зробити його універсальним, не прив'язаним до якогось конкретного фреймворку, його обмеженням і недолікам. При наявності власної імплементації доступні всі шляхи для оптимізації і прискорення коду.

Власний модуль для запуску глибоких мереж значно спрощує процедуру створення гібридних алгоритмів, що поєднують в собі швидкість класичного комп'ютерного зору і чудову узагальнюючу здатність глибоких нейронних мереж.

Варто зауважити, що модуль не є, строго кажучи, повноцінним фреймворком для глибокого навчання. На даний момент в модулі представлена виключно можливість отримання результатів роботи мережі.

Основна можливість dnn полягає, звичайно ж, в завантаженні і запуску нейронних мереж. При цьому модель може бути створена в будь-якому з трьох фреймворків глибокого навчання – Caffe, TensorFlow або Torch; спосіб її завантаження і використання зберігається незалежно від того, де вона була створена.

Таким чином, підтримуючи відразу три популярних фреймворки, ми можемо досить легко комбінувати результати роботи завантажених з них моделей без необхідності створювати все заново в одному єдиному фреймворку.

При завантаженні відбувається конвертація моделей у внутрішнє представлення, близьке до використовуваного в Caffe. Так сталося в силу історичних причин – підтримка Caffe була додана найпершою. Однак взаємно однозначної відповідності між уявленнями немає.

Крім того, бібліотека підтримує всі основні верстви, починаючи від базових і закінчуючи більш спеціалізованими – всього понад 30.

Крім підтримки окремих верств, важлива також і підтримка конкретних архітектур нейронних мереж. Модуль містить приклади для класифікації (AlexNet, GoogLeNet, ResNet, SqueezeNet), сегментації (FCN, ENet), детектування об'єктів (SSD) та інші.

Таким чином, в рамках експерименту було вирішено протестувати не лише результати роботи однієї з наявних нейронних мереж, але й наявних моделей OpenCV для роботи з ними, що дозволило зробити висновки щодо якості натренованих моделей, оскільки дуже часто боротьба на вершині рейтингу кращих моделей йде за частки відсотків якості.

3.2 Дослідження та оцінка ефективності розпізнавання зображень

Для задачі тестування були обрані різні моделі на основі OpenCV з числа наявних прикладів для різних фреймворків і різних завдань: AlexNet (Caffe), GoogLeNet (Caffe), GoogLeNet (TensorFlow), ResNet-50 (Caffe), SqueezeNet v1.1 (Caffe) для завдання класифікації об'єктів та FCN (Caffe), ENet (Torch) для завдання семантичної сегментації.

При проведенні експерименту було вирішено зосередитися на таких порівняльних характеристиках як:

– опубліковане та виміряне значення (в вихідному фреймворку та dnn);

– показники середньої та максимальної різниці між вихідними тензорами фреймворку та dnn.

Результати експерименту наведені в Таблицях 3.1 і 3.2.

Таблиця 3.1 – Результати оцінки якості для задачі класифікації.

Модель (вихідний фреймворк)	Опубліковане значення acc @ top-5	Виміряне значення acc @ top-5 в вихідному фреймворку	Виміряне значення acc @ top-5 в dnn	Середня різниця на елемент між вихідними тензорами фреймворка і dnn	Максимальна різниця між вихідними тензорами фреймворка і dnn
AlexNet (Caffe)	80.2%	79.1%	79.1%	6.5E-10	3.01E-06
GoogLeNet (Caffe)	88.9%	88.5%	88.5%	1.18E-09	1.33E-05
GoogLeNet (TensorFlow)	—	89.4%	89.4%	1.84E-09	1.47E-05
ResNet-50 (Caffe)	92.2%	91.8%	91.8%	8.73E-10	4.29E-06
SqueezeNet (Caffe)	80.3%	80.4%	80.4%	1.91E-09	6.77E-06

Вимірювання для завдання класифікації об'єктів проводились на наборі ImageNet 2012 (ILSVRC2012, 50000 прикладів).

Таблиця 3.2 – Результати оцінки якості для завдання семантичної сегментації.

Модель (вихідний фреймворк)	Опублікован е значення mean IOU	Виміряне значення mean IOU в вихідному фреймворку	Виміряне значення mean IOU в dnn	Середня різниця на елемент між вихідними тензорами фреймворка і dnn	Максимальна різниця між вихідними тензорами фреймворка і dnn
FCN (Caffe)	65.5%	60.402874%	60.402879%	3.1E-7	1.53E-5
ENet (Torch)	58.3%	59.1368%	59.1369%	3.2E-5	1.20

Результати завдання семантичної сегментації для FCN обчислені на валідаційному наборі сегментаційної частини PASCAL VOC 2012 (736 прикладів). Результати для ENet обчислені на валідаційному наборі Cityscapes (500 прикладів).

Слід сказати кілька слів про те, який сенс мають зазначені вище числа. Для задач класифікації загальноприйнятої метрикою якості моделей є точність для топ-5 відповідей мережі (accuracy @ top-5, [1]): якщо правильна відповідь є серед 5 відповідей мережі з максимальними показниками впевненості, то дану відповідь мережі зараховується як вірний. Відповідно, точність – це відношення числа вірних відповідей до числа прикладів. Даний спосіб вимірювання дозволяє врахувати не завжди коректну розмітку даних, коли, наприклад, відзначається об'єкт, який займає далеко не центральне положення на кадрі.

Для завдань семантичної сегментації використовуються кілька метрик – попіксельна точність (pixel accuracy) і середнє по класах відношення перетину до об'єднання (mean intersection over union, mean IOU) [5].

Попіксельна точність – це відношення кількості правильно класифікованих пікселів до кількості всіх пікселів. mean IOU – більш складна характеристика: це усереднене за класами відношення правильно відзначених пікселів до суми числа пікселів даного класу і числа пікселів, позначених даний клас.

З таблиць слід, що для задач класифікації та сегментації різниця в точності між запусками моделі в оригінальному фреймворку і в dnn відсутня. Цей факт означає, що модуль можна сміливо використовувати, не побоюючись непередбачуваних результатів.

Різницю між опублікованими та отриманими в експериментах числами можна пояснити тим, що автори моделей проводять всі обчислення з використанням GPU, в той час як я використовував CPU. Також, як вже було відмічено, різні бібліотеки можуть по-різному декодувати формат jpeg. Це могло позначитися на результатах для FCN, так як датасет PASCAL VOC 2012 містить зображення саме цього формату, а моделі для семантичної сегментації виявляються досить чутливі до зміни розподілу вхідних даних.

Як можна помітити, в таблиці 4.2 присутня аномально велика максимальна різниця виходів dnn і Torch для моделі ENet. Мабуть, це пов'язано з тим, що модель ENet використовує кілька операцій MaxPooling. Дана операція вибирає максимальний елемент в околиці кожної позиції і записує у вихідний тензор це максимальне значення, а також передає далі індекси обраних максимальних елементів.

Ці індекси далі використовуються операцією, в деякому сенсі зворотного даної – MaxUnpooling. Вона записує елементи вхідного тензора в позиції вихідного, відповідні тим самим індексам. У цьому місці і виникає велика помилка: в певній околиці операція MaxPooling вибирає елемент з неправильним індексом; при цьому різниця між правильним виходом Torch і виходом dnn для даного шару лежить в межах обчислювальної похибки (10E-7), а різниця в індексах відповідає сусіднім елементам околиці.

Тобто, в результаті невеликої флуктуації сусідній елемент став трохи більше, ніж елемент з правильним індексом. Результат операції MaxUnpooling, при цьому, залежить не тільки від виходу попереднього шару, але і від індексів відповідної операції MaxPooling, яка розташовується набагато раніше (на початку обчислювального графа моделі). Таким чином, MaxUnpooling записує елемент з правильним значенням в невірну позицію. В результаті, накопичується помилка.

На жаль, усунути цю помилку неможливо, так як першопричини її появи пов'язані, швидше за все, з незначними відмінами у реалізації алгоритмів, використаних при тренуванні і не пов'язані з наявністю помилки в реалізації.

Однак, справедливо відзначити, що середня помилка на елемент вихідного тензора залишається низькою – тобто помилки в індексах виникають досить рідко. Більш того, наявність цієї помилки не призводить до погіршення якості роботи моделі, про що свідчать числа в тій же таблиці 3.2.

Розглянуті основні результаті ефективності роботи нейронних мереж з їх особливостями показали, що вони мають усі переваги для використання у галузях комп’ютерного зору, а ефективного навчання розпізнавання об’єктів можна досягти саме методом зворотнього поширення помилки.

ВИСНОВКИ

Під час атестаційної магістерської роботи було досліджено ефективність моделей класифікації зображень.

В роботі виконано аналіз сучасних методів розпізнавання зображень, створена класифікація типів існуючих підходів до розпізнавання.

Огляд методів вирішення задач комп'ютерного зору за групами обґрунтував переваги нейромережевого методу над усіма запропонованими. Принцип роботи нейронів штучної мережі дозволяє виявити його активність, що визначається його параметрами. Труднощі при спотвореннях, такі як зміна образу у розмірах і поворот зображення, вирішуються при моделюванні неокогнітрона, що використовує якісно нову архітектуру. В основу архітектури неокогнітрона покладена організація зорової системи живих істот. Розглянуті основні принципи навчання нейронних мереж з їх особливостями показали, що ефективного навчання розпізнавання об'єктів можна досягти саме методом зворотнього поширення помилки.

В ході атестаційної роботи магістра було:

– проведено аналіз методів розпізнавання зображень;

– проведено аналіз існуючих згорткових нейронних мереж для класифікації зображень;

– проведено експериментальне дослідження ефективності існуючих згорткових нейронних мереж на базі бібліотеки OpenCV;

– проведено експериментальне дослідження ефективності фреймворків глибинного навчання на базі бібліотеки OpenCV;

– за результатами проведеного дослідження була виконана оцінка ефективності застосування існуючих згорткових нейронних мереж на базі бібліотеки OpenCV та сформовано рекомендації щодо застосування існуючих фреймворків глибинного навчання на базі цієї бібліотеки.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Sebe, N. Machine learning in computer vision, 2005. – p. 29.

2. Murphy-Chutorian, E. Head pose estimation in computer vision: A survey. Pattern Analysis and Machine Intelligence, IEEE Transactions, 2009. – p. 607-626.

3. Grauman, K. Visual object recognition, 2010. – p. 165-186.

4. LeCun, Y. Backpropagation applied to handwritten zip code recognition, 1989. – p. 541-551.

5. Duin, R. P. W. Open issues in pattern recognition, 2005. – p. 27-42.

6. Lee, H. Efficient sparse coding algorithms. Advances in neural information processing systems, 2006. – p. 801-808.

7. Turk, M. A. Face recognition using eigenfaces, 1991. – p. 586-591.

8. Olshausen, B. A. Emergence of simple-cell receptive field properties by learning a sparse code for natural images, 1996. – p. 607-609.

9. Hinton, G.E. A practical guide to training restricted Boltzmann machines, 2010. – p. 926.

10. Hubel, D. H. Receptive fields and functional architecture of monkey striate cortex, 1998. – p. 215-243.

11. Krizhevsky, A. Imagenet classification with deep convolutional neural networks, 2012. – p. 1097-1105.

12. Hubel, D. H. Eye, brain, and vision, 1988. – 85-87 pp.

13. Bradski, G. The OpenCV library, 2000. – p. 120–126.

14. Rublee, E. ORB: an efficient alternative to SIFT or SURF. Computer Vision (ICCV), 2011. – p. 256-271.

15. Pazke, A. An analysis of deep neural network models for practical applications, 2010. – p. 12-24.

16. Hinton G. E., Srivastava N., Krizhevsky A., Sutskever I. Improving neural networks by preventing co-adaptation of feature detectors, 2012. – p. 7-18.

17. Min L., Qiang C., and Shuicheng Y. Network in network, 2013. – p. 9-14.

18. Krizhevsky, A., Sutskever I., Hinton G. E. Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems, 2012. – p. 17–25.

19. Zisserman A. Very deep convolutional networks for large-scale image recognition, 2015. – p. 25-28.

20. Szegedy C., Vanhoucke V., Ioffe S., Wojna Z. Rethinking the inception architecture for computer vision, 2015. – p. 8-19.

21. Sangdoo Y. CutMix: Regularization Strategy to Train Strong Classifiers with Localizable Features, 2011 – p. 13–20.

22. Bradski, G. The OpenCV library, 2000. – p. 128–141.