

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет комп'ютерної інженерії та управління
(повна назва)

Кафедра електронних обчислювальних машин
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

Рівень вищої освіти другий (магістерський)

Алгоритми кластеризації зображень на базі
штучної нейронної мережі

(тема)

Виконав:

студент II курсу, групи КСМм-21-1
Житник М.С.
(прізвище, ініціали)

Спеціальність 123 «Комп'ютерна інженерія»
(код і повна назва спеціальності)

Тип програми освітньо-професійна
(освітньо-професійна або освітньо-наукова)

Освітня програма Комп'ютерні системи та мережі
(повна назва освітньої програми)

Керівник: зав. каф. Коваленко А.А.
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри ЕОМ

(підпис)

Коваленко А.А.

(прізвище, ініціали)

2022 р.

Харківський національний університет радіоелектроніки

Факультет _____ комп'ютерної інженерії та управління _____

Кафедра _____ електронних обчислювальних машин _____

Рівень вищої освіти _____ другий (магістерський) _____

Спеціальність _____ 123 «Комп'ютерна інженерія» _____
(код і повна назва)

Тип програми _____ освітньо-професійна _____
(освітньо-професійна або освітньо-наукова)

Освітня програма _____ Комп'ютерні системи та мережі _____
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)

“ _____ ” _____ 20__ р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

студенту _____ Житнику Миколі Сергійовичу _____
(прізвище, ім'я, по батькові)

1. Тема роботи _____ Алгоритми кластеризації зображень на базі штучної нейронної мережі _____

затверджена наказом по університету від “ 07 ” листопада 2022 р. № 1453 Ст

2. Термін подання студентом роботи до екзаменаційної комісії _____ 13 грудня 2022 р.

3. Вхідні дані до роботи _____

згортовка штучна нейронна мережа

кластеризація

обробка зображень

AlexNet

4. Перелік питань, що потрібно опрацювати у роботі _____

Обробка зображень

Модифіковані методи кластеризації методом k-середніх

Розробка штучної нейронної мережі для кластеризації та ранжування зображень

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів) 16 слайдів

6. Консультанти розділів роботи (заповнюється за наявності консультантів згідно з наказом, зазначеним у п.1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Отримання завдання. Аналіз літератури	08.11.2022–14.11.2022	
2	Огляд існуючих алгоритмів.	15.11.2022–21.11.2022	
3	Аналіз існуючих архітектур ЗНМ.	22.11.2022–29.11.2022	
4	Реалізація розглянутих алгоритмів	30.11.2022–02.12.2022	
5	Вибір баз даних для тестування.	03.12.2022–04.12.2022	
6	Отримання результатів	05.12.2022–06.12.2022	
7	Оформлення ПЗ	07.12.2022–12.12.2022	

Дата видачі завдання 07 листопада 2022 р.

Студент _____
(підпис)

Керівник роботи _____
(підпис)

зав. каф. Коваленко А.А.
(посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи: 61 с., 20 рис., 1 дод., 18 джерел.

ГЛИБОКА НЕЙРОННА МЕРЕЖА, ЗГОРТКОВА НЕЙРОННА МЕРЕЖА, НАБІР ДАНИХ, ALEXNET, КЛАСТЕРИЗАЦІЯ, ILSVRC, MNIST, ГІПЕРПАРАМЕТР.

Метою кваліфікаційної роботи є аналіз алгоритмів кластеризації зображень на базі штучної нейронної мережі.

У ході виконання кваліфікаційної роботи проведено аналіз існуючих алгоритмів кластеризації даних, а також аналіз існуючих архітектур згорткових нейронних мереж. Розроблено архітектуру самонавчальної згорткової нейронної мережі з поєднаними процедурами навчання, що використовуює метод кластеризації k-середніх з додаванням адаптаційних шарів.

ABSTRACT

Master's thesis: 61 pages, 20 figures, 1 appendices, 18 sources.

DEEP NEURAL NETWORK, CONVOLUTIONAL NEURAL NETWORK, DATASET, ALEXNET, CLUSTERIZATION, ILSVRC, MNIST, HYPERPARAMETER.

The major goal of this thesis is the analysis of image clustering algorithms based on an artificial neural network.

In order to comparative analysis of existing data clustering algorithms was carried out, as well as an analysis of existing convolutional neural network architectures. The architecture of a self-learning convolutional neural network with combined learning procedures, which uses the k-means clustering method with the addition of adaptive layers, has been developed.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ	8
ВСТУП	9
1 СУЧАСНА ОБРОБКА ЗОБРАЖЕНЬ.....	12
1.1 Пошук зображення за змістом	12
1.2 Вилучення властивостей зображень	16
1.2.1 Колірна гістограма	16
1.2.2 FAST	17
1.2.3 SIFT-детектор	18
1.2.4 PCA-SIFT детектор	19
1.2.5 SURF-детектор	20
1.2.6 F-SIFT детектор	21
1.3 Методика видалення зображень, заснована на нейромережах	22
1.4 Кластеризація зображень методом k-середніх.....	26
2 МОДИФІКОВАНІ АЛГОРИТМИ КЛАСТЕРИЗАЦІЇ МЕТОДОМ К- СЕРЕДНІХ	29
2.1 Алгоритм кластеризації з множинною випадковою ініціалізацією	29
2.2 Кластеризація методом k-середніх ++	30
2.3 Аналіз архітектур нейронних мереж.....	33
2.3.1 Глибокі нейронні мережі.....	33
2.3.2 Згорткові нейронні мережі.....	35
2.3.3 Налаштування гіперпараметрів	39
3 РОЗРОБКА ШТУЧНОЇ НЕЙРОННОЇ МЕРЕЖІ ДЛЯ КЛАСТЕРИЗАЦІЇ ЗОБРАЖЕНЬ.....	41
3.1 Вибір архітектури мережі.....	41
3.2 Навчання ознакам та кластеризація	44
3.3 Проведення експериментів	47

ВИСНОВКИ.....	50
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	51
ДОДАТОК А Графічний матеріал кваліфікаційної роботи.....	53

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ
І ТЕРМІНІВ

CCNN – кластеризуюча згоркова нейронна мережа (КЗНМ)

CNN – згорткова нейронна мережа (ЗНМ)

NMI – нормалізація взаємної інформації (від англ. Normalized Mutual Information)

SIFT – масштабно-інваріантна трансформація ознак

SURF – детектор стійких прискорених ознак

VGG – згорткова мережа для виділення ознак

ВСТУП

Обробка зображень є одним із основних напрямів обробки інформації у сучасних інформаційних системах різного призначення та має широкий спектр потенційних областей застосування, таких як пошукові системи (у тому числі, в мережі Інтернет); каталогізація зображень предметів мистецтва; аналіз результатів медичних досліджень; контроль над вмістом графічної інформації; додатки у військовій сфері. Однак слід зауважити, що зараз найбільш значних успіхів досягнуто у вузьких областях, коли спочатку є уявлення про вид і структуру об'єктів, пов'язаних з зображенням. Тому на сьогоднішній день такі популярні та успішні сервіси розпізнавання осіб та різних об'єктів, простежується успіх у аналізі томографічних знімків та клітинної будови організмів, активно розвиваються системи електронної комерції, що включають до своєї архітектури підсистеми категоризації (каталогізації) товарів.

Багато в чому така ситуація склалася з тієї причини, що надзвичайно складно вигадати методикку, яка б дозволяла машинним способом досягти розпізнавання довільних образів. Цю проблему дослідники пояснюють наявністю семантичного розриву між справжнім сприйняттям образу людиною та обчислювальною методикою побудови цього сприйняття. Людина сприймає навколишній світ через систему понять та уявлень. Поняття (лінгвістичний опис) описують, відображають уявлення (візуальні об'єкти). Людське мислення дозволяє відволіктися від несуттєвих сторін об'єкта і виділити з нього найважливіші ознаки. Результатом є поява абстрактних понять – колір, кривизна, форма, краса та інші.

Наявність системи понять дозволяє систематизувати уявлення, розбити їх за різними категоріями з урахуванням ступеня подібності. Для визначення подібності зображень у машинному аналізі вводиться поняття міри відстані у просторі зображень, параметрами якого є колір, текстура та форма. Залежно

від значень відстані приймається рішення про схожість/відмінність двох зображень. У просторі зображень можна ввести додаткові характеристики, які дозволять ще точніше визначати подібність зображень. Тоді можна провести явну паралель між простором зображень та багатовимірним простором ознак. Така відповідність дозволяє застосовувати до обробки зображень математичний апарат, що імітує спосіб сприйняття інформації людиною. Зокрема, таким апаратом є нейронні мережі, що наближають машинний аналіз зображення до людського. Без їх застосування довелося б мати справу лише із низькорівневою візуальною інформацією. Як би не були досконалі методики аналізу подібної інформації, вони не зможуть відобразити механізми людського сприйняття, що вкрай важливо для пошукової системи.

Існує значна кількість моделей та методів, широко застосовуваних в області пошуку зображень за змістом, серед яких відзначимо такі підходи:

- спектральна кластеризація на графах;
- категоризації та ранжирування зображень на основі імовірнісних гіперграфів;
- модель узагальнення результатів, що базується на методі опорних векторів;
- адитивні нечіткі системи та нечіткі класифікатори;
- двомірний кратномасштабний аналіз, побудований на прихованих ланцюгах Маркова;
- згорткові нейронні мережі.

Розробка нових та вдосконалення існуючих математичних моделей аналізу зображень є ключем до досягнення максимальної ефективності таких пов'язаних з обробкою зображень областей як пошук зображень за змістом, класифікація та кластеризація, при цьому використання нейромережевих технологій обробки графічної інформації сприяє зниженню семантичного розриву між очікуваннями користувача та машинним аналізом.

Обробка зображень є одним із напрямків розвитку інформаційних технологій стосовно прикладних систем різного призначення, зокрема, систем комп'ютерного зору. При вирішенні завдань комп'ютерного зору, як правило, виникає питання про ефективний та швидкий розподіл зображень за групами (класами, кластерів) на основі візуальних ознак. В останнє десятиліття особливих успіхів вдалося досягти у цифровізації комерційного сектору економіки, у результаті з'явилася нова сфера – електронна комерція, що включає фінансові та торгові транзакції, виконувані за допомогою комп'ютерних мереж та бізнес-процеси, пов'язані із проведенням таких транзакцій.

Системи електронної комерції дозволяють удосконалювати технології комунікації, створюючи нові способи пошуку ніші на ринку для підприємства та цільової аудиторії для продавця; ефективно формувати споживчі переваги в суспільстві; знижувати ризики виходу на ринок товарів та послуг, які не користуються попитом у населення.

Актуальність теми дослідження обумовлена необхідністю вдосконалення існуючих та розробки нових методів обробки зображень у великих базах даних, пов'язаних із розв'язанням задачі кластеризації зображень за змістом, а також їх ранжування.

Метою кваліфікаційної роботи є аналіз алгоритмів кластеризації зображень на базі штучної нейронної мережі.

Завданнями є:

- аналіз існуючих алгоритмів кластеризації даних;
- аналіз існуючих архітектур згорткових нейронних мереж;
- розробка архітектури самонавчальної згорткової нейронної мережі з поєднаними процедурами навчання, використовуючи метод кластеризації k-середніх з додаванням адаптаційних шарів;
- програмна реалізація розглянутих алгоритмів.

1 СУЧАСНА ОБРОБКА ЗОБРАЖЕНЬ

1.1 Пошук зображення за змістом

В даний час одним із затребуваних і активно розвиваються напрямів обробки інформації є обробка зображень. Обсяги архівів зображень у мережі Інтернет щодня поповнюються з неймовірною швидкістю. Цьому сприяє поява цифрових камер, комп'ютерів та інших носіїв, здатних зберігати гігабайти інформації. За даними звітів 2020 та 2021 років застосунок, який надає користувачам можливість ділитися фотографіями, відвідували 1 млрд користувачів щомісяця, при цьому кількість фотографій становила близько 7 млрд. Для зберігання такої кількості фотографій потрібні величезні бази даних зображень. А для швидкої обробки потрібні швидкі системи аналізу, пошуку та індексування зображень. Можна виділити кілька основних напрямків:

- геометричні перетворення та колірна корекція;
- відновлення та реконструкція зображень, фільтрація для придушення різних шумів на зображеннях;
- передобробка з метою покращення якості зображення, виявлення меж об'єктів;
- розробка та вдосконалення критеріїв оцінки візуальної якості зображень;
- стиснення зображень;
- розпізнавання зображень (формування опису заданого) зображення або віднесення зображення до певного класу.

Обробка зображень є одним із напрямків розвитку інформаційних технологій стосовно прикладних систем різного призначення, зокрема, систем комп'ютерного зору.

При вирішенні завдань комп'ютерного зору, як правило, виникає питання про ефективному та швидкому розподілі зображень за групами (класами, кластерів) на основі візуальних ознак. До основних класів прикладних систем обробки зображень відносяться такі:

- системи комп'ютерного зору для технологічних процесів;
- системи діагностики та моніторингу станів об'єктів;
- системи пошуку та розпізнавання об'єктів (для обробки медичних зображень, зображень карток земної поверхні; фотографій та інших складних зображень);
- системи відеоспостереження;
- технології автоматизації документообігу (системи обробки зображень документів; системи оптичного розпізнавання символів).

Найважливішими областями застосування методів обробки зображень є комп'ютерне моделювання, автоматизовані системи управління різними процесами та системи автоматизації (наукових досліджень, проектування, конструювання, виробництва), електронна комерція, цифрова обробка зображень у космічному дослідженні та дистанційне зондування, засоби масової інформації та видавничі системи, мистецтво та мультимедіа.

Пошук зображень за змістом – це розділ комп'ютерного зору, який вирішує завдання пошуку зображень, які мають необхідний зміст великого набору цифрових зображень [3]. "За змістом" означає, що пошук аналізує вміст зображення, а не метадані, такі як: ключові слова, теги або описи, асоційовані з зображенням. До змісту зображення можна віднести колір, форму, текстуру та іншу інформацію, яка може бути витягнута з зображення.

Системи пошуку зображень за змістом (CBIR) вирішують проблему отримання бажаних для користувача зображень із цифрової бібліотеки. У своїй основі система пошуку зображень – це комп'ютерна система разом з необхідною апаратною частиною та програмним забезпеченням, метою якої є пошук у досить великій цифровій базі даних зображень або бібліотеки та отримання результатів, схожих на зображення-запит користувача.

База даних зображень – це колекція зображень, розбитих на окремі асоціативні групи. Першим термін СВІР ввів у вжиток дослідник Като [4] у 1992 р. році для опису експерименту автоматичного пошуку цифрових зображень шляхом порівняння кольору зображення та характеристик форми кожного зображення з бази даних з кольором та характеристиками форми вхідного зображення. З того часу цей термін було перенесено на всі схожі методології та процеси пошуку зображень з бази даних, які використовують загальні характерні властивості, такі як кольори, форми, текстури і т.і.

Ранні СВІР-системи зазвичай покладалися на вилучення характеристик з зображення та стратегії зіставлення для пошуку релевантних зображень у базі даних.

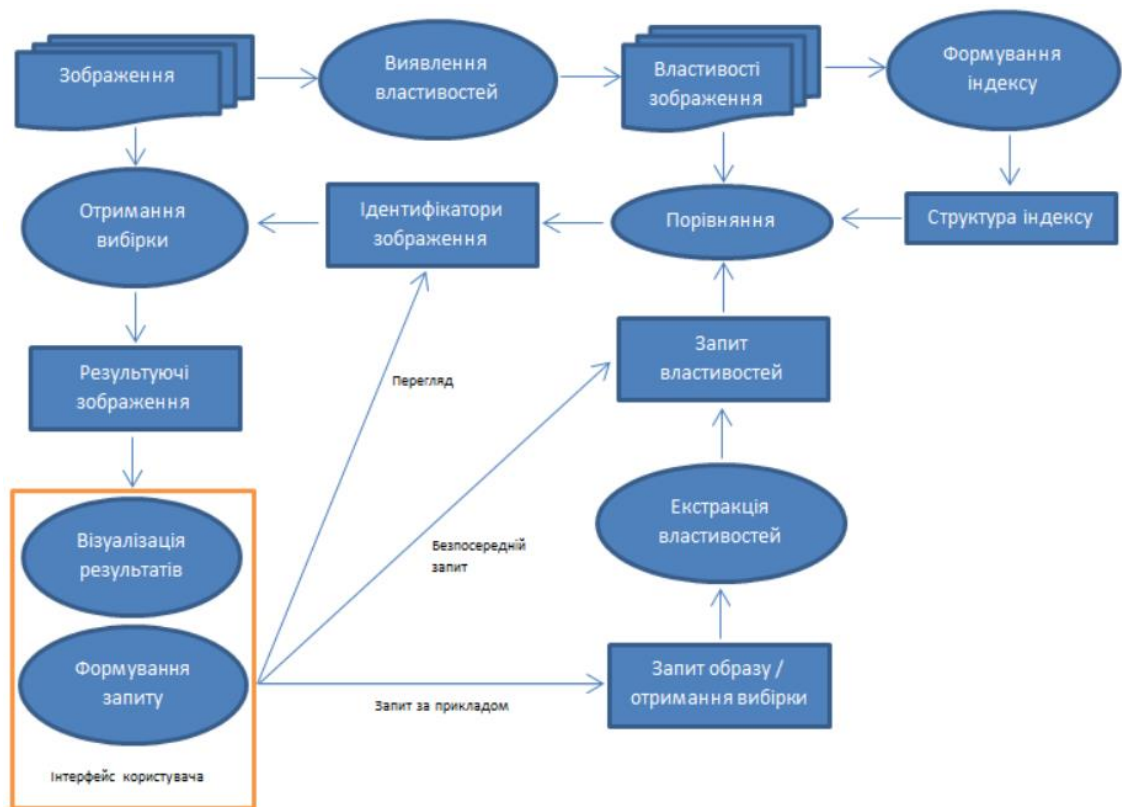


Рисунок 1.1 – Архітектура СВІР-систем

СВІР методики є життєздатними рішеннями для знаходження потрібних зображень у цифрових бібліотеках. У найпростішій СВІР системі

всі цифрові зображення в бібліотеці представлені візуальними характеристиками. Типові візуальні характеристики включають кольори, форми, кордони та текстури для представлення зображення. Спочатку ці візуальні характеристики витягуються з кожного зображення та зберігаються в базі даних властивостей зображень, відповідній цифровій бібліотеці зображень, для спрощення подальшого використання. Потім застосовується метод відповідності для визначення подібності між візуальними характеристиками зображення запиту та візуальними характеристиками всіх цифрових зображень у базі даних зображень. Тільки зображення, що мають найвищі значення подібності, повертаються кінцевому користувачеві як результати. На рисунку 1.1 зображена архітектура найпростішої системи CBIR.

Як видно із рисунку 1.1, система пошуку зображень повинна підтримувати такі функції:

- вилучення характеристик із зображень;
- пошук зображень за характеристиками;
- забезпечення ефективного зберігання зображень.

Розглянемо докладніше першу функцію зі списку, друга функція є по суті похідною з першої, для реалізації останньої функції використовуються готові рішення, такі як ImageNet [12] або ProductNet [13]. Як можна було переконатися, вилучення характеристик із зображень є найважливішою частиною CBIR, тому що без цього модуля системи неможливо організувати зіставлення зображень. Зіставлення – операція порівняння двох і більше об'єктів із метою визначення ступеня їхньої подібності.

Отримання надійного зіставлення пари зображень – це важке завдання, для його вирішення було винайдено безліч методик, але жодна з них не є універсальною. Вибір відповідної методики для вирішення завдання зіставлення багато в чому залежить від типу співвідносних зображень та відмінностей між зображенням та його парою по кількох параметрам, таким, як:

- масштаб (англ. scale): принаймні два елементи всередині пари зображень мають різний масштаб;
- оклюзія (англ. occlusion): концепція того, що два об'єкти, просторово розділені у тривимірному просторі, можуть накладатися один на одного у двовірному;
- орієнтація (англ. orientation): зображення у парі повернуті друг щодо друга;
- афінне перетворення: об'єкт плоский, текстурований (присутність об'ємних геометричних патернів) або з виділеними межами;
- розмивання (англ. blurring): розмиття об'єктів, що швидко рухаються в нерухоме зображення;
- освітлення (англ. illumination): різниця у освітленні об'єктів представляє типову проблему для точного зіставлення характеристик зображень.

1.2 Вилучення властивостей зображень

Розглянемо відомі підходи до отримання властивостей та наведемо їх порівняльний аналіз.

1.2.1 Колірна гістограма

Колірна гістограма (також графік рівнів або просто рівні) – це гістограма рівнів насиченості зображення (сумарна або розділена по колірних каналах), є поданням розподілу різних кольорів у зображенні. Колірна гістограма дозволяє оцінити кількість та різноманітність відтінків зображення, а також загальний рівень яскравості зображення. Колірні гістограми є гнучкою конструкцією, яка може бути побудована з зображень у різних колірних просторах, будь то RGB або будь-який інший колірний простір будь-якого виміру [16].

Основний недолік гістограм для класифікація полягає в тому, що уявлення залежить від кольору досліджуваного об'єкта, але його форма і текстура не враховуються. Кольорові гістограми можуть бути ідентичними для двох зображень з різним змістом, але схожою інформацією кольорів. І навпаки, без просторової інформації або інформації про форму схожі об'єкти різного кольору можуть бути визначені методикою як різні лише на основі порівняння колірних гістограм. Неможливо відрізнити червоно-білу чашку від червоно-білої тарілки. Іншими словами, алгоритми, засновані на гістограмі, не оперують такими уявленнями, як, наприклад, "чашка", а знання про подання червоно-білої чашки марно, якщо їй протиставлена інша ідентична синьо-біла чашка.

1.2.2 FAST

Алгоритм FAST (англ. Features from Accelerated Segment Test) – це алгоритм на основі кутового критерію SUSAN (англ. Smallest Univalued Segment Assimilating Nucleus) [18]. Для виявлення ознак SUSAN18 поміщає круглу маску над пікселем (ядром), що перевіряється. Область маски – M і піксель у цій масці позначений як m , і кожен піксель порівнюється з ядром, використовуючи функцію порівняння такого виду:

$$C(m) = e^{-\left(\frac{I(m) - I(m_0)}{t}\right)^6}, \quad (1.1)$$

де t означає радіус, а значення показника ступеня було визначено емпірично. Площа SUSAN визначається формулою

$$n(M) = \sum_{m \in M} C(M) \quad (1.2)$$

З появою FAST перевага стала надаватися методиці виявлення кутів, а не методиці виявлення меж, оскільки стверджувалося, що кути є одним з найбільш інтуїтивних типів ознак, які показують сильну зміну інтенсивності у двох вимірах, і, отже, добре відрізняються від сусідніх точок. Згідно порівняльному дослідженню існуючих кутових детекторів, заснованих на вищевказаних критеріях (узгодженість, точність та швидкість), було виявлено, що більшість із цих детекторів задовольняють одному з критеріїв, але не працюють у інших [19].

1.2.3 SIFT-детектор

SIFT (масштабно-інваріантна трансформація ознак) [20,21] складається із чотирьох основних етапів:

- виявлення екстремумів масштабного простору;
- локалізація ключових точок;
- призначення орієнтації;
- дескриптор ключової точки.

На першому етапі використовується функція різниці гауссіанів (DoG) виявлення потенційних ключових точок, які виявилися інваріантні до масштабу та орієнтації. DoG був використаний замість гауссіана для покращення швидкості обчислень.

$$D(x, y, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) = L(x, y, k\sigma) - L(x, y, \sigma), \quad (1.3)$$

де * – оператор згортки, $G(x, y, \sigma)$ – гауссіан змінного масштабу, $I(x, y)$ – вхідне зображення, $D(x, y, \sigma)$ – різниця гауссіанів у масштабі k . На етапі локалізації ключових точок для кожного кандидата ключової точки проводиться інтерполяція ближніх даних для точного визначення положення, потім відкидаються ключові точки низького розмаїття, і оскільки функція

різниці гауссіанів буде мати сильні значення вздовж ребер, навіть якщо кандидат у ключові точки не стійкий до невеликого шуму, то для збільшення стабільності варто виключати ключові точки, які мають певне місце, але мають великий внесок від ребер.

Матриця Г'юссе використовується для обчислення головних кривизн та виключення ключових точок, у яких головна кривизна поперек ребра буде багато більше головної кривизни вздовж нього. На етапі призначення орієнтації кожній ключовій точці призначається одна або кілька орієнтацій, ґрунтуючись на напрямки градієнтів у локальному зображенні. Це ключовий крок досягнення інваріантності щодо обертання, оскільки дескриптор ключовий точки може бути представлений щодо цієї орієнтації, а тому стає інваріантним щодо обертання зображення.

Гістограма орієнтації була сформована з орієнтацій градієнта точок зразка в області навколо ключової точки, щоб отримати призначення орієнтації. На останньому кроці створюється набір гістограм напрямків на 4×4 сусідніх пікселях з 8 областями в кожній, оскільки $4 \times 4 = 16$ гістограм з 8 областями у кожній, фінальний вектор (дескриптор SIFT) має 128 елементів. Потім вектор нормалізується до одиничної довжини з метою забезпечення інваріантності афінних змін у висвітленні. Для скорочення ефекту нелінійного освітлення, застосовується поріг завбільшки 0,2 (значення було підібрано емпірично) та вектор знову нормалізується. Процес застосування порога може покращити результати зіставлення, навіть якщо ефекти нелінійного освітлення відсутні.

1.2.4 PCA-SIFT детектор

PCA-SIFT (метод головних компонентів SIFT) – це стандартна методика і новий алгоритм, він з'явився як спроба покращити SIFT, зменшивши розмірність та усунувши обчислювальні витрати, пов'язані з імплементацією за Лоу. Він добре підходить для представлення області

ключових точок і дозволяє лінійно проектувати зразки високої розмірності в низькорозмірний простір ознак. Іншими словами, PCA-SIFT використовує PCA замість згладжених зважених гістограм для нормалізації градієнтної ділянки. Вектор ознак, в результаті, виявляється значно меншої розмірності, ніж стандартний вектор ознак SIFT, і його можна використовувати з тими самими алгоритмами зіставлення. PCA-SIFT, як і SIFT, також використовує евклідову відстань, щоб визначити, чи відповідають два вектори одній і тій же ключовій точці на різних зображеннях. У PCA-SIFT вхідний вектор створюється шляхом об'єднання відповідей горизонтального та вертикального градієнта для ділянки 41×41 з центром у ключовій точці, який має $2 \times 39 \times 39 = 3042$ елементів. Згідно PCA-SIFT, менша кількість компонентів потребує меншого обсягу пам'яті та призводить до більш швидкого зіставлення. PCA-SIFT вдалося на порядок прискорити процес зіставлення SIFT, але виявилось, що його здатність до розрізнення слабкіша, ніж у SIFT.

1.2.5 SURF-детектор

Детектор стійких прискорених ознак (SURF) був задуманий для забезпечення високої швидкості у трьох етапах виявлення ознак: виявлення, опис та зіставлення. На відміну від PCA-SIFT, SURF прискорює процес виявлення SIFT, не знижуючи якість виявлених ключових точок. Алгоритми SIFT та SURF використовують дещо різні способи виявлення функцій: SIFT будує піраміди зображень, фільтруючи кожен шар по гауссіям, за зростаючими значеннями сигми і витягуючи різницю; з іншого боку, SURF створює "стек" без знижувальної дискретизації 2:1 для більш високих рівнів у піраміді, що призводить до зображень з однаковим дозволом.

На етапі зіставлення ключових точок найближчий сусід визначається як ключова точка з мінімальною евклідовою відстанню для інваріантного вектора дескриптора. Лоу використовував більш ефективний вимір,

отриманий шляхом порівняння відстані найближчого сусіда з цим другим найближчим сусідом.

1.2.6 F-SIFT детектор

F-SIFT (Fast-SIFT) складається з тих же чотирьох основних етапів SIFT, але для виявлення ознак використовується алгоритм виявлення кутів FAST. Алгоритм також відрізняється офлайновою фазою підготовки, де ознаки створюються на різних рівнях масштабу, та онлайнною фазою, де ознаки створюються лише для фіксованого рівня масштабу зображення з мобільного пристрою. Крім того, ознаки створюються тільки з фіксованих областей розміром 15×15 пікселів та створюється лише 36-мірний SIFT дескриптор.

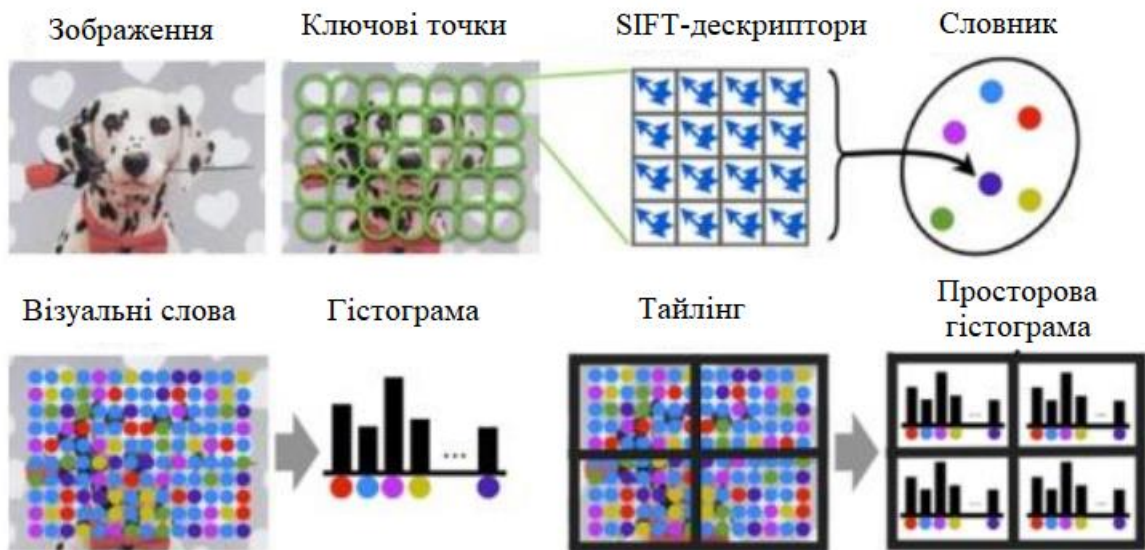


Рисунок 1.2 – Послідовність кроків Fast-SIFT

Підхід був надалі розширений шляхом інтеграції з деревом слів, що масштабується (англ. Scalable Vocabulary Tree). Це дозволяє ефективно розпізнавати велику кількість об'єктів мобільного телефону. Підхід обмежений головним чином кількістю доступної оперативної пам'яті. Таким чином, вектор ознак квантується у візуальні слова, при цьому вектор

виявляється значно меншої розмірності порівняно зі стандартним вектором ознак SIFT. Частота кожного візуального слова записується в гістограму для кожного тайла просторового тайлінгу, як показано на рисунку 1.2. Останній вектор ознак для зображення є об'єднання цих гістограм.

1.3 Методика видалення зображень, заснована на нейромережах

Наведені в попередніх пунктах методики отримання властивостей з зображень мали широке поширення до того моменту, як не збіглося кілька важливих факторів:

- системи, що містять модулі вилучення ознак на базі кольірних та текстурних методик, мали великі похибки при оцінці подібності зображень;
- у таких розділах науки як машинне навчання та обробка сигналів намітилися серйозні позитивні зрушення;
- продуктивність графічних процесорів (GPU) значно зросла;
- вартість комп'ютерної апаратної частини суттєво знизилася.

Як було сказано раніше, методики, побудовані на аналізі кольірних низькорівневих характеристик і текстур, що не можуть виділити поняття з масиву даних, обидві методики лише дозволяють коректніше описати низькорівневі властивості зображення для того, щоб аналіз подібності зображень виявився точнішим. Однак навіть незважаючи на це, точність систем, побудованих на таких методиках, залишається недостатньо високою.

Це відбувається через те, що між виділенням властивостей об'єкта людиною і методиками існує семантичний розрив, який пояснюється тим, що людина при розпізнаванні об'єкта оперує абстрактними високорівневими характеристиками, як і методики, побудовані на обробці кольору та текстури, нездатні виділити. Проблема семантичного розриву між машинним та людським сприйняттям відома науковій спільноті вже давно, передбачалося, що вона буде усунена, якщо машина думати як і людина. Тобто, коли стане можливим збудувати штучну нейронну мережу, що дозволяє у досить

хорошому наближенню імітувати діяльність біологічної нейронної мережі. Як відомо, мозок людини містить мільярди нейронів, взаємопов'язаних нервових клітин, чиєю функцією є обробка та передача хімічних та електричних сигналів, з'єднання нейронів утворюють багаторівневу мережеву структуру.

Поява згорткових нейронних мереж була мотивована біологічними візуальними системами. Головною відмінністю згорткового шару (англ. convolutional layer) від повнозв'язкового є те, що в ньому до кожного фрагмента вхідних даних застосовується одна і та ж згортка (поелементне множення згортки з фрагментом та підсумовування отриманих доданків, пакунок у літературі також часто називають фільтром), як правило, згортка являє собою вікно певної розмірності, що застосовується до фрагмента вхідних даних, а потім зсувається до наступних фрагментів, результат застосування згортки записується у відповідну позицію вихідних даних. приклад описаної операції можна побачити на рисунку 1.3.

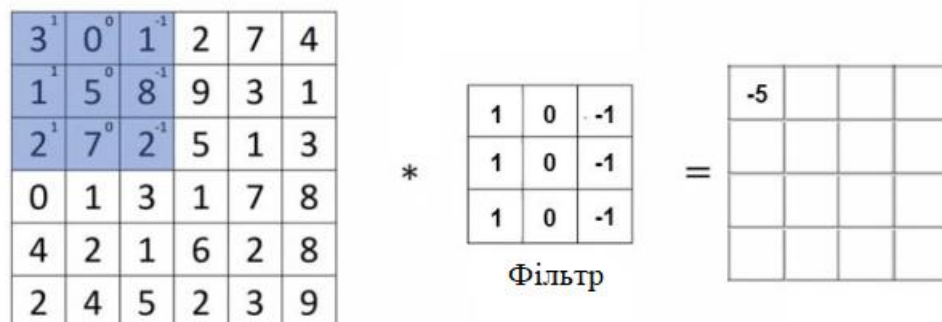


Рисунок 1.3 – Використання фільтра вертикальний детектор до вхідних даних

В архітектуру згорткової мережі закладається знання з комп'ютерного зору: кожен піксель зображення сильніше пов'язаний із сусідніми пікселями в порівнянні з віддаленими, а об'єкт на зображенні може знаходитись у будь-якому місці. Крім цього, необхідно виділити ще два аспекти, що роблять CNN справді потужним інструментом: розподіл параметрів та розрідженість

з'єднань. Розподіл параметрів має на увазі, що фільтр (наприклад, детектор вертикальних кордонів), що виявився корисним для однієї частини зображення цілком можливо може бути корисним і для іншого його частини. Розрідженість з'єднань передбачає, що вихідні значення кожного шару залежить тільки від невеликої кількості вхідних значень. Ці аспекти дозволяють згортковій нейромережі витягувати властивості зображення інваріантні до переносів, відображень, розмиття та інших операцій.

У разі одновимірних вхідних даних результат застосування згорткового шару l з фільтром розміру k позиції i обчислюється наступним чином:

$$x_i^l = g\left(\sum_{j=1}^k w_j^l x_{i+j}^{l-1}\right). \quad (1.4)$$

Крім згорткових шарів у CNN використовуються субдискретизуючі шари (англ. subsample layer або pooling layer) зменшення розмірності. У випадку з max- або average-pooling кожен фрагмент вхідних для шару даних замінюється на відповідно максимальне та середнє значення розглянутого фрагмента. У global-pooling описана операція виробляється задля околиці, а одночасно всіх вхідних даних. Для зменшення ефекту перенавчання (надмірне підстроювання алгоритму під навчальну вибірку) нейронної мережі використовується dropout-регуляризація (dropout layer).

Ідея полягає в наступному: на кожному кроці навчання кожен нейрон виключається з мережі з деякою ймовірністю p . Це дозволяє нейронам налаштуватися незалежно один від одного. Згорткова нейронна мережа зазвичай являє собою чергування згорткових, субдискретизуючих та повнозв'язних шарів. Після кожного шару може бути використаний dropout шар. Навчання нейронної мережі зазвичай проводиться градієнтним спуском за допомогою методу зворотного розповсюдження помилки (англ. backpropagation) для обчислення градієнта вагів. У як функцію активації можуть використовуватися:

$$\text{ReLU}(x) = \max(x, 0), \sigma(x) = \frac{1}{1 + \exp^{-x}}, \tanh(x) = \frac{\exp^x - \exp^{-x}}{\exp^x + \exp^{-x}}. \quad (1.5)$$

На рисунку 1.4 можна побачити приклад згорткової нейронної мережі для розпізнавання зображень.

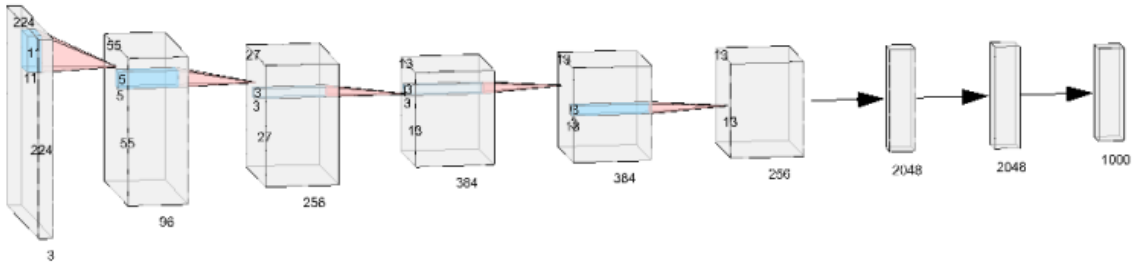


Рисунок 1.4 – Приклад згорткової нейронної мережі

Розглянемо структуру мережі як набір шарів від входу до виходу та активації, отримані з k -го прихованого шару. Існує гіпотеза, що активації мають дедалі більшу дискримінативну здатність зі збільшенням k (рисунок 1.5).



Рисунок 1.5 – Результати застосування функцій активацій до прихованих шарів порядок зростання

Численні дослідження показують, що активація шарів нейронної мережі є хорошими ознаками для інших алгоритмів машинного навчання. Крім цього, використання крос-ентропійної функції втрат разом із softmax активаціями останнього шару рівноцінно використанню логістичної регресії над активаціями, отриманими з останнього прихованого шару нейромережі

Таким чином, якщо нейронна мережа показує хорошу точність, можна вважати, що активація останнього шару є хорошою дискримінативною ознакою об'єктів, а нейронну мережу можна використовувати як перетворювач даних. Незважаючи на те, що нейронні мережі здатні виділяти ознаки зображень краще, ніж стандартні методики, засновані на кольорі та текстурах, вони мають і свої очевидні мінуси. Так, наприклад, SIFT набагато простіше спроектувати і він містить меншу кількість параметрів по порівняно з CNN, SIFT вимагає менших обчислювальних потужностей та пам'яті обробки характеристик зображень. Однак, для задачі побудови системи пошуку зображень точність пошуку є кращою, а частина обчислювальної складності, пов'язаної з побудовою CNN, можна усунути, якщо скористатися вже готовими навченими CNN як, наприклад, AlexNet.

1.4 Кластеризація зображень методом k-середніх

Маємо масив спостережень (об'єктів), кожен з яких має певні значення по ряду ознак. Відповідно до цих значень об'єкт розташовується у багатовимірному просторі. Дослідник визначає кількість кластерів, що необхідно утворити. Випадковим чином обирається k спостережень, які на цьому кроці вважаються центрами кластерів.

Кожне спостереження «приписується» до одного з n кластерів – того, відстань до якого найкоротша. Розраховується новий центр кожного кластера як елемент, ознаки якого розраховуються як середнє арифметичне ознак об'єктів, що входять у цей кластер.

Відбувається така кількість ітерацій (повторюються кроки 3-4), поки кластерні центри стануть стійкими (тобто при кожній ітерації в кожному кластері опинятимуться одні й ті самі об'єкти), дисперсія всередині кластера буде мінімізована, а між кластерами – максимізована. Вибір кількості кластерів відбувається на основі дослідницької гіпотези. Якщо її немає, то

рекомендують створити 2 кластери, далі 3,4,5, порівнюючи отримані результати.

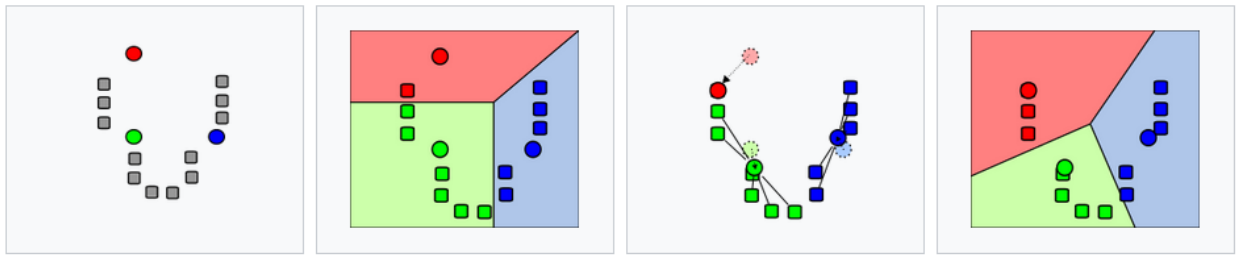


Рисунок 1.6 – Демонстрація алгоритму

Принцип алгоритму полягає в пошуку таких центрів кластерів та наборів елементів кожного кластера при наявності деякої функції $\Phi(\cdot)$, що виражає якість поточного розбиття множини на k кластерів, коли сумарне квадратичне відхилення елементів кластерів від центрів цих кластерів буде найменшим:

$$V = \sum_{i=1}^k \sum_{x_j \in S_i} (x_j - \mu_i)^2 \quad (1.6)$$

Приклади розбиття по кластерам представлені на рисунку 1.7.

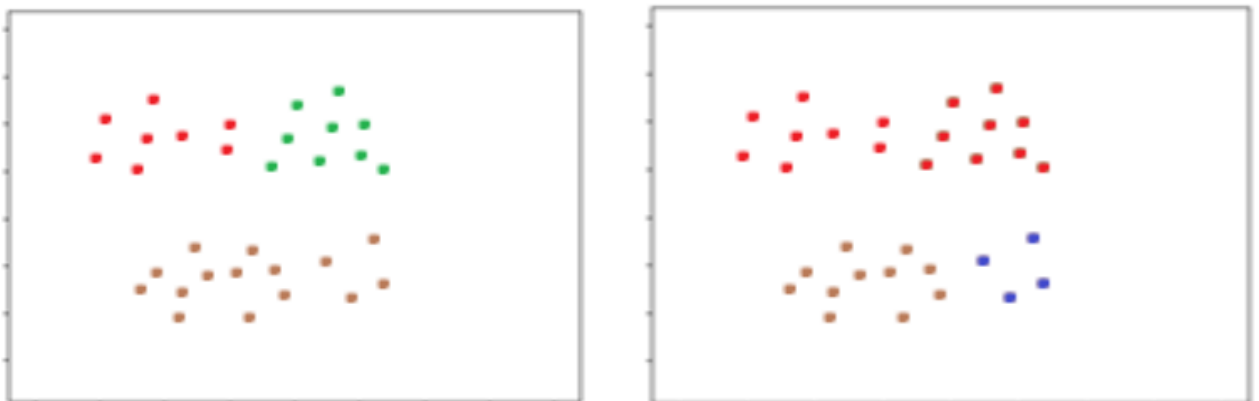


Рисунок 1.7 – Приклади розбиття по кластерам

Зрозуміло, що при виконанні алгоритму К-середніх кількість кластерів K , має бути менше кількості елементів у навчальній вибірці M . Простим і очевидним способом зробити випадкову ініціалізацію є вибір будь-яких k примірників з навчальної вибірки та припущення, що вони є центрами мас. Залежно від вибору навчальної вибірки алгоритм k -середніх може в результаті сходиться до різних рішень. Це може бути ідеальне розбиття за кластерами, як на рисунку 1.7 ліворуч. Але у разі менш вдалої випадкової ініціалізації алгоритм може зійтися до локального мінімуму (рисунок 1.7 праворуч).

2 МОДИФІКОВАНІ АЛГОРИТМИ КЛАСТЕРИЗАЦІЇ МЕТОДОМ К-СЕРЕДНІХ

Кластеризація методом k-середніх є одним з найпростіших та досить ефективних способів кластеризації об'єктів заданого множини. Нехай у даних можна виділити K кластерів, i є навчальна вибірка зображень $\{x^{(1)}, x^{(2)}, \dots, x^{(m)}\}$, $x^{(i)} \in \mathbb{R}^n$, яка є простором характеристик зображень із вибірки. Алгоритм кластеризації K-середніх реалізується в такий спосіб.

2.1 Алгоритм кластеризації з множинною випадковою ініціалізацією

Для запобігання завершенню алгоритму в локальному мінімумі пропонується множинна випадкова ініціалізація (лістинг 2.1).

Лістинг 2.1 – Алгоритм кластеризації з множинною випадковою ініціалізацією

```
for i = 1 to 50 {
випадкова ініціалізація методом k-середніх;
виконати k-means. Отримати  $c^{(1)}, \dots, c^{(m)}, \mu_1, \dots, \mu_k$ ;
обчислити функцію емпіричного ризику  $J(c^{(1)}, \dots, c^{(m)}, \mu_1, \dots, \mu_k)$ ;
}
```

Проблемою цього підходу є те, що, за оцінками дослідників, він добре працює в тому випадку, якщо $K < 10$, у випадку $K \gg 10$ коректність результатів буде трохи вище, але ефективність алгоритму сильно постраждає через обчислювальні витрати.

Алгоритм загального призначення k-середніх має серйозний недолік: він чутливий до ініціалізації центрів мас або середніх точок. Це означає, що й центр мас ініціалізується як “далека” точка, тобто, може статися так, що поряд з ним не буде жодних точок, і в той же час більше одного кластера

може бути пов'язаними з одним і тим же центром мас. Одночасно більше одного центру мас можна ініціалізувати в один і той же кластер, що призводить до поганої кластеризації.

2.2 Кластеризація методом k-середніх ++

Інтуїція, що лежить в основі підходу k-середніх, полягає в тому, що розподіл початкових центрів кластерів є гарною ідеєю: перший центр мас вибирається випадковим чином із кластеризованих точок, після чого кожен наступний кластерний центр мас вибирається з точок даних, що залишилися, з ймовірністю, пропорційною його квадрату відстані від найближчого існуючого кластерного центру маси точки. Нижче наведено опис алгоритму (лістинг 2.2).

Лістинг 2.2 – Алгоритм кластеризації методом k-середніх ++

Вибрати довільним чином один центр мас із навчальної вибірки.

Для кожного елемента x , обчислити $D(x)$, відстань між x та найближчим центром мас, який вже було обрано.

Вибрати один центр мас випадковим чином з елементів навчальної вибірки, використовуючи зважений розподіл ймовірностей, де елемент вибирається з ймовірністю, пропорційною $(D(x))^2$.

Повторювати кроки 2 і 3 до k центрів мас не відібрані.

Продовжити виконання кроків алгоритму методу K-середніх.

Необхідно відзначити, що ефективність алгоритму кластеризації залежить від обраного значення k . Тому замість використання одного зумовленого значення k , часто розглядається певний набір потенційних значень, кожне з яких має бути достатньо більшим щоб відобразити характеристики елементів, що кластеризуються, але також і досить малим, щоб бути значно менше загальної кількості елементів.

Опубліковані дослідження щодо кластеризації k-середніх та його модифікацій зазвичай не містять будь-яких пояснень або обґрунтувань

вибору конкретних значень для k . З аналізу цих досліджень можна зробити висновок, що, по-перше, ряд дослідників використовували тільки одне або два значення k , по-друге, в той час як інша група дослідників використовувала відносно великі значення k порівняно з кількістю елементів.

У деяких практичних програмних реалізаціях методу K -середніх значення k вводиться користувачем. Для того, щоб домогтися прийняттого результату зазвичай доводиться запускати алгоритм k -середніх різними значеннями K , адекватність результатів у таких системах визначається “на око” користувачем, такого роду підхід не підходить для задач, де безліч елементів, які потребують кластеризації, величезна. При цьому способі кількість кластерів вважається рівною кількості класів у даних. У такому разі алгоритм кластеризації може бути використаний як класифікатор, що застосовується до даних, для яких немає мітки класу та одержуючої для них оцінки приналежності до класів. Цей метод покладається на умову, що кожен кластер містить лише об'єкти, що належать до одного класу.

На жаль, здебільшого для реальних завдань ця умова не виконується. Для визначення значення k можливе використання міри найближчих сусідів, яка могла б бути доповненням до функції емпіричного ризику. Однак незважаючи на те, що така методика добре себе показала для деяких наборів даних, її ефективність у практичних застосуваннях ще тільки повинна бути доведена. Оскільки у функцію емпіричного ризику повинні вноситись зміни, ця методика не може бути застосована до оригінального алгоритму кластеризації k -середніх.

Функція оцінки $f(K)$, яка отримує результати кластеризації, може бути також використана для вибору кількості кластерів. Розглянемо основні фактори, які ця функція повинна враховувати при отриманні результатів. Функція має бути тісно пов'язана з критерієм кластеризації. Як згадувалося раніше, таке ставлення може запобігти несприятливому впливу на процес валідації. Зокрема, в алгоритмі K -середніх критерієм є функція мінімізації емпіричного ризику, тому функція оцінки має враховувати цей параметр.

Спостерігач, який вивчив лише низькорівневі деталі об'єкта, може отримати лише загальне уявлення про об'єкт. Після збільшення рівня деталізації спостерігач зможе отримати більше інформації про об'єкт, але, в той же час, обсяг даних, який має бути оброблений також значно збільшиться.

Через обмеженість ресурсів високий рівень деталізації зазвичай використовується тільки для того, щоб вивчити найбільш важливі частини об'єкта. Такий підхід може бути застосований у кластеризації. Набір даних з n об'єктами може бути згрупований в будь-яку кількість кластерів від 1 до n , відповідних найнижчим і найвищим рівням деталізації. Визначаючи різні значення K , можна оцінити результати угруповання об'єктів у різну кількість кластерів. В результаті такої оцінки кілька значень можуть бути рекомендовано користувачам, але остаточний вибір залишається за ними. Кластеризація використовується для пошуку невідповідностей у розподілі даних та ідентифікації областей, у яких зосереджені об'єкти. Однак не всяка область з високою концентрацією об'єктів вважається кластером.

Для того, щоб область була ідентифікована як кластер, важливо проаналізувати не лише її внутрішній розподіл, а й також її взаємозалежність коїться з іншими групами об'єктів у наборі даних. У кластеризації методом K -середніх функція мінімізації емпіричного ризику – це усереднена сума квадратів відстаней об'єктів до центрів мас, можна запровадити поняття деформації кластера так:

$$Distortion_j(c^{(j)}, \mu_j) = \frac{1}{|c^{(j)}|} \sum_{i=1}^{|c^{(j)}|} \|x^{(i)} - \mu_c(j)\|^2 \quad (1.7)$$

Тоді, обчислюючи показники деформації кожного кластера, можна буде визначити його вплив на весь обсяг даних з урахуванням деформацій усіх кластерів:

$$S_k = \sum_{i=1}^K Distortion_i, \quad (1.8)$$

де K – кількість кластерів.

Така інформація є важливою для розуміння того, чи може конкретна область у просторі об'єктів вважатиметься окремим кластером чи ні. Стійкість $f(K)$ дуже важлива. Так як ця функція заснована на результаті кластеризації, виконаної алгоритмом, важливо, щоб значення результату не змінювалися або змінювалися незначно при різних запусках алгоритму K -середніх і незмінному K , незважаючи на залежність методу k -середніх від випадкової ініціалізації. Роль $f(K)$ полягає у виявленні закономірностей у розподілі даних, тому важливо, щоб $f(K)$ не залежало від кількості елементів. Число кластерів, K , має бути набагато менше кількості об'єктів N .

При збільшенні K величина $f(K)$ повинна сходиться до деякого постійного значення. Тоді, якщо для будь-якого проміжного K , величина $f(K)$ проявляє особливу поведінку (наприклад, досягає мінімуму або максимуму), то таке значення K може бути прийняте як значення бажаної кількості кластерів. Таким чином, у цьому розділі визначено основні напрямки модифікації методу K -середніх з метою його адаптації для обробки зображень із баз, у яких зберігається їх велика кількість.

2.3 Аналіз архітектур нейронних мереж

2.3.1 Глибокі нейронні мережі

Обчислення в нейронній мережі зазвичай виробляються у два етапи: на першому виконується пряме поширення (англ. *forward propagation step*) помилки, другий етап – зворотне поширення помилки (англ. *backward propagation step*). Логістична регресія є алгоритмом для розв'язання задачі бінарної класифікації.

Єдиною відмінністю глибокої нейронної мережі від неглибокої є кількість прихованих шарів, іншими словами, глибока нейронна мережа – це нейронна мережа з більш ніж одним прихованим шаром.

Розглянемо приклад глибокої нейронної мережі із 3 прихованими шарами (рисунок 2.1).

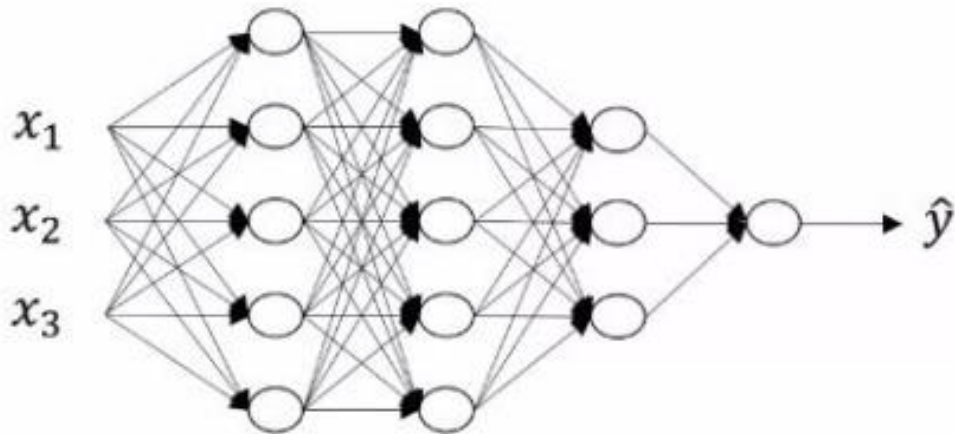


Рисунок 2.1 – Глибока чотирьохшарова нейронна мережа

Як видно із рисунку 2.1, крім відмінностей у кількості прихованих шарів з неглибокою нейронною мережею, є відмінність і в кількості нейронів у прихованих шарах. Для подання довільної глибокої нейронної мережі необхідно отримати нове уявлення в якому було б відображено розмірність параметрів щодо кількості нейронів у шарах.

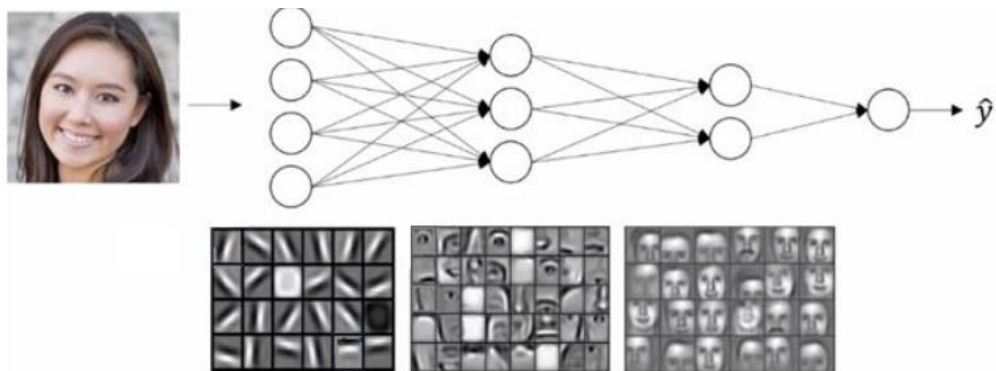


Рисунок 2.2 – Розпізнавання облич з використанням глибокої нейронної мережі

Нейрони кожного із шарів глибокої нейронної мережі здатні визначати характеристики об'єкта, що вивчається, так, наприклад, в задачі розпізнавання осіб ранні версії мережі обчислюють прості функції, наприклад, характеристики кордонів, потім, на пізніших шарах, ці характеристики зіставляються одна з одною і стає можливим визначення таких об'єктів як очі, брови, і, нарешті, на останніх шарах вирішується проблема розпізнавання осіб (рисунок 2.2).

Таким чином, у нейронній мережі з кожним новим шаром витягуються більш високорівневі властивості об'єктів, які, зрештою, допомагають вирішити конкретне завдання.

Однак може виникнути питання, чому для вирішення задачі розпізнавання образів краще використовувати архітектуру глибокої нейронної мережі, а не, наприклад, неглибокої мережі з одним прихованим шаром, але збільшеним числом нейронів. Виявляється, що незважаючи на те, що використання такої неглибокої мережі можливе, кількість нейронів в прихованому шарі такої мережі експоненційно зростатиме. Отже, були розглянуті типові приклади нейронних архітектур мереж. Досі найбільша увага в математичних моделях мереж відводилося на обчислення оптимальних значень параметрів мережі W та b , однак, виявляється, що крім параметрів нейронної мережі, існують окрема група параметрів – гіперпараметри, які безпосередньо впливають отримання фінальних значень параметрів нейромереж.

2.3.2 Згорткові нейронні мережі

Раніше розглянута класична архітектура нейронних мереж добре підходить для вирішення завдань класифікації та кластеризації у разі щодо невеликої кількості вхідних даних – векторів-ознак для аналізу (тобто $X = \{x_1, \dots, x_{n_x}\}$). Однак на прикладі зображень можна побачити, що потенційних вхідних даних для нейромережі може виявитися надзвичайно багато, якщо

розміри зображення досить великі, наприклад, становлять 1000×1000 пікселів у колірному просторі RGB, то кількість ознак, витягнутих для кожного з пікселів або групи пікселів може скласти мільйони.

Алгоритми навчання, побудовані на базі такої нейронної мережі з такою великою кількістю параметрів, які не зможуть працювати досить швидко, тому на практиці в задачі комп'ютерного зору, якому належить і розпізнавання образів, тобто, у тому числі, різних товарів, використовується згортова архітектура нейронної мережі.

Одна із очевидних переваг такої мережі – це звуження кількості вхідних даних для навчання нейронної мережі за рахунок застосування методик відокремлення найбільш важливих та суттєвих ознак із зображення на основі операцій згортки.

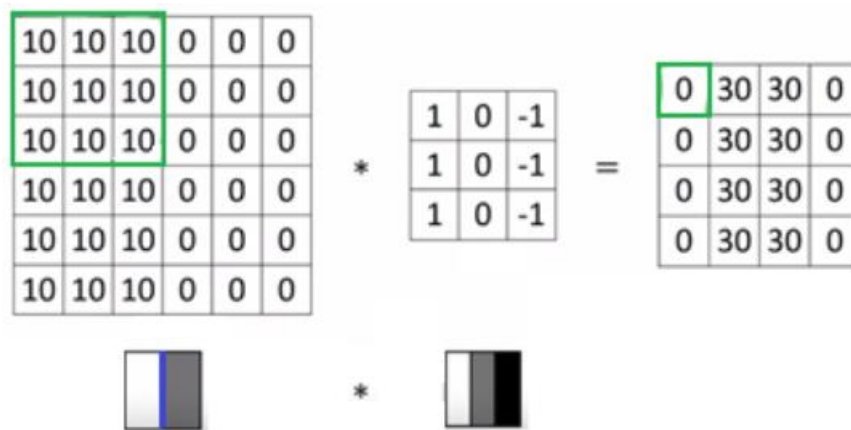


Рисунок 2.3 – Застосування операції згортки для отримання вертикальних кордонів

Розглянемо основні ідеї згоргової архітектури. Наприклад, з допомогою операції згортки, що виконується всередині нейромережі, можна виділяти нові ознаки з цілих областей зображення, наприклад, можна виявляти межі об'єктів (рисунок 2.3). Припустимо, що є зображення з розміром $6 \times 6 \times 3$ і є два фільтри, кожен з яких відповідає визначенню вертикальних або горизонтальних кордонів у зображенні, застосуємо

операцію згортки к зображенню та кожному з фільтрів – отримаємо на виході два зображення 4×4 , кожне з яких відповідає вертикальним та горизонтальним кордонів відповідно (рисунок 2.4).

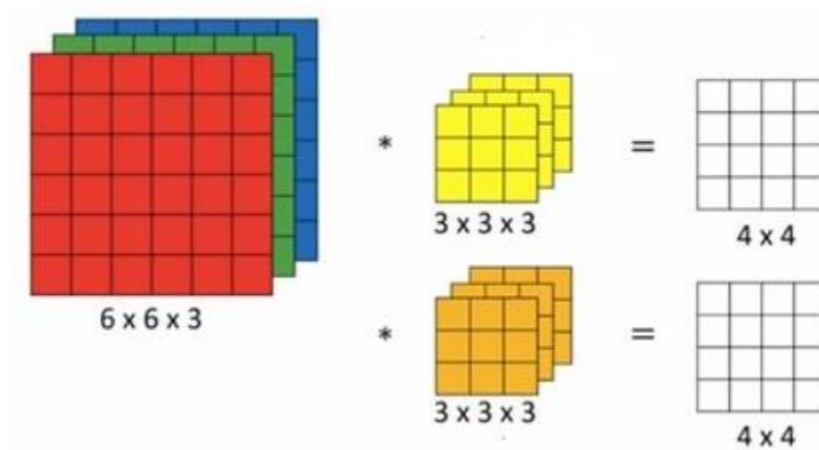


Рисунок 2.3 – Застосування операції згортки для отримання вертикальних та горизонтальних кордонів із зображення

Розмірність 4×4 не випадкова. Якщо n_H – висота зображення, n_W – ширина зображення, n_C – кількість каналів у зображенні, а f – ширина і висота фільтра, то підсумковий розмір зображення можна обчислити за формулі $n_H - f + 1 \times n_W - f + 1$. Дивлячись на рисунок 2.3, нескладно звернути увагу на те, що візуально фільтри та результуючі зображення можна розглядати як окремі шари і тоді застосування шару фільтрів к початковому зображенню $6 \times 6 \times 3$ призводить до появи нового шару з виділених властивостей зображення розмірністю $4 \times 4 \times 2$, як видно, значення n_H та n_W , зменшилися після застосування операції згортки, в цьому полягає один із побічних ефектів застосування згорткової операції – крім виділення нових властивостей із зображення, операція від шару до шару призводить до зменшення значень n_H та n_W , а значить і скорочення вхідних даних навчання. Значення n_C від шару до шару, взагалі кажучи, може зростати, але його зростання на порядки нижче за швидкість скорочення параметрів n_H та n_W .

Агрегування. Агрегування (англ. pooling) - операція, що зменшує розмірність досліджуваних властивостей. Агрегування застосовується для кожного каналу шару окремо. Як приклад, розглянемо агрегування по максимуму та агрегування за середнім. В обох операціях (рисунки 2.4 та 2.5) фільтр $f \times f$ застосовується до ділянки i потім зсувається по горизонталі та вертикалі.

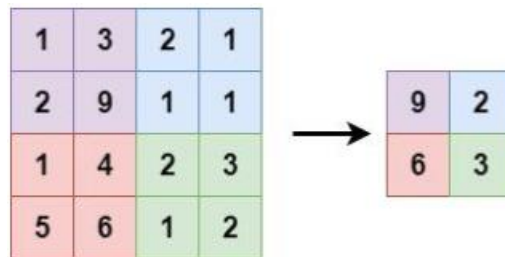


Рисунок 2.4 – Агрегування по максимуму

Відмінність полягає лише в тому, що в агрегуванні по максимуму "Вікно" вибирає максимальне значення з ділянки, а в агрегуванні по середньому – обчислює середнє.

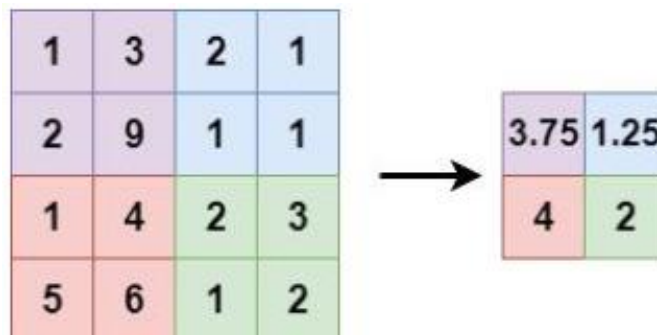


Рисунок 2.4 – Агрегування по середньому

Розглянемо архітектуру нейронної мережі з прикладу LENET-5 (рисунки 2.5). Як видно з рисунку, нейронна мережа включає 2 згорткових шари, 2 агрегують шари та звичайну нейронну мережу з 2 шарів.

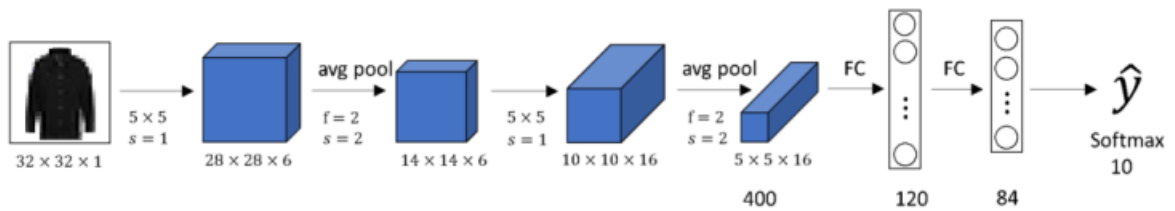


Рисунок 2.5 – Архітектура нейронної мережі LENET-5

Проаналізуємо структуру. На вхід подається зображення типу grayscale з 1 каналом. Розмірність кожного наступного шару l визначається за формулі

$$\frac{n_H^{[l-1]} + 2p^{[l-1]} - f^{[l-1]}}{s^{[l-1]}} + 1 \times \frac{n_W^{[l-1]} + 2p^{[l-1]} - f^{[l-1]}}{s^{[l-1]}} + 1$$

в даному прикладі $p^{[l-1]}=s^{[l-1]}=0$. Неважко помітити, що при застосуванні фільтрів згортки для отримання нових властивостей (згорткових шарів) розмірність шару зростає, тому в архітектурі мережі слідом за отриманням згорткового шару слідує етап агрегування, що істотно скорочує розмірність отриманих властивостей. Після останньої операції агрегування всі отримані властивості з багатовимірного простору наводяться до векторного вигляду, отриманий вектор подається на стандартний вхід нейронної мережі (її ще називають повнозв'язковою), на виході до результуючих даних застосовується softmax функція для отримання мітки класу.

2.3.3 Налаштування гіперпараметрів

Гіперпараметри – це параметри, що керують значеннями параметрів W , b . Фактично саме гіперпараметри визначають кінцеві значення W та b . Прикладами гіперпараметрів є:

- швидкість навчання α ;
- кількість ітерацій градієнтного спуску;

- кількість прихованих шарів L ;
- кількість нейронів у прихованих шарах $n^{[1]}$, $n^{[2]}$, ... ;
- функції активації.

Вибір оптимальних гіперпараметрів – спіралеподібне завдання. Зазвичай спочатку беруться якісь перевірені чи випадкові значення гіперпараметрів, потім проводиться експеримент, обчислюється функція емпіричного ризику.

Потім процес повторюється знову і знову з інтуїтивно зміненими гіперпараметрами, доки не будуть отримані оптимальні результати. На навчальній вибірці модель навчається, при цьому контрольна вибірка використовується для того, щоб побачити, яка модель (алгоритм) працює краще на контрольній вибірці. Після певної кількості ітерацій ми можемо знайти найкращу модель, яку тепер необхідно випробувати на тестовій вибірці, щоб отримати об'єктивну оцінку (англ. unbiased estimate) того, наскільки алгоритм добре вирішує поставлене завдання.

Необхідно зауважити, що елементи контрольної та тестовий вибірок повинні належати одному розподілу, водночас час елементи навчальної вибірки можуть відрізнятися від елементів контрольної та тестової вибірок та належати іншому розподілу.

3 РОЗРОБКА ШТУЧНОЇ НЕЙРОННОЇ МЕРЕЖІ ДЛЯ КЛАСТЕРИЗАЦІЇ ЗОБРАЖЕНЬ

Незважаючи на те, що згорткові нейронні мережі (CNN) показали хороші результати у застосуваннях навчання з учителем до таких областей комп'ютерного зору, як локалізація об'єктів та категоризація, сучасні CNN не можуть впоратися із великомасштабним завданням кластеризації зображень, тому що зазвичай не існує достатньої кількості розмічених даних на навчання ознакам CNN.

Для ефективного вирішення проблеми вивчення репрезентативних ознак у нерозмічених вхідних зображеннях у задачі великомасштабної кластеризації зображень пропонується до використання кластеризуюча CNN для одночасного вирішення проблеми кластеризації та навчання ознак. Основна ідея запропонованого методу полягає в тому, що отримання кращих ознак вхідних зображень призводить до кращих результатів кластеризації. У свою чергу, найкраща кластеризація зображень також позитивно впливатиме на навчання ознакам. Щоб зменшити витрати на обчислення та пам'ять був задіяний мініпакетний метод k -середніх, він був включений до запропонованої структури кластеризуючого фреймворку на основі CNN.

3.1 Вибір архітектури мережі

Як відомо, AlexNet, ResNet, VGG є найпопулярнішими з існуючих моделей згорткових нейронних мереж для класифікації зображень. При цьому AlexNet до того ж є однією з найбільш ефективних нейромереж. В AlexNet як функції активації використовуються ReLU функції, а проблема перенавчання вирішується за рахунок використання dropout замість регуляризації. AlexNet навчається на безлічі зображень ImageNet, яке являє собою 15 мільйонів розмічених зображень з високою роздільною здатністю,

при цьому загальна кількість категорій зображень складає 22000. Архітектуру AlexNet наведено на рисунку 3.1

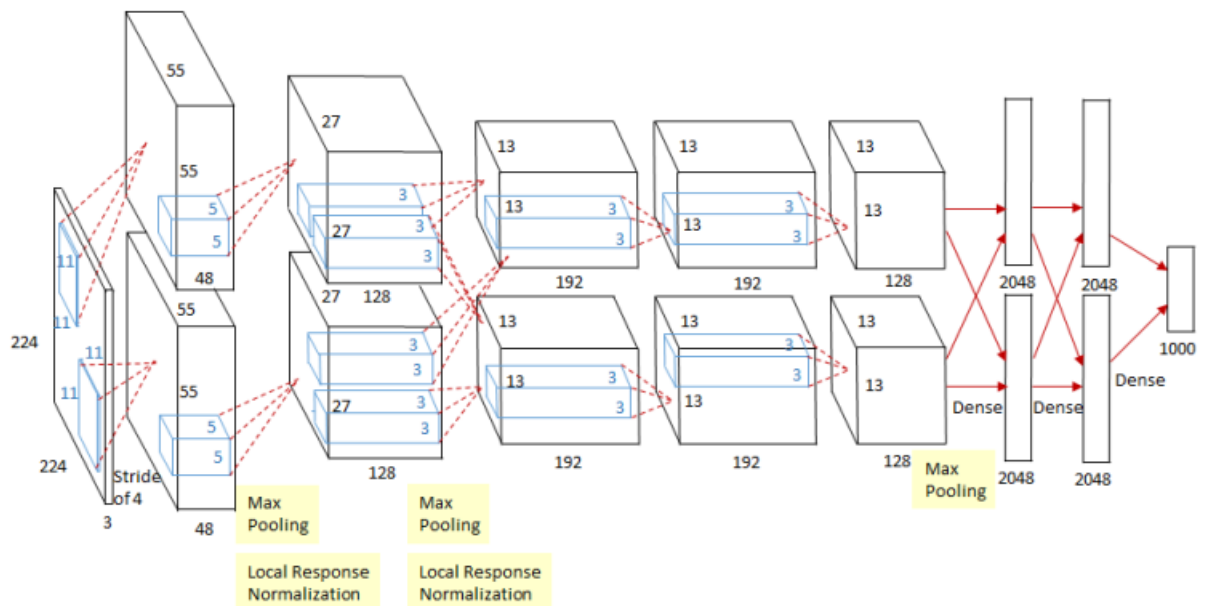


Рисунок 3.1 – Архітектура AlexNet

Як видно з рисунку 3.1, AlexNet містить вісім шарів, перші п'ять із яких – згорткові, а інші три – повнозв'язні. Вихідні дані пропускаються через функцію втрат softmax, яка формує розподіл зображень за 1000 мітками класів. Ядра другого, четвертого і п'ятого згорткових шарів пов'язані тільки з тими ядрами в попередньому шарі, які знаходяться на тому самому графічному процесорі. Нейрони в повнозв'язному шарі пов'язані один з одним.

Таким чином, AlexNet складається з 5 згорткових шарів та 3 повнозв'язних. ReLU застосовується після кожного шару. Мережа містить 62.3 мільйона параметрів. Згорткові шари виробляють 95% обчислень, у своїй нимі припадає лише 6% всіх параметрів.

Як видно із рисунку 3.2, запропонована кластеризуюча CNN запозичує 5 згорткових шарів з AlexNet і доповнює їх адаптаційними шарами (англ. adaptation layer), серед яких 3 згорткових шари Conv 6 – Conv 8 з кількістю

каналів 6144, 2048 і k відповідно повністю замінили собою повнозв'язковий шар AlexNet, фільтри розмірністю 3×3 та агрегуючий шар вибору глобального максимального елемента (англ. global max-pooling layer), який отримує максимальні значення з кожного каналу Conv 8 і перетворює їх на вектор $1 \times k$, де k – кількість кластерів.

У запропонованій варіації CNN найбільш знакові, що несуть основну інформацію, області зображення (англ. salient region) локалізуються Conv 8. Таким чином, запропонована кластеризуюча CNN витягує властивості безпосередньо із знакових областей зображення. Як показано на рисунку 3.3 для того, щоб вирішити проблему з високою обчислювальною складністю кластеризуючого алгоритму у великому обсязі даних, пропонується включити в нейромережу міні-пакетну кластеризацію методом К-середніх, покращену за рахунок використання методу опорних векторів.

Таким чином, завдання кластеризації та навчання ознак (властивостей) вирішуватимуться одночасно на кожній епісі оновлення параметрів міні-пакетним стохастичним градієнтом. На першому етапі необхідно визначитися зі значенням K , з цією метою був розроблено метод знаходження значення K , заснований на міні-пакетній кластеризації та обчисленні декількох функцій емпіричного ризику, залежить від K . Цей метод обчислює вектор можливих значень K , починаючи з першого потенційно коректного значення. Потім обране значення k з вектору використовується для виділення в кластері початкових K центрів мас. Далі витягуються ознаки (властивості) із зображень за допомогою CNN. Потім для кожного міні-пакета виконується міні-пакетний метод K середніх для присвоєння міток кластера конкретним зображенням, ґрунтуючись на витягнутих з нейромережі властивостях зображень, далі з за допомогою методу стохастичного градієнта оновлюються параметри кластеризуючої CNN.

У результаті нові властивості витягуються на основі оновлених параметрів мережі і потім використовуються для рекластеризації зображень.

Процес повторюється доти, доки результат кластеризації не зійдеться до стабільного стану. Якщо під час експлуатації системи з'ясувалося, що значення K було підбрано недостатньо точно, тобто кількість кластерів насправді більша – то цикл повторюється знову, починаючи з наступного значення K вектора.

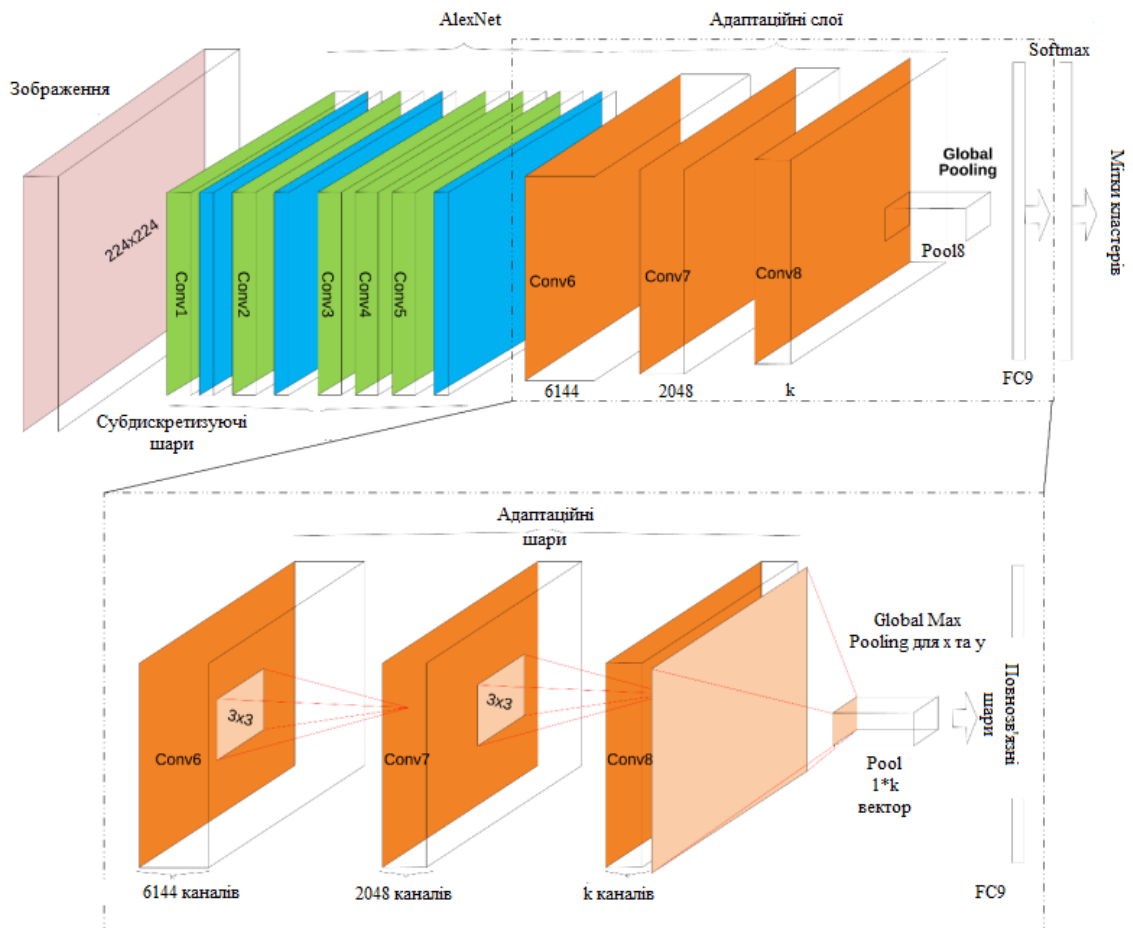


Рисунок 3.2 – Запропонована архітектура ЗНМ

3.2 Навчання ознакам та кластеризація

Для створення ефективної системи пошуку зображень на базі кластеризуючої CNN необхідно на початковому етапі визначитися з тим, яке значення кількості кластерів буде використано для налаштування неймережі.

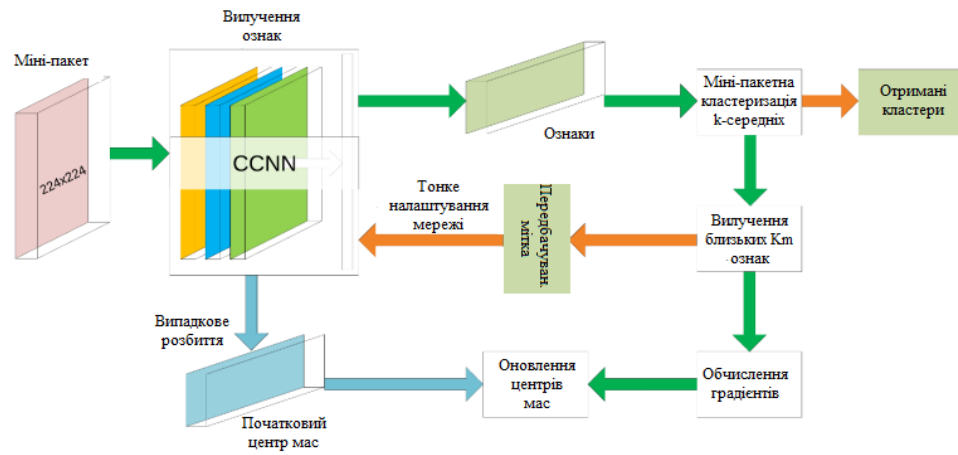


Рисунок 3.3 – Блок-діаграма кластеризуючої CNN для кластеризації та навчання представленням

Це дуже важливо, тому що кількість компонентів всередині softmax шару відповідає кількості кластерів, і якщо була зроблена помилка в оцінці кількості кластерів, то надалі неминуче доведеться зробити перенавчання нейромережі, а це додаткові витрати часу та ресурсів. Тому було запропоновано до використання новий підхід для знаходження початкового значення кількості кластерів K . Нижче наведено опис підходу. На першому етапі необхідно провести багаторазове виконання мініпакетної кластеризації методом K -середніх для кількох значень K з діапазону можливих значень [15, 30, 45,...], щоб зібрати значення функції емпіричного ризику щодо різних значень K . Процедура K meansMiniBatch, крім розбиття елементів за кластерами, що фактично виконує мінімізацію функції емпіричного ризику певного значення K . З результату виконання процедури, вектора центрів мас, нескладно отримати формулу функції емпіричного ризику:

$$J(C_1, \dots, C_k, \mu_{C_1}, \dots, \mu_{C_k}) = \frac{1}{b} \sum_{j=1}^b \|I_j - \mu_{C_j}\|^2 \quad (3.1)$$

Для кожного із значень K збережемо у векторі J_n значення емпіричної функції. З певним ступенем везіння ми можемо отримати так званий

ліктювий випадок (рисунок 3.4), це означає, що було отримано дуже точне наближення для K_{init} (на рисунку $K=3$). Але найчастіше буває інший сценарій, що можна побачити на рисунку 3.5. У цьому випадку важко зрозуміти яке значення вибрати для K_{init} . У запропонованому алгоритмі рекомендується брати значення K_{init} , що відповідає формулі:

$$K_{init} = \min_k \left(\frac{J_{k+1} - J_k}{J_{init}} < 0.1 \right) \quad (3.2)$$

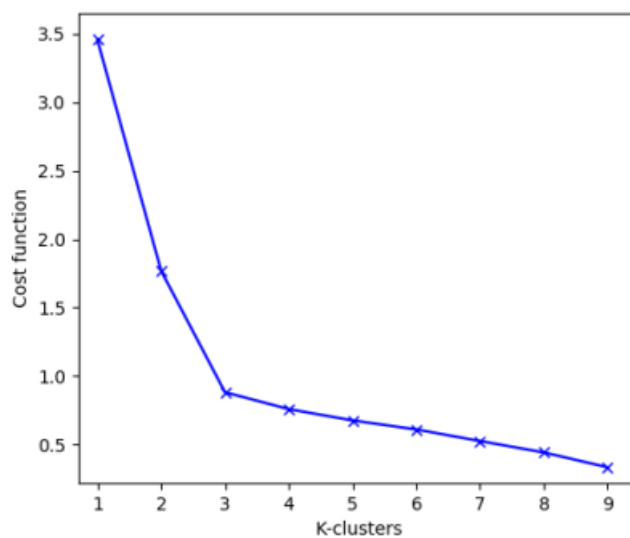


Рисунок 3.4 – Знаходження значення k

Отримане значення K_{init} буде добрим наближенням для K , яке буде використовуватися в `KmeansMiniBatch`. Сформуємо вектор можливих значень K наступним чином:

$$Range_K = \{K_{init}, K_{init} + \text{крок}, \dots, K_{init} + n \times \text{крок}\} \quad (3.3)$$

саме цей вектор (емпіричним шляхом крок був підібраний рівним 20) буде використовуватися системою у разі, якщо початкове значення K виявиться недостатньо хорошим наближенням практично.

3.3 Проведення експериментів

Було вибрано 2 великі датасети зображень: ILSVRC12 в ImageNet і Places2 для оцінки ефективності кластеризації. ILSVRC12 складається з 1.2 мільйона зображень для навчання та 50000 зображень для валідації, зібраних з 1000 категорій, Places2 складається з 1.6 мільйона зображень для навчання та 18250 зображень для валідації, зібраних з 356 категорій. Тому що для швидкої збіжності параметрів запропонованої кластеризуючої згорткової нейронної мережі (CCNN) були передбачені на даних з навчальної вибірки ILSVRC12 (позначимо як ILSVRC-Training). Ми не будемо розраховувати ефективність методів кластеризації на ILSVRC12 збереження об'єктивності. Будемо оцінюватимемо ефективність кластеризації на навчальній вибірці Places2 (позначимо як Places-Training) та валідаційній вибірці (Places-Validation) і також на валідаційній вибірці ILSVRC12 (позначимо як ILSVRC-Validation).

Для навчальної та валідаційної вибірок Places2 число каналів у ILSVRC12 та кількість нейронів у Softmax шарі в запропонованій CCNN були виставлені значення 365, у той час як для валідаційної вибірки ILSVRC12 в як значення для кількості каналів у Conv 8 і числа нейронів у Softmax було вибрано 1000. Усі зображення були обрізані до центру до розміру 256×256 . Крім перевірки на великих датасетах, ефективність була розрахована також на відносно невеликому датасеті MNIST, що містить 60000 чорно-білих зображень розміром 28×28 . Так як розмір зображень у MNIST невеликий, вони не були обрізані.

Як було сказано раніше для оновлення параметрів мережі кластеризуючої CNN вибираються тільки перші k_m зображень з кожного міні-пакет. На рисунку 3.5 показаний ефект від вибору різних значень k_m на ефективність кластеризації для валідаційної вибірки ImageNet (ILSVRC-Validation).

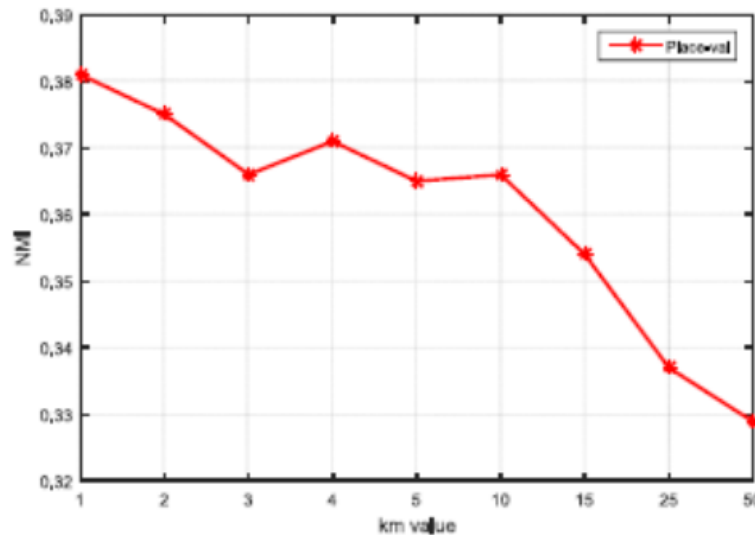


Рисунок 3.5 – Порівняння NMI запропонованого методу на різних значеннях k_m для ILSVRC-Validation

З рисунку 3.5 видно, що при $k_m=1$ досягається найкращий результат NMO, однак отримання такого результату займає найбільший час, який витрачається поліпшення параметрів мережі. Для досягнення компромісу між ефективністю та складністю, комбінація значень $N_m = 50$ та $k_m = 10$ здається розумним вибором, який веде до дещо кращого результату за NMI, та суттєво меншими витратами за часом у порівнянні з $k_m=1$. Зазначимо, що у випадку, коли розмір міні-пакета виявляється рівним розміру вибірки зображень і $k_m=1$, метод поводитиметься на кшталт CNN-RC, але лише його міні-пакетної варіації.

У порівнянні з CNN-RC, головна перевага кластеризуючої CNN полягає в тому, що міні-пакетна оптимізація може впоратися із проблемою кластеризації великого обсягу зображень. Так як ефективність кластеризації та обчислювальна складність мініпакетного CCNN зростає разом із збільшенням кількості епох для оновлення параметрів мережі, необхідно розрахувати ефективність кластеризуючої CNN для трьох датасетів із різними значеннями числа епох. На рисунку 3.6 наведено порівняння запропонованого підходу з випадковою ініціалізацією та з попередньою

моделлю для трьох тестових вибірок (ILSVRC12-Validation, Places-Training, Places-Validation) на різних значеннях кількості епох навчання.

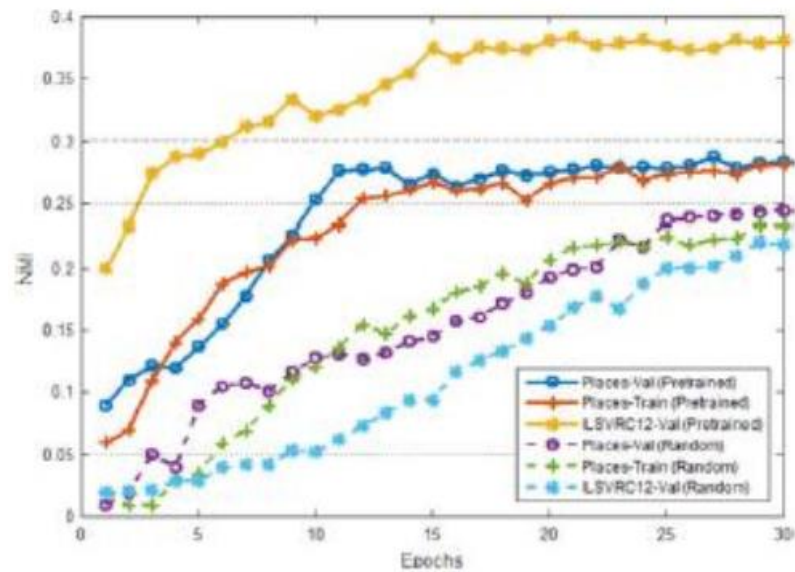


Рисунок 3.6 – Порівняння ефективності NMI запропонованого підходу на трьох різних тестових вибірках для різних значень числа епох

Як бачимо, NMI ефективність CCNN з попередньою моделлю досягає свого піку до 14 епохи для всіх тестових вибірок, тоді як CCNN із випадковою ініціалізацією досягає піку після 30 епохи. При значеннях $N_m = 50$ та $k_m = 10$ всі екземпляри у вхідній множині зображень можуть бути обрані принаймні один раз кожні 5 епох ітеративного процесу оновлення параметрів. Таким чином, пропонується встановити число епох рівним 10 або менше 10 для навчання параметрів мережі CCNN з метою досягнення прийняттого компромісу між ефективністю кластеризації та обчислювальною складністю.

ВИСНОВКИ

У ході підготовки кваліфікаційної роботи проведено аналіз алгоритмів кластеризації зображень на базі штучної нейронної мережі. Розроблено архітектуру самонавчальної згорткової нейронної мережі з додаванням адаптаційних шарів, при цьому основою є згорткова нейронна мережа AlexNet, а кластеризація реалізується вдосконаленим алгоритмом k-середніх. Для роботи мережі було використано алгоритм функціонування суміщеного навчання на основі згорткової нейронної мережі, методу кластеризації K-середніх та методу опорних векторів як способу ідентифікації некоректного розбиття на кластери на ранній стадії.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Блохина Т. Особенности исследования алгоритмов обработки изображений // Современные наукоемкие технологии. – 2014. – № 5(2). – С. 3-31.
2. Круглов В.Н. ЦИФРОВАЯ ОБРАБОТКА ИЗОБРАЖЕНИЙ // Успехи современное естествознание. – 2011. – № 10. – С. 88-89.
3. Торрес Р.Д.С. Поиск изображений на основе контента: теория и приложения / Р.Д.С. Торрес, А.Х. Falcão // Rev. Informática Teórica e Apl. РИТА. – 2006. – Вып. 13. – № 2. – С.161–185.
4. Като Т. Архитектура базы данных для поиска изображений на основе контента / Т. Като // Системы хранения и поиска изображений. – 1992. – Вып. 1662. – С. 112–123.
5. Фликнер М. Запрос по изображению и видеоконтенту: система QVIC / М. Фликнер [и др.] // Компьютер (Лонг. Бич, Калифорния). – 1995. – Вып. 28. – № 9. – С. 23–32.
6. Хампапур А. Видеодвижок Virage / А. Хампапур [и др.] // Хранение и Поиск по базам данных изображений и видео. Т. – 1997. – Том. 3022. – С. 188–198.
7. Мукерджа С. AMORE: Механизм поиска изображений во Всемирной паутине / С. Мукерджа, К. Хирата, Ю. Хара // Конференция по человеческому фактору в вычислениях Системы -- Труды. – 1999. – С. 17–18.
8. Пентланд А. Фотокнига: Контентная обработка баз данных изображений / А. Пентланд, Р. В. Пикард, С. Скларофф // Int. Дж. Вычисл. Вис. – 1996. – Вып. 18. – № 3. – С. 233–254.
9. Смит Дж. Р. VisualSEEk: полностью автоматизированный запрос изображения на основе содержимого. система / Дж.Р. Смит, С.Ф. Чанг // Материалы 4-й Международной конференции АСМ. Конференция по мультимедиа, МУЛЬТИМЕДИА 1996. – 1997. – С. 87–98.

10. Ма В.Ю. NeTra: набор инструментов для навигации по большим базам данных изображений / В.Ю. Ма, Б.С. Манджунатх // Мультимедийная система. – 1999. – Вып. 7. – № 3. – С. 184–198.

11. Ван Дж.З. Индексация и поиск изображений на основе контента с помощью Daubechies вейвлеты / Дж.З. Ван [и др.] // Int. Дж. Цифра. Либр. – 1997. – Вып. 1. – № 4. – С. 311–328.

12. Крижевский А. Классификация ImageNet с глубокой сверточной нейронной сетью Сети/ А. Крижевский, И. Суцкевер, Э.Г. Джеффри // Нейронная информация Системы обработки. 25. – 2012. – С. 1106–1114.

13. Wang C. ProductNet: коллекция высококачественных наборов данных о продуктах. репрезентативное обучение / К. Ван [и др.] // The Web Conference 2019 - Companion конференции World Wide Web, WWW 2019. – 2019. – С. 411–414.

14. Миколайчик К. Оценка эффективности локальных дескрипторов / К. Миколайчик, К. Шмид // IEEE Transactions on Pattern Analysis and Machine Интеллект. – 2005. – Вып. 27. – № 10. – С. 1615–1630.

15. Юн С.Х. Инвариантное стереосопоставление освещения и камеры / С.Х. Йонг, М.Л. Кён, У.Л. Санг // 26-я конференция IEEE по компьютерному зрению и шаблонам Признание, CVPR. – 2008. – С. 1–8.

16. Дяченко В.О., Житник М.С., Коваленко А.А. Алгоритми кластеризації зображень на базі штучної нейронної мережі// Проблеми інформатизації : десята міжнародна науково-технічна конференція. Черкаси – Баку – Бельсько-Бяла – Харків, 2022, т.2, с.92.