Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Інфокомунікацій
(повна назва)

Кафедра Інформаційно-мережної інженерії
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти другий (магістерський)

Дослідження методів сегментації зображень з використанням нейронних мереж

(тема)

Виконав:
здобувач 2 року навчання,
групи ІМІм-24-1
Борох А.В.
(прізвище, ініціали)

Спеціальність 172 Електронні комунікації та радіотехніка
(код і повна назва спеціальності)

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

Освітня програма Інформаційно-мережна інженерія
(повна назва освітньої програми)

Керівник: доц. Омельченко С.В.
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри _____ Микола МОСКАЛЕЦЬ
(підпис) (прізвище, ініціали)

2025 р.

Не містить відомостей, заборонених до відкритого публікування

Студент _____ / Борох А.В . /
(підпис) (прізвище та ініціали)

Керівник _____ / Омельченко С.В. /
(підпис) (прізвище та ініціали)

Харківський національний університет радіоелектроніки

Факультет Інфокомунікацій

Кафедра Інформаційно-мережної інженерії

Рівень вищої освіти другий (магістерський)

Спеціальність 172 Електронні комунікації та радіотехніка

(код і повна назва)

Тип програми освітньо-професійна

Освітня програма Інформаційно-мережна інженерія

(повна назва)

ЗАТВЕРДЖУЮ:
Зав. кафедри _____
(підпис)

« 28 » жовтня 2024 р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві Борох Артем Володимирович
(прізвище, ім'я, по батькові)

1. Тема роботи Дослідження методів сегментації зображень з використанням нейронних мереж

затверджена наказом по університету від« 24» жовтня 2025 р. № 959 Ст

2. Термін подання студентом роботи до екзаменаційної комісії 22 грудня 2025 р.

3. Вхідні дані до роботи Виконати огляд методів сегментації зображень Розглянути особливості попередньої обробки зображень та оцінювання параметрів ознак та методи прийняття рішень. Розглянути сегментацію зображень з використанням нейронних мереж. Виконати дослідження сегментації зображень.

4. Перелік питань, що потрібно опрацювати у роботі Вступ

1. Огляд методів сегментації зображень

2. Попередня обробка та оцінювання параметрів

3 Сегментація зображень з використанням нейронних мереж

4 Результати досліджень сегментації зображень

Висновки

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (п.5 включається до завдання за рішенням випускової кафедри) слайди презентації в форматі Power Point (назва роботи, актуальність, огляд методів, результати досліджень, висновки)

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Вступ	26.10-2.11.25	виконано
2	Огляд методів сегментації зображень	3.11-09.11.2025	виконано
3	Попередня обробка та оцінювання параметрів	10.11-12.11.25	виконано
4	Сегментація зображень з використанням нейронних мереж	13.11-20.11.25	виконано
5	Результати досліджень сегментації зображень	21.11-02.12.25	виконано
6	Висновки		виконано
7	Оформлення пояснювальної записки	03.12-22.12.25	виконано

Дата видачі завдання 25 жовтня 2025 р.

Здобувач _____
(підпис)

Керівник роботи _____ доц. Омельченко С.В.
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка: 83 с., 26 рис., 8 табл., 36 джерел, 1додаток.

Об'єкт дослідження– Процес автоматизованої сегментації цифрових зображень за допомогою нейронних мереж різних архітектур

Предметом дослідження– Чисельні залежності точності сегментації зображень від архітектурних та гіперпараметричних параметрів нейронних оцінювані за метриками DSC та JSC.

Мета роботи –Визначити та проаналізувати чисельні залежності точності сегментації зображень від архітектурних та гіперпараметричних параметрів нейронних мереж .

Виконано огляд методів в області автоматичної сегментації зображень. Вона дозволяє аналізувати більший обсяг даних з вищою швидкістю, точністю та узгодженістю в різних застосуваннях. Розглянуто кілька архітектур з точки зору їхньої структури, кількості параметрів та компонентів, що навчаються, які здатні ефективно досягати найсучасніших показників у різних завданнях сегментації.

Наведено порівняння результатів, отриманих при використанні різних нейронних мереж.

СЕГМЕНТАЦІЯ ЗОБРАЖЕННЯ, НЕЙРОННА МЕРЕЖА, ТОЧНІСТЬ СЕГМЕНТАЦІЇ.

THE ABSTRACT

Explanatory note: 83 p., 26 fig., 8 tab., 36 sources, 1 appendix.

Object of research – The process of automated segmentation of digital images using neural networks of different architectures

Subject of research – Numerical dependences of image segmentation accuracy on architectural and hyperparametric parameters of neural networks estimated by DSC and JSC metrics.

Purpose of work – To determine and analyze numerical dependences of image segmentation accuracy on architectural and hyperparametric parameters of neural networks.

A review of methods in the field of automatic image segmentation is carried out. It allows analyzing a larger amount of data with higher speed, accuracy and consistency in various applications. Several architectures are considered in terms of their structure, number of parameters and components being trained, which are able to effectively achieve state-of-the-art performance in various segmentation tasks.

A comparison of the results obtained when using different neural networks is presented.

IMAGE SEGMENTATION, NEURAL NETWORK, SEGMENTATION ACCURACY.

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ

ЗНМ - Згорткова нейронна мережа;

ШІ -штучний інтелект;

MLP - Багатошаровий персептрон;

ЗГЗ- Згорткова глибока мережа;

ВСТУП

Сегментація зображення – це процес поділу цифрового зображення на декілька елементів. Сегментація необхідна для спрощення аналізу, так як утворюються менші елементи зображення, що входять до загального, а це полегшує аналіз. В результаті сегментації зображень виконується виділення об'єктів та їх меж в вигляді лінії, кривих на зображеннях. Тому в процесі сегментації зображень виконується процес присвоєння таких міток кожному пікселю зображення, при цьому пікселі з однаковими мітками будуть мати спільні візуальні характеристики.

Сегментація зображень у комп'ютерному зорі виявилися високоефективними з використанням нейронних мереж, зокрема згорткових нейронних мереж (ЗНМ). Сегментація зображень має численні практичні застосування, такі як медична візуалізація, автономне керування та спостереження. ЗНМ здатні вивчати складні ознаки безпосередньо із зображень та досягати видатної продуктивності в кількох наборах даних. Існують дослідження ефективності різних методів попередньої обробки та класифікації для точної сегментації та класифікації різних структур нейронних мереж.

У світі штучного інтелекту (ШІ) та інформатики комп'ютерний зір – це галузь, яка зосереджена на наданні комп'ютерам можливості інтерпретувати, обробляти та розуміти візуальні дані із зовнішнього середовища. Вона включає розробку алгоритмів та методів для обробки та аналізу зображень і відео, а також вилучення з них змістовних висновків та інформації. Згорткова нейронна мережа (ЗНМ) – це тип глибокої нейронної мережі, створеної для обробки та оцінки, яка має структуру, подібну до сітки даних, разом із зображеннями або фільмами. Щодо комп'ютерного зору, ідентифікація зображень є ключовим завданням, і ЗНМ стали найсучаснішою технікою для цього.

Мотивацією використання нейронних мереж, зокрема змішаних нейронних мереж (CNN), для сегментації є їхня продемонстрована ефективність у різних завданнях комп'ютерного зору.

Нейронні мережі пропонують підвищену точність, вищу швидкість обробки, адаптивність до різноманітних типів даних та потенціал для нових застосувань. Використовуючи структуру та роботу людського мозку, ці моделі можуть автономно обробляти та аналізувати зображення, зменшуючи залежність від ручного втручання.

Однак використання нейронних мереж в обробці зображень створює труднощі. Навчання великомасштабних глибоких нейронних мереж вимагає значних обчислювальних ресурсів.

1 ОГЛЯД МЕТОДІВ СЕГМЕНТАЦІЇ ЗОБРАЖЕНЬ

Сегментація зображень є необхідною умовою майже для всіх програм комп'ютерного зору. Вона дозволяє витягувати значущу інформацію з візуальних вхідних даних шляхом розділення зображень на сегменти зі спільними ознаками. Існує широкий набір методів сегментації, починаючи від класичних фреймворків і закінчуючи архітектурами глибокого навчання.

1.1 Завдання сегментації зображень

Як зазначає Бандьопадхай [1], існують три види завдань сегментації зображень, залежно від їхніх результатів. Вони проілюстровані на рис. 1.1.

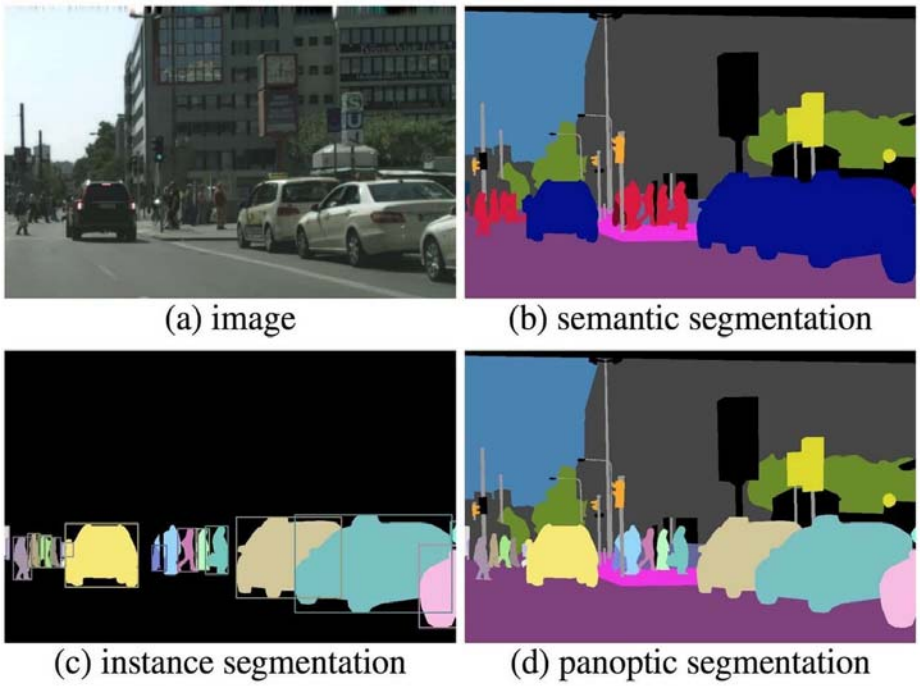


Рисунок 1.1 – Семантична, екземплярна та паноптична сегментація

Виходячи з концепцій глибокого навчання [2], основна інтуїція полягає в тому, щоб класифікувати кожен піксель у клас незалежно від кількості класів. Недоліком використання семантичної сегментації є те, що вона класифікує багато подібних класів як один [1]. Більше того, варіант семантичної сегментації, який називається семантичною сегментацією на рівні сцени, зосереджується на загальному зображенні як на одній сцені . Хоча це має

ширший фокус на загальній композиції зображення, семантична сегментація, як правило, є більш детальною, ідентифікуючи окремі об'єкти в межах сцени [3].

Також концептуалізована глибоким навчанням, сегментація екземплярів сегментує окремі екземпляри до точних меж, замість того, щоб зосереджуватися на загальних класах. Це досягається шляхом характеристики результату з вищим рівнем складності та деталізації [5].

Простими словами, паноптична сегментація – це поєднання семантичної та екземплярної сегментації. Цей тип комбінації не лише кластеризує сегменти в різні класи, але й використовує можливість підрахунку екземплярів кожного з них. Недоліком паноптичної сегментації є її вимога до значної узгодженості та обчислювальної ефективності, що є перешкодою в більшості випадків [6].

1.2 Огляд традиційних методів сегментації зображень

Як зазначає Бандьопадх'яй [1], традиційні методи сегментації зображень – це, перш за все, комбінація цифрової обробки зображень з алгоритмами, що допомагають в оптимізації. На відміну від моделей на основі глибокого навчання, архітектури цієї категорії включають оточення життєво важливих країв за допомогою енергетичних сил [8], використання логіки потоку та підйому [9] та групування подібних кластерів без їх маркування [10].

У своїй статті Вінсент та Сойль [9] пропонують ґрунтовну методологію того, що яскравіший піксель відповідає вищому значенню висоти. Після цього на зображенні створюються покажчики на основі точок локального мінімуму, таких як загальна зміна яскравості сусідніх пікселів. Автори називають це розділення основою водозбору . За збігом обставин, початкові варіанти використання водозбору поширювалися в галузі топографії, вивчення земних поверхонь.

Хоча використання цього методу обмежувалося зображеннями у градаціях сірого приблизно з 1991 року [9], відбулися суттєві розробки для подолання цього обмеження. Протягом шести років Шафаренко та ін. [11] розробили методологію, в якій початкова архітектура водозбору в поєднанні з алгоритмом сегментації знизу вгору дозволяє успішно працювати не лише з кольоровими зображеннями, але й із зображеннями з різними текстурами.

Активний контур починається з попередньо визначеного деформованого контуру (контуру) [8] і повзе всередину до країв (ребр, ліній або градієнтів) об'єкта, як показано на рис. 2. Цей метод базується на мінімізації енергетичної функції контуру. Касс та ін. [8] стверджують, що енергетична функція використовує (i) зовнішні сили, які притягують контур до елементів зображення, таких як ребра, лінії або градієнти, та (ii) внутрішні сили, які контролюють гладкість і вигин контуру, запобігаючи його надмірній нерівності.

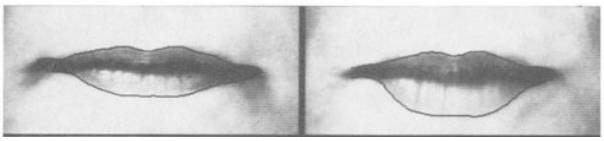


Рисунок 1.2 – Використання активної контурної сегментації

Алгоритми кластеризації, такі як відомі K-середні, аналізують точки даних для групування їх на основі подібних характеристик [10]. У випадку зображень, точки даних – це пікселі. Частина алгоритму вимагає визначення ознаки, за якою слід формувати кластери. Це, зокрема, включає відповідні ознаки, такі як колір або текстура. Потім алгоритм виконує групування цих пікселів у кластери, і процедура завершується перетворенням кожного кластера на сегмент. Коулман та ін. [10] зазначають, що кластеризація була частиною традиційних методів сегментації зображень протягом кількох десятиліть, коли вона вперше була використана для постановки задач таксономії. Цей метод також широко застосовується в медичній галузі для визначення ROI [4]. Наразі використовуються чотири види алгоритмів кластеризації: K-середні, нечіткі C-середні, гірський та субтрактивний кластеризація [12].

1.3 Огляд методів сегментацій на основі нейронних мереж

Хоча традиційні архітектури спочатку прокладали шлях, вони почали втрачати продуктивність, оскільки інші технологічні досягнення враховували це. З цього приводу Мінаї та ін. [7] стверджують, що архітектури сегментації зображень, концептуалізовані на моделях глибокого навчання, очевидно, забезпечили кращу продуктивність. По-перше, глибокі нейронні мережі (DNN) – це назва, дана категорії машинного навчання (ML), яка в основному

займається візуальним навчанням з використанням нейронних структур, подібних до людського мозку. Однак автори в статті далі поділяють їх на підгрупи на основі їхніх технічних архітектур [7].

Архітектури на основі DL потребують навчання на великих наборах даних. Коли отримання достатньої кількості позначених елементів даних неможливе, отримана модель погано показує точність тестових даних. Практичний підхід використовує перенесення навчання [7], яке попередньо навчає модель на загальному наборі даних, що містить достатню кількість позначених даних. Потім останні кілька шарів цієї моделі перенавчаються на меншому наборі даних для конкретної задачі. Це значно скорочує час навчання та розмір набору даних, зберігаючи при цьому високу точність.

Трансферне навчання було використано в сегментації зображень десять років тому ван Опбруком та ін. у медичній галузі, зокрема, при МРТ-скануванні мозку [13]. Вони навчали класифікатори з трансферним навчанням та без нього та дійшли висновку, що включення трансферного навчання покращило якість сегментації, потребувало меншої кількості мічених зразків для тієї ж точності та мінімізувало понад половину помилок класифікації, що виникають в іншому випадку. Зовсім недавно Маюрський та ін. покращили сегментацію контурів клітин на мікроскопічних зображеннях за допомогою трансферного навчання та генеративно-змагальних мереж (GAN) [14].

Згорткові нейронні мережі (ЗНМ) є однією з перших архітектур нейронних мереж, CNN складається з 3 основних шарів – згорткового, нелінійного та об'єднуючого. Кожен шар функціонує унікально, щоб забезпечити успіх цієї архітектури. Деякі з характеристик включають використання ваг для вилучення ознак, можливість моделювання нелінійних функцій та перетворення карт ознак на корисну статистичну інформацію. На відміну від традиційних нейронних мереж, CNN навчається автоматично вилучати багаторівневі ознаки з суміжних ділянок зображення. Це усуває необхідність ручної інженерії ознак. Єдиний аспект, де CNN обмежена, стосується розміру зображення, оскільки вони часто попередньо навчаються з фіксованими розмірами зображень. Одним з найвідоміших методів, побудованих на CNN, є VGG-16 [15], шістнадцятишаровий CNN, який виводить карту ознак.

Вперше випущені в 1980 році Фукусімою К. [16], ЗНМ є одними з перших глибоких нейронних мереж такого типу, розроблених для сегментації зображень. Перший реліз цього методу мав назву «Нейрокогнітрон» [16]. Сьогодні вони продовжують успішно використовуватися в багатьох областях для обробки природної мови та аналізу часових рядів і є одним з найуспішніших методів сегментації зображень на сьогодні [7]. Ще один важливий розвиток в її історії спостерігався в 1995 році, коли Вайбель та ін. [17] ще більше революціонізували ЗНМ, ввівши ваговий компонент до вже існуючої архітектури. Прямий випадок використання цієї версії використовується в розпізнаванні фонем, яке визначається як дослідження, пов'язане з найменшою одиницею звуку.

Спираючись на приклад CNN, Лонг та ін. [18] запропонували повністю згорткову мережу (FCN). Це були деякі з початкових методів сегментації зображень, які розвинули існуючі архітектури CNN. На відміну від свого попередника, FCN можуть приймати зображення довільного розміру як вхідні дані. Ще однією важливою відмінністю між CNN та FCN є те, що деякі CNN є повністю зв'язними, що еквівалентно пропуску їх згорткових аспектів. З іншого боку, FCN виконує згорткові операції незалежно від цього.

Хоча FCN вважалися оновленою версією передчасних CNN, вони мали свій набір недоліків. Одним з них була нездатність добре працювати з семантичними вхідними даними на рівні сцени. Озираючись на типи завдань сегментації, сегментація на рівні сцени розглядається в набагато ширшій перспективі порівняно із семантичною сегментацією [9]. Щоб вирішити цю проблему, Саттон та ін. [23] запропонували Умовні Випадкові Поля (CRF). Це один з небагатьох методів, які підходять до сегментації зображень з доданим до неї графічним компонентом. Один з інших методів, Марковські Випадкові Поля (MRF), також є чудовим прикладом цього. Хоча CNN функціонували ефективно та демонстрували високі показники продуктивності в завданнях класифікації, додавання CRF або MRF стосувалося головним чином того, щоб зробити ці моделі додатково здатними виконувати сегментацію об'єктів на рівні класифікації [7].

В останні роки моделі, засновані на увазі, були визнані однією з послідовних моделей семантичної сегментації, що базуються на глибокому навчанні. Мінае та ін. [7] стверджують, що моделі, засновані на увазі,

продовжують досліджуватися для сегментації зображень. З цією метою нижче розглядається, що таке моделі уваги та що робить їх невід'ємною частиною існуючих методів сегментації.

Якби існувало одне слово, яке могло б зручно описати цей метод, то це було б «розсудливий ». Це означає, що моделі на основі уваги можуть сегментувати важливі аспекти зображення незалежно від того, наскільки вони віддалені порівняно з іншими важливими класами у вхідних даних. Технічна складність цього полягає у призначенні м’якої ваги багатомасштабним ознакам [7]. Як продемонстрували Чен та ін. [20], на рис. 3 показано, як обидва суб’єкти, незалежно від відстані, були враховані шляхом перемикання уваги на різних суб’єктів у різних масштабах. Крім того, архітектура на цьому рисунку поєднує вже існуючу семантичну модель та вводить модель уваги посередині, одночасно навчаючи на них багатомасштабні зображення [20].

Що стосується продуктивності, Мінае та ін. [7] зазначають, що моделі, засновані на увазі, можуть домінувати в середньому та максимальному об'єднанні, двох шарах, які часто використовуються в CNN для розпізнавання зображень. Наявність моделі, яка враховує важливість ознак на різних відстанях та масштабах, може ще більше покращити тривимірні аспекти навіть двовимірного зображення, що, очевидно, змінило межі, до яких можна рекомендувати сегментацію зображень.

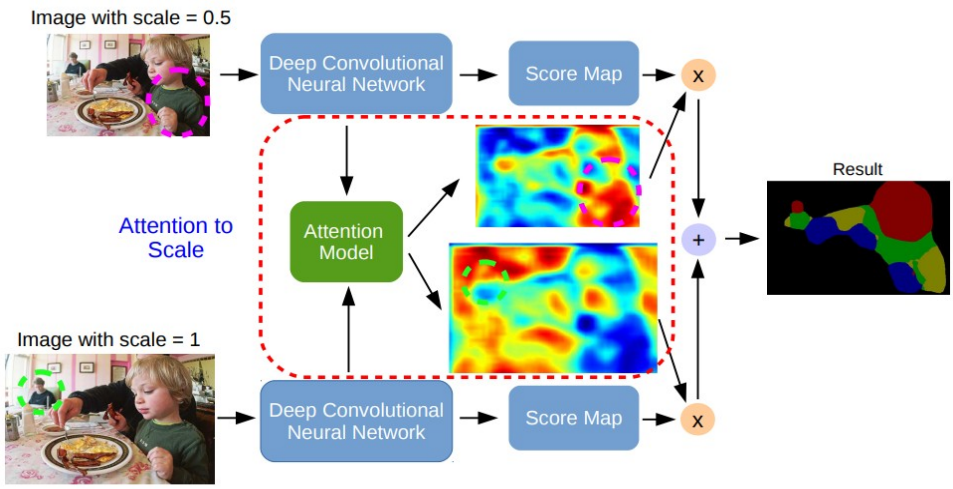


Рисунок 1.3 –Архітектура DL на основі уваги

Один зі способів розглянути методи, засновані на архітектурі кодера-декодера, полягає в візуалізації вертикального пісочного годинника, повернутого на 90 градусів. З огляду на це, ця архітектура складається з двох основних компонентів - згорткової мережі та деконволюційної (транспонованої згорткової) мережі. Мережа кодера натхненна мережею VGG-16, CNN, описаною раніше. У кодерах-декодерах за 16 шарами з VGG-16 йде деконволюційна мережа [7]. Ключова відмінність між використанням CNN, таких як VGG-16, та використанням кодерів-декодерів, таких як SegNet [21], полягає в їхніх виходах. У той час як VGG-16 створює карту ознак, SegNet, за допомогою компонента деконволюції, створює карту сегментації, яка має такий самий розмір, як і вхід.

Архітектуру енкодер-декодер вперше запропонував Но та ін. [22], де вони детально пояснюють деконволюцію та причини, чому вона буде прийнята як промисловий варіант використання. Ще однією з особливостей архітектури кодера-декодера є те, що вона не містить жодних повністю зв'язаних шарів, що робить її просто згортковою. У 2016 році Бадрінараян та ін. [21] запропонували вищезгадану техніку, Segnet. Незначна зміна в архітектурах кодера-декодера за замовчуванням полягає в тому, що кодер використовує архітектуру, топологічно подібну лише до 13 шарів з VGG-16 порівняно зі звичайними 16 шарами, за якими йдуть деконволюційна мережа (декодер) та рівень класифікації. Мінае та ін. [7] зазначають, що причина, чому SegNet проникає в промислові варіанти використання, полягає в унікальному способі підвищення роздільної здатності вхідних даних з нижчою роздільною здатністю без необхідності фактичного навчання підвищенню роздільної здатності. На відміну від CNN та інших архітектур, SegNet також використовує меншу кількість параметрів, що навчаються [7], що призводить до скорочення часу навчання, зменшення використання обчислювальних ресурсів та ефективного використання пам'яті.

1.4 Порівняння методів сегментацій на основі нейронних мереж

У науковій літературі багато публікацій зосереджені на методах опитування та їх ефективності. Вони включають розширення численних класичних методів та методів глибокого навчання, порівнюючи їх ефективність

на добре відомих наборах даних. Ці добре відомі дослідницькі та оглядові статті демонструють домінування методів, заснованих на глибокому навчанні, над їхніми класичними аналогами з числовими доказами. Це дослідження також має на меті зробити внесок у ці аспекти. Крім того, побічним продуктом цього внеску може бути розширення можливостей класичних методів, здатних перевершити найсучасніші архітектури, засновані на глибокому навчанні, принаймні в одному практичному сценарії.

Така реальність може призвести до кількох міркувань щодо майбутньої сфери сегментації зображень, зокрема наступних питань. Наскільки віддалені поточні показники продуктивності класичних методів та методів глибокого навчання? Чи існують випадки використання, коли комбінація двох підходів перевершує їх окремі показники? Незважаючи на існування сучасних методів на основі глибокого навчання, якою мірою класичні методи слід розглядати для майбутньої сфери сегментації зображень?

У цій області набори даних класифікуються як 2D, 2.5D та 3D набори даних, де D – це глибина або кількість просторових вимірів. Наприклад, PASCAL Visual Object Classes (VOC) – це 2D набір даних, що представляє анотовані зображення, які можна використовувати для 5 завдань у 21 класі об'єктів. Інші глобально використовувані набори даних включають Microsoft Common Objects in Context (MS COCO) та Cityscapes. З іншого боку, 2.5D набори даних, такі як SUN RGB-D, визначають глобальний еталон для вхідних даних RGB-D та в основному диктують мету для завдань рівня сцени з масштабом, подібним до PASCAL VOC. Нарешті, ShapeNetCore – це 3D набір даних, що представляє окремі чисті 3D-моделі приблизно 51 категорії та 51 300 унікальних 3D-моделей [7]. Ці набори даних, як правило, добре відомі та є чітко визначеною зменшеною проекцією реальних вхідних даних та сценаріїв.

Перш ніж розрізняти класичну та глибоко навчальну архітектури, нижче наведено деякі широко прийняті показники оцінки, що використовуються сьогодні для алгоритмів сегментації.

Точність пікселів (PA) зосереджена на співвідношенні правильно класифікованих пікселів, поділеному на загальну кількість пікселів у (K + 1) класах (K класів та фон) [7].

$$\text{PA} = \frac{\sum_{\text{i=0}}^{\text{k}} \text{p}_{\text{ii}}}{\sum_{i=0}^{k}\sum_{j=0}^{k} \text{p}_{\text{ij}}},$$

де p_{ij} – кількість пікселів у класі i, що проектуються як частина класу j .

Середня точність пікселів (MPA) охоплює співвідношення правильних пікселів за класами перед їх усередненням за кількістю класів [7].

$$MPA = \frac{1}{K+1}\sum_{i=0}^{K}\frac{p_{ii}}{\sum_{j=0}^{K} p_{ij}}.$$

Перетин над об'єднанням (IoU), значення якого коливається від 0 до 1, є мірою, що визначає спільність, поділену на площу передбачуваної карти сегментації та істинний вихідний код

$$IoU = J(A,B) = \frac{|A \cap B|}{|A \cup B|},$$

де A позначає базову істину, а B позначає карту сегментації.

На додаток до цього, Mean-IoU позначає середнє значення по всіх класах і використовується глобально для методів, опублікованих в сучасну епоху обчислень [7].

Точність та повнота – це два з трьох глобально визнаних показників ефективності, що стосуються достовірності сегментації [7]

$$\boldsymbol{Precision = \frac{TP}{TP+FP}}, \quad \boldsymbol{Recall = \frac{TP}{TP+FN}},$$

де TP позначає істинно позитивний результат, FP позначає хибнопозитивний результат, а FN позначає хибнонегативний результат.

Третій показник, F1-оцінка, виглядає наступним чином [7]:

$$\boldsymbol{F1 - score = \frac{2\ Prec\ Rec}{Prec + Rec}}$$

Подібно до F1-оцінки, багато наборів даних також використовують коефіцієнт подібності кубиків (DSC) [27] як значущу тестову метрику, яка описується як:

$$DSC = \frac{2TP}{(TP + FP) + (FN + TP)}.$$

Гарсія-Гарсія та ін. [24] у своїй оглядовій статті відзначають чудову продуктивність архітектур глибокого навчання для загальної сегментації зображень та відео. У таблиці 1.1 наведено відсотки точності кількох із цих методів.

В іншому дослідженні Джйоті та Сінгх [25] досліджують різні методи сегментації медичних зображень, що включають сегментацію ділянок мозку для виявлення пухлин. Для покращення продуктивності різні автори використовували певні методи зі своєю існуючою архітектурою. Деякі з них

Таблиця 1.1 – Точність методів сегментації зображень

Архітектура на основі DL	Набір даних	Найкраща точність (%)
PSPNet	PASCAL VOC-2012	85,4
Глибока лабораторія	PASCAL VOC-2012	79,7
DAG-RNN	CamVid	91,6
SegNet	CamVid	60,1
rCNN	Стенфордська історія	80,2
LSTM-CF	SUN3D	58,5
PointNet	Частина ShapeNet	83,7
PointNet++	Частина ShapeNet	85,1
DGCNN	Частина ShapeNet	85,1

Це лінеаризований класифікатор розріджених репрезентативних даних ядра (LKSRC), доповнення даних (DA), просторове перетворення (ST) та перетворення інтенсивності (IT). Деякі з методів на основі DL включають використання баєсівської нечіткої кластеризації (BFC), автокодувальників (AE), розбавленої нейронної мережі (DNN), генеративно-змагальної мережі (GAN).

Більшість методів сегментації, які працюють значно краще, побудовані на глибокому навчанні. Однак, існують два такі класичні методи, які демонструють значну близькість точності до архітектур на основі глибокого навчання порівняно з архітектурою, запропонованою Тангом та ін. [29]. Глибший розгляд їхніх методів показує, що як Лю та ін., так і Чжао та ін. включили LKSRC, DA, ST та IT як частину своїх допоміжних методів для існуючих класичних архітектур. Як далі стверджують Джйоті та Сінгх [29], використання цих методів призвело до того, що вони перевершили деякі архітектури на основі глибокого навчання.

Третє дослідження, проведене Ганом та ін. [26] щодо використання класичної та на основі глибокого навчання сегментації зображень для остеоартриту коліна, показало, що класичні методи сегментації хряща підтримували точність від 70 до 88%, тоді як класичні моделі сегментації кісток продемонстрували значне зростання точності до 90-97%. З іншого боку, моделі сегментації хряща та кісток, засновані на глибокому навчанні, отримали ДСК від 80–90% до 97–98% відповідно.

Ключовим спостереженням, зробленим з опитування, є те, що хоча моделі на основі глибокого навчання можуть демонструвати високі показники продуктивності, існує кілька областей, таких як медична візуалізація, де класичні методи здатні відображати незначні відмінності в точності порівняно з методами, заснованими на глибокому навчанні (DL). Причина полягає в тому, що медичні зображення, такі як рентгенівські знімки, часто містять менше деталей порівняно із загальними кольоровими зображеннями. Це обмежує можливості методів DL, оскільки вони не здатні витягти достатньо інформації з таких зображень. Крім того, у випадку Джйоті та Сінгха [25], можливість використання класичних методів у поєднанні з певними коригувальними методами може дати конкурентоспроможні результати порівняно з їхніми аналогами на основі DL. Додатковою перевагою є те, що класичні методи ефективно використовують обчислювальні ресурси, зберігаючи при цьому хорошу точність.

Таким чином мережі понад десять років тому, в рамках так званого глибокого навчання (ГН). Відтоді проблеми комп'ютерного зору були переглянуті шляхом їх вирішення за допомогою методів ГН. Це включає проблему сегментації, одну з найважливіших задач у більшості задач

комп'ютерного зору, яка є темою дослідження в цій роботі, а також детальний опис видів завдань, категорій, які вона включає, та кількох відомих архітектур, що належать до цих категорій. Більше того, завдяки наявності великих обсягів навчальних даних, методи на основі ГН передбачувано перевершили багато інших раніше розроблених методів. Однак це сталося за рахунок тривалого часу обробки та використання додаткових обчислювальних ресурсів. У цій статті ми розглянули розвиток та точність методів ГН та класичних методів у задачах сегментації, пов'язаних з реальними сценаріями. Наш головний внесок полягає в тому, що ми показали, як деякі дослідники нещодавно повторно спробували сегментацію, використовуючи класичні підходи в додатках, де точність була порівнянною, тоді як обчислювальна ефективність була значно на їхню користь.

2 ПОПЕРЕДНЯ ОБРОБКА ТА ОЦІНЮВАННЯ ПАРАМЕТРІВ

Для забезпечення ефективної реалізації нейронної мережі необхідно оцінювання параметрів та попередня обробка. Для цього необхідні різні гіперпараметри та алгоритми, попередня та постобробка даних, а також метод оцінювання параметрів. Попередня обробка даних включає нормалізацію, покращення, доповнення та балансування, тоді як постобробка даних включає операції математичної морфології та маркування зв'язних компонент.

2.1 Гіперпараметри та алгоритми

Алгоритм оптимізації використовується в навчанні нейронної мережі для мінімізації функції вартості з метою збіжності до глобального мінімуму.. Однак, оскільки задача, як правило, неопукла, вона не обов'язково збігається до глобального мінімуму. Натомість, вона прагне збігатися до інших стаціонарних точок, таких як сідлові точки або локальні мінімуми. Для оптимізації навчання нейронної мережі можна розглянути ряд гіперпараметрів та алгоритмів. Гіперпараметри включають швидкість навчання, константу імпульсу, швидкість спаду та константу регуляризації. Кожен з цих гіперпараметрів впливає на збіжність та здатність мережі до узагальнення.

2.2 Попередня обробка даних

Попередня обробка даних – це процес, який перетворює дані у формат, що підходить для вхідних даних мережі. Вона також може бути використана для перетворення вхідних даних в інше представлення для полегшення процесу навчання та досягнення кращої узагальнюючої здатності мережі. У більшості практичних застосувань попередня обробка даних є важливим елементом рішення, що визначає ефективність навчання .

Найпоширеніші попередні обробки даних для нейронних мереж включають нормалізацію, покращення, доповнення та балансування.

2.3 Нормалізація

Нормалізація – це процес, який перемасштабує дані до значень подібного діапазону, зазвичай від 0 до 1.

Найпростішим методом нормалізації є мінімаксна нормалізація, яка виражається як:

$$x' = \frac{x - \min(X)}{\max(X) - \min(X)}, \tag{2.1}$$

де x – елемент початкового набору даних , x' – нормалізоване значення, X -позначає початкову множину.

Основним недоліком min-max нормалізації є те, що вона може зіткнутися з помилкою, якщо будь-які нові дані не потрапляють у початковий вхідний діапазон, тобто [min(X) , max(X)]. Таким чином, цей метод підходить лише для вхідних даних з відомим та фіксованим діапазоном даних .

Іншим типом нормалізації є нормалізація z-показника, також відома як стандартизація. Вона центрує дані навколо 0 та ділить їх на стандартне відхилення даних, тобто

$$x' = \frac{x - X_\mu}{X_\sigma}, \tag{2.2}$$

де x – елемент множини X, x'– нормалізоване значення, X μ – середнє значення множини X , а X σ позначає стандартне відхилення множини X.

Як можна очікувати з рівняння 2.2, нормалізація z-оцінки вирішує проблему похибки «виходу за межі» при нормалізації мінімум-максимум. У даних зображення X μ та X σ можна обчислювати глобально для всіх каналів (наприклад, червоний, зелений та синій канали на зображенні RGB) або локально для кожного каналу по даних у міні-пакеті або для всього навчального набору даних.

2.3 Попередня обробка даних

Покращення даних – це метод, що використовується для покращення представлення даних для певного завдання. Його можна застосовувати до будь-

якого типу даних, включаючи дані зображень. Покращення зображення можна виконувати шляхом зміни атрибутів, таких як розподіл контрастності, яскравості або інтенсивності, і застосовувати як у просторовій, так і в частотній областях для досягнення бажаних результатів. У випадку застосування в просторовій області операція покращення застосовується безпосередньо до значень пікселів зображення. Для застосування в частотній області операція покращення застосовується до перетворення Фур'є зображення.

Точкова операція – це тип операції покращення зображення, де вихідний сигнал у кожному місці пікселя генерується виключно на основі пікселя у відповідному місці вхідного зображення, тобто операція не залежить від сусідніх пікселів. Розтягування контрасту – це базова точкова операція, яка використовує функцію кусково-лінійного перетворення для розтягування діапазону пікселів зображення таким чином, щоб воно займало повний діапазон пікселів, наприклад, [0, 255] для 8-бітного зображення. Розтягування контрасту збільшує динамічний діапазон рівнів інтенсивності зображення з низькою контрастністю. Операція полягає в наступному:

$$y(i,j) = \frac{x(i,j) - \min(x)}{\max(x) - \min(x)} (\max(y) - \min(\mathrm{y})) + \min(y) \tag{2.3}$$

де y – вихідне зображення, x – вхідне зображення, i,j – відповідно індекси рядка та стовпця, що представляють розташування пікселя на зображенні. Приклад операції розтягування контрасту показано на рисунку 3.1 . Отримане зображення має вищу контрастність, ніж вхідне зображення, отже, на зображенні можна спостерігати більше деталей.

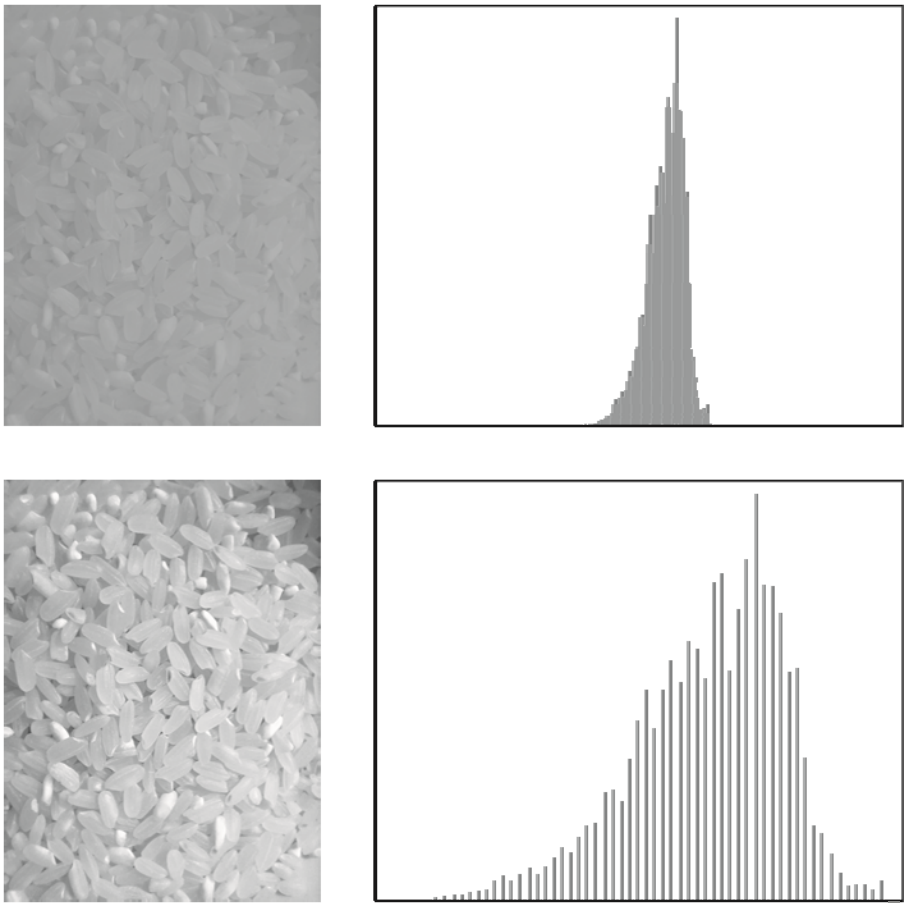


Рисунок 2.1 – Приклад операції розтягування контрасту. Зображення та гістограма значень пікселів на зображенні вхідного (зверху) та вихідного (знизу) сигналів.

Розтягування контрасту також може бути виконане з пороговим значенням, щоб вихідне зображення містило лише два рівні інтенсивності з надзвичайно високою контрастністю [70]. Ця операція зазвичай виконується на зображенні у градаціях сірого та виражається як:

$$y(i,j) = \begin{cases} 255, & коли \quad x(x,j) \geq T \\ 0, & коли \quad x(i,j) < T \end{cases}$$

де y – вихідне зображення, x – вхідне зображення, i,j – індекси рядка та стовпця відповідно, що представляють розташування пікселя на зображенні, а T– значення порогу.

Порогове значення можна визначити за допомогою різних методів вибору порогу, включаючи аналіз гістограми рівнів сірого та усереднення рівнів сірого.

Приклад операції розтягування контрасту з порогом, який визначається на основі аналізу гістограми рівнів сірого, показано на рисунку 2.2 .

Рисунок 2.2 – Приклад операції розтягування контрасту з пороговим значенням. Зображення та гістограма значень пікселів на зображенні вхідного (зверху) та вихідного (знизу) сигналів.

Ще одна операція посилення контрастності називається вирівнюванням гістограми. Метою вирівнювання гістограми є зміна зображення таким чином, щоб гістограма вихідного зображення була приблизно рівномірною. Операція вирівнювання гістограми полягає в наступному:

$$y(i,j) = (K-1)\sum_{k=0}^{x(i,j)} \frac{c_x(k)}{MN},$$

де y – вихідне зображення, x – вхідне зображення, M та N – висота та ширина зображення в пікселях відповідно, K – кількість рівнів інтенсивності, що використовуються для значень пікселів, а cx – загальна кількість певних значень пікселів у вхідному зображенні.

Як видно з прикладу операції вирівнювання гістограми на рисунку 2.3 , вирівняне зображення має вищу контрастність, ніж вхідне зображення.

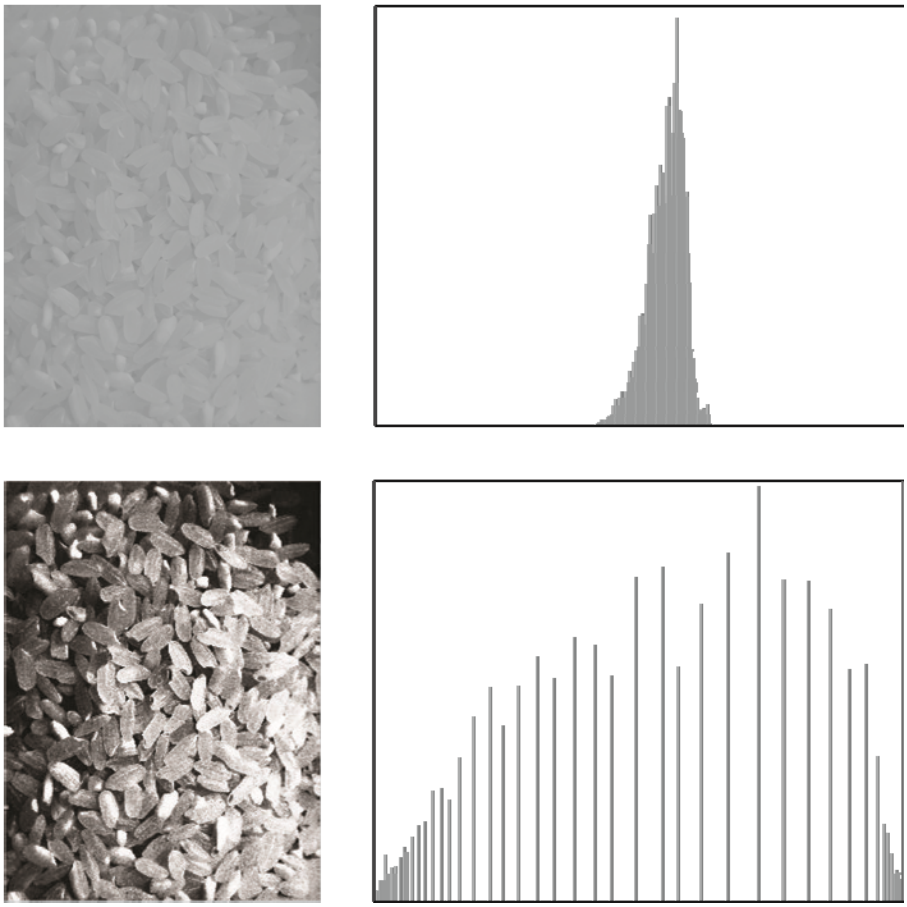


Рисунок 2.3 – Приклад операції вирівнювання гістограми. Зображення та гістограма значень пікселів на зображенні вхідного (зверху) та вихідного (знизу) сигналів

Операція фільтрації – це тип операції покращення зображення, де вихідний сигнал у кожному місці розташування пікселя генерується на основі пікселя у відповідному місці та сусідніх пікселів у вхідному зображенні. Фільтри зазвичай поділяються на два типи: лінійні та нелінійні.

Лінійний фільтр поєднує значення пікселів лінійним чином за допомогою згортки. Вихідний сигнал у кожному місці пікселя генерується шляхом згортки маски або ядра фільтра з пікселем у відповідному місці та сусідніми пікселями на вхідному зображенні. Маска або ядро фільтра визначає розмір і форму області вхідного зображення, яка буде використовуватися для кожного обчислення, а також вагові коефіцієнти окремих пікселів. Операція згортки така:

$$y(i,j) = \sum_m \sum_n x(i-m, j-n)k(m,n),$$

де y – вихід, x – вхідне зображення, k – матриця ядра фільтра, а m та n – індекси рядка та стовпця ядра фільтра відповідно, з центром ядра, позначеним як (0,0). Зауважте, що вхідне зображення доповнюється нулями для збереження розмірності вихідного зображення.

На відміну від операції згортки в CNN, яка насправді є операцією крос-кореляції, ядро фільтра операції згортки повертається на 180 навколо свого центрального елемента, звідси знак мінус у x ($i - m, j - n$), перед виконанням множення та підсумовування. Однак, якщо ядро фільтра симетричне, обертання можна пропустити.

Існують різні типи лінійних фільтрів. Двома найпоширенішими лінійними фільтрами, що використовуються в попередній обробці даних для нейронних мереж, є згладжувальний та різницевий фільтри.

Згладжувальний фільтр обчислює середньозважене значення пікселів вхідного зображення в області ядра фільтра та створює зображення зі згладженими краями, як показано на рисунку 3.4 . Цей фільтр складається лише з позитивних вагових коефіцієнтів.

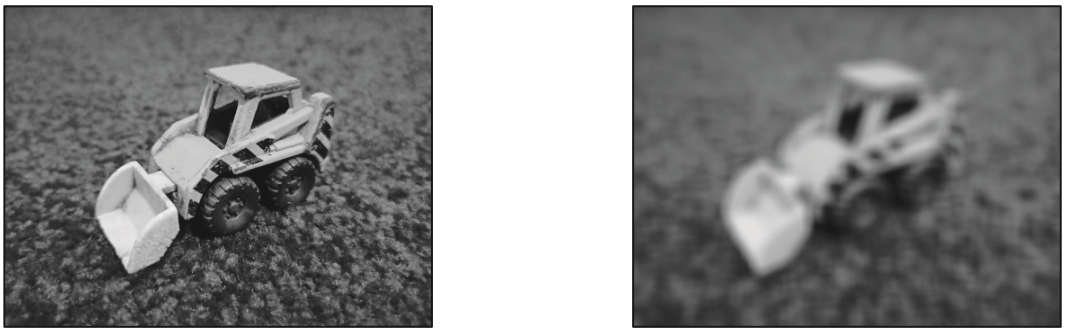


Рисунок 2.4 – Приклад застосування згладжувального фільтра. Вхідне зображення (ліворуч) та вихід згладжувального фільтра (праворуч)

Найпростіший згладжувальний фільтр називається коробчастим фільтром. Він називається коробчастим фільтром, оскільки його форма схожа на коробку. Фільтр має позитивні вагові коефіцієнти з однаковими значеннями. Приклад коробчастого фільтра показано на рисунку 2.5 .

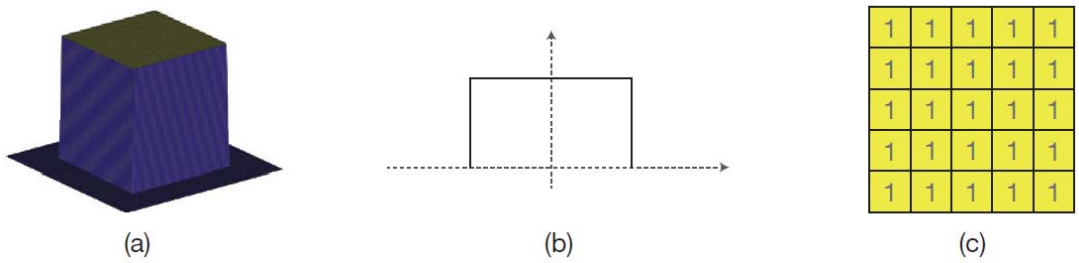


Рисунок 2.5 – Приклад коробчастого фільтра. (a) 3D-ілюстрація; (b) Профіль; (c) Апроксимації неперервної функції у (a) за допомогою дискретних матриць фільтрів

Одним із застосувань блокового фільтра є його використання як усереднюючого фільтра. Встановивши всі коефіцієнти фільтра рівними 1 та поділивши результат згортки на суму коефіцієнтів фільтра, вихідний результат буде дорівнювати середньому значенню значень пікселів у області фільтра.

Оскільки коробковий фільтр має різкі обрізи по краях, він створює перехідні процеси або сильні ефекти «дзвінка» в частотній області. Щоб зменшити цей небажаний ефект, як згладжувальний фільтр кращий варіант, оскільки він «добре поводиться» в частотній області, оскільки фільтр більше акцентує увагу на пікселях у центрі та менше на віддалених пікселях, як показано на рисунку 3.6 .

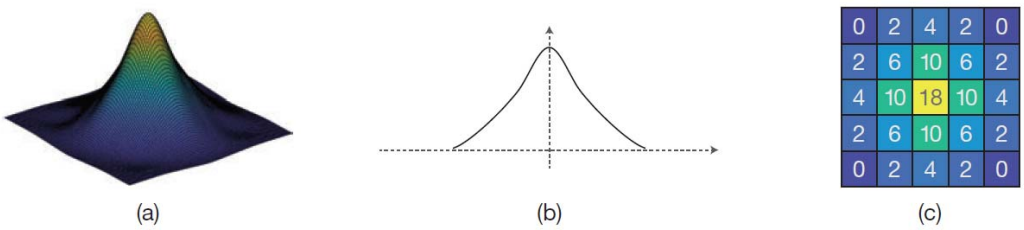


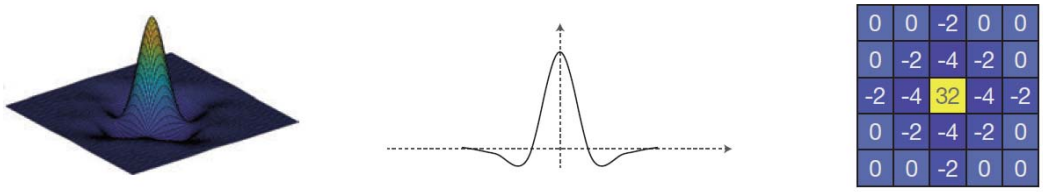
Рисунок 2.6 – Приклад гаусового фільтра. (a) 3D-ілюстрація; (b) Профіль; (c) Апроксимації неперервної функції у (a) за допомогою матриць дискретних фільтрів

Коефіцієнти двовимірного гаусового фільтра можна визначити за формулою:

$$K(m,n) = e^{-\frac{m^2+n^2}{2\sigma^2}},$$

де σ – стандартне відхилення дзвоноподібної функції, m та n – кількість рядків та стовпців від центру фільтра відповідно.

Різницевий фільтр, наприклад, фільтр «Лапласа» або «Мексиканський капелюх», – це ще один тип лінійного фільтра, який обчислює зважену різницю між центром та навколишніми пікселями. Зважена різниця виконується за допомогою позитивних коефіцієнтів у центрі фільтра та негативних коефіцієнтів навколо центру, або навпаки, як показано на рисунку 3.7 [73].



а б в

Рисунок 2.7 – Приклад лапласіанського фільтра. а - 3D-ілюстрація; б - Профіль; в - Апроксимації неперервної функції у а- за допомогою дискретних матриць фільтрів

Цей тип фільтра підсилює локальні розриви інтенсивності та зменшує області з повільно змінними рівнями інтенсивності, що призводить до світлих крайових ліній та темного фону. Тому його зазвичай використовують для виявлення країв, як показано на рисунку 2.8 .

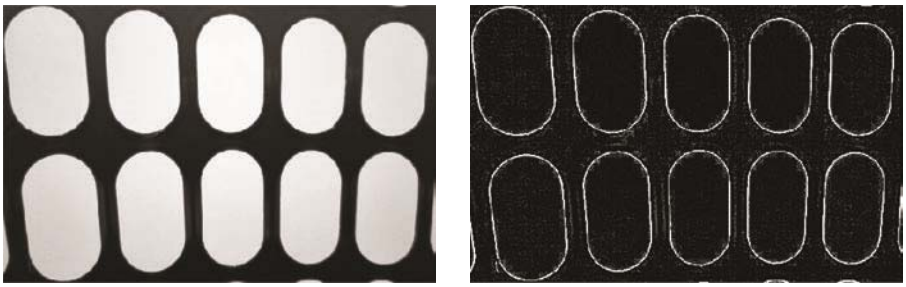


Рисунок 2.8 – Приклад застосування лапласіанського фільтра для виявлення країв. Вхідне зображення (ліворуч) та вихід лапласіанського фільтра (праворуч).

Подібно до лінійних фільтрів, нелінійні фільтри генерують кожен вихідний піксель з вхідних пікселів в області ядра фільтра. Однак замість використання лінійної функції для генерації вихідного сигналу використовуються нелінійні функції. Нелінійні фільтри зазвичай застосовуються для видалення імпульсного шуму, який виглядає як чорні та білі точки, розкидані по зображенню.

Нелінійний двовимірний фільтр це метод обробки зображень, який працює без використання класичних лінійних операцій згортки. Він застосовується безпосередньо до пікселів у двовимірному просторі (матриці зображення) і враховує локальні особливості, але результат не є простою лінійною комбінацією сусідніх значень. Отже, нелінійний двовимірний фільтр це інструмент цифрової обробки зображень, який дозволяє ефективно видаляти шум і зберігати важливі структури (краї, контури), що робить його незамінним у комп’ютерному зорі та медичній візуалізації.

Адаптивні фільтри враховують локальну статистику (наприклад, дисперсію) і змінюють спосіб згладжування залежно від структури області. Вони мають стійкість до шуму та краще зберігають краї та деталі, ніж лінійні фільтри.Завдяки адаптивності фільтри можуть змінювати поведінку залежно від локальних характеристик зображення.

Найпростішими нелінійними фільтрами є мінімальний та максимальний фільтри.

Мінімальний фільтр призначає кожному вихідному пікселю мінімальне значення пікселя з вхідних пікселів в області ядра фільтра, як визначено за формулою

$$y(i,j) = \min\{ x(i+m, j+n) \mid (m,n) \in R \},$$

де y позначає вихідний сигнал, x позначає вхідний сигнал, а R позначає область ядра фільтра. Зверніть увагу, що вхідний сигнал доповнюється постійними значеннями для збереження розмірності вихідного сигналу.

Максимальний фільтр призначає кожному вихідному пікселю максимальне значення пікселя вхідного сигналу в області ядра фільтра [73], як визначено за формулою

$$y(i,j) = \max\{ x(i+m, j+n) \mid (m,n) \in R \}.$$

Іншим типом нелінійного фільтра є медіанний фільтр. Медіанний фільтр призначає кожному вихідному пікселю значення медіанного пікселя вхідного сигналу в області ядра фільтра, як визначено за формулою,

$$y(i,j) = \text{медіана}\{ x(i+m, j+n) \mid (m,n) \in R \}.$$

Медіанний піксель обчислюється шляхом сортування значень пікселів в області ядра фільтра у зростаючій послідовності, а потім вибору середнього значення (для непарної кількості елементів) або середнього значення двох середніх значень (для парної кількості елементів).

2.4 Обробка вихідних даних

Післяобробка даних – це процес, який перетворює вихідні дані в потрібний формат. Його також можна використовувати для покращення або очищення вихідних даних для досягнення кращої загальної продуктивності. Процес часто включає попередні знання про бажаний результат, такі як форма та мінімальний розмір цільового класу в сегментації.

Найфундаментальніші процеси постобробки даних для сегментації зображень включають математичну морфологію та операції маркування зв'язних компонентів.

2.5 Метод оцінювання

Під час реалізації нейронної мережі, після розробки моделі, важливо оцінити її, щоб переконатися, що вона здатна до узагальнення.

Методи перехресної перевірки та витримки – це методи оцінки моделі, які зазвичай застосовуються для оцінки здатності до узагальнення стосовно показника продуктивності, такого як функція перехресної ентропії .

Під час перехресної валідації набір даних часто розділяється на дві підмножини, тобто навчальний та тестовий набір. Зазвичай для розділення даних на навчальний та тестовий набори відповідно використовується співвідношення 80% до 20%. Навчальний набір використовується для навчання моделі, тоді як тестовий набір використовується для оцінки продуктивності моделі на невидимому наборі даних. Якщо обсяг даних достатній, навчальний набір можна додатково розділити на навчальний та валідаційний набори у співвідношенні 75% до 25%. Валідаційний набір можна використовувати для оцінки моделі на невидимому наборі даних під час навчання для налаштування гіперпараметрів та вибору моделі.

Розподіл даних у методі перехресної перевірки зазвичай виконується багаторазово, а результати перевірки усереднюються за кількістю повторень. Залежно від способу розподілу даних для перевірки, існує два основних типи перехресної перевірки, тобто вичерпна та невичерпна.

У методі вичерпної перехресної перевірки фіксована кількість зразків даних використовується як тестовий набір, а решта зразків даних використовуються як навчальний набір для кожного раунду перевірки. Це повторюється $\frac{n!}{p!(n-p)!}$ кілька разів, тобто для кожного можливого розділення набору даних, де n позначає загальну кількість зразків даних у наборі даних, а p позначає кількість зразків даних, які будуть використані як тестовий набір. Цей метод отримав назву відповідно до кількості зразків даних, що використовуються як тестовий набір, наприклад, перехресна перевірка з

виключенням p для p > 1 та перехресна перевірка з виключенням одного для p = 1. Основним недоліком цього методу є те, що він може бути дуже повільним.

У методі невичерпної перехресної перевірки набір даних розбивається на певні розділи, тобто складки, і враховуються не всі можливі розділи. Метод невичерпної перехресної перевірки включає k- кратну перехресну перевірку. У k -кратній перехресній перевірці набір даних розбивається на k складок однакового розміру (k > 1), де одна складка даних використовується як тестовий набір, а k −1 складок даних – як навчальний набір. Метод перевірки повторюється k разів, причому кожна складка даних є тестовим набором лише один раз. Загальні значення для k – 3, 5 та 10.

У методі hold out дані розподіляються, як у методі k -кратної перехресної перевірки, але перевірка даних виконується лише один раз замість k разів для оцінки продуктивності моделі. Основним недоліком цього методу є те, що не всі вибірки даних будуть у тестовому наборі.

Метод k-кратної перехресної перевірки є найбільш застосовним, оскільки він займає достатньо часу для виконання та дає краще уявлення про ефективність узагальнення моделі, ніж метод перевірки з витримкою.

3 СЕГМЕНТАЦІЯ ЗОБРАЖЕНЬ З ВИКОРИСТАННЯМ НЕЙРОННИХ МЕРЕЖ

Нейронні мережі – це підмножина алгоритмів машинного навчання, які є аналогом нейронних мереж в біологічному мозку. Важливим є тип нейронної мережі, включаючи її компоненти, структуру та механізм навчання. Існують різні CNN, типу нейронної мережі, спеціалізованих для високовимірних даних. Важливим є вибір сучасних архітектур CNN та їх архітектури з точки зору їхньої структури, кількості параметрів, що навчаються, та компонентів. Важливими є основні елементи нейронної мережі, а також її базові конфігурації та механізм навчання.

3.1 Основи нейронних мереж

Нейронна мережа, яку іноді називають штучною нейронною мережею, — це система, що складається зі штучних нейронів, призначених для моделювання роботи людського мозку, наприклад, навчання певних знань та їх зберігання для виконання певного завдання, наприклад, розпізнавання образів.

Штучний нейрон моделюється трьома основними елементами: набором синаптичних ваг w , суматором, , та функцією активації ϕ(·), як показано на рисунку 2.1 , де n – розмірність вхідних даних, $x_1,...,x_n$ – вхідні дані, wk1 , ... , wkn – синаптичні ваги нейрона k відносно вхідних даних, z_k – потенціал активації нейрона k , b_k – зміщення нейрона k , ϕ (·) – функція активації, а a_k – вихідна реакція нейрона k .

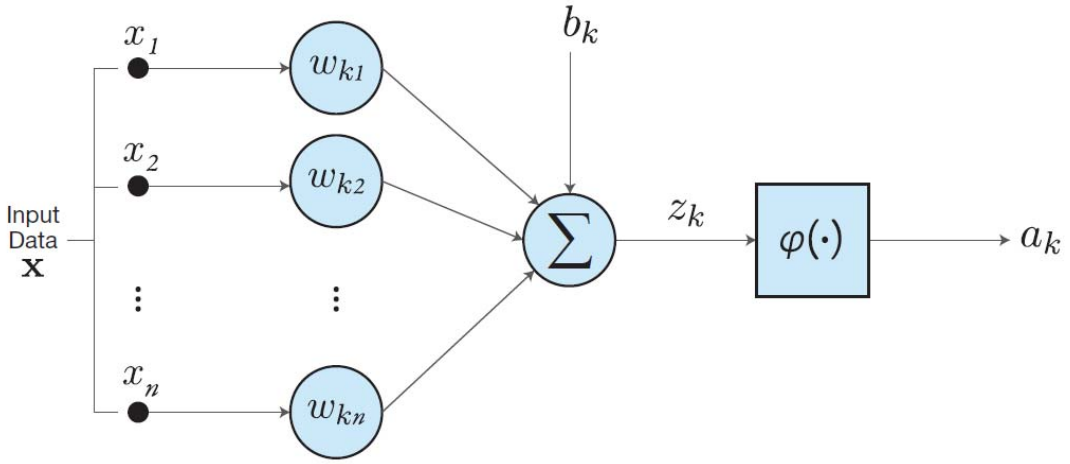


Рисунок 3.1 – Модель штучного нейрона

У математичних термінах вихід нейрона можна описати так:

$$z_k = \sum_{j=1}^{n} w_{kj} x_j + b_k,$$

та $a_k = \varphi(z_k)$.

Це можна записати більш загально так,

$$h_k(x) = \varphi((\sum_{j=1}^{n} w_{kj} x_j) + b_k).$$

Функція активації $\varphi(\cdot)$ використовується для визначення вихідного сигналу нейрона через потенціал активації z. Вона визначає стан активації та вихідне значення нейрона. Існує кілька типів поширених функцій активації.

Порогова функція: видає значення 0, якщо потенціал активації нейрона негативний, і 1, якщо він позитивний. Вона визначається як:

$$\varphi(z) = \begin{cases} 1, коли \quad z \geq 0, \\ 0, коли \quad z < 0. \end{cases}$$

Кусочно-лінійна функція: створює лінійний вихідний сигнал, коли потенціал активації нейрона знаходиться в межах певної області, інакше буде створено значення насичення 0 або 1. Вона визначається як,

$$\varphi(z) = \begin{cases} 1, & z \geq +\frac{1}{2} \\ z, & +\frac{1}{2} > z > -\frac{1}{2} \\ 0, & z \leq -\frac{1}{2}. \end{cases}$$

Сигмоїдна функція: диференційована S-подібна функція, яка створює неперервний діапазон значень від 0 до 1. Вона визначається як,

$$\varphi(z) = \frac{1}{1+e^{-z}}.$$

Нейронна мережа складається з певної кількості штучних нейронів, що утворюють структуру, яку можна використовувати як потужний

обчислювальний інструмент для вирішення складних задач. Вона має здатність узагальнювати, тобто робити розумні прогнози для нових вхідних даних тієї ж категорії (класу), що й вивчені дані, які мережа ніколи не бачила.

Навчання нейронної мережі вимагає рандомізованого набору позначених даних. Вона навчається, коригуючи свої синаптичні ваги, щоб генерувати бажані відповіді, що відповідають кожним унікальним вхідним даним. Головною метою навчання є мінімізація похибки між відповіддю мережі та міткою, пов'язаною з кожними вхідними даними. Функція вартості формується на основі деякої міри похибки для кожного виходу.

Функція вартості залежить від завдання, тобто чи воно призначене для регресії, класифікації чи інших цілей. Наприклад, функція вартості, яку можна використовувати для регресії, — це функція середньоквадратичної, яка задається як

$$J(\mathbf{w}, \mathbf{b}) = \frac{1}{2m} \sum_{i=1}^{m} \sum_{k=1}^{K} \left(h_k \left(\mathbf{x}^{(i)} \right) - y_k^{(i)} \right)^2,$$

де J (w , b) позначає вартість, w позначає синаптичні ваги, а b позначає зміщення, K позначає кількість нейронів, m позначає кількість вибірок даних, x (i) позначає i -ту вибірку даних з набору даних, $h_k(\mathbf{x}^{(i)})$ позначає вихідну відповідь мережі на дані x (i) на нейроні k , y (i) позначає мітку даних x (i) .

Функція вартості, яка зазвичай використовується для задач класифікації. Це функція перехресної ентропії, що задається формулою:

$$J(\mathbf{w}, \mathbf{b}) = -\frac{1}{m} \left[\sum_{i=1}^{m} \sum_{k=1}^{K} y_k^{(i)} \log \left(h_k \left(\mathbf{x}^{(i)} \right) \right) + \left(1 - y_k^{(i)} \right) \log \left(1 - h_k \left(\mathbf{x}^{(i)} \right) \right) \right]$$

Мережа навчається за допомогою алгоритму оптимізації для розв'язання задачі argmin J (w , b) .w , b

3.2 Персептрон

Перцептрон – це найпростіша форма нейронної мережі, яка складається з одного штучного нейрона з набором регульованих синаптичних ваг та зміщення, як показано на рисунку 2.1 . Перцептрон навчається шляхом

коригування синаптичних ваг та зміщення відповідно до вибірок навчальних даних, щоб він міг виробляти бажані відповіді.

Згідно з теоремою про збіжність персептрона, персептрон гарантовано знаходить оптимальне рішення у вигляді гіперплощини, яка діє як межа прийняття рішення в задачі класифікації шаблонів з двох класів, як показано на рисунку 3.2.

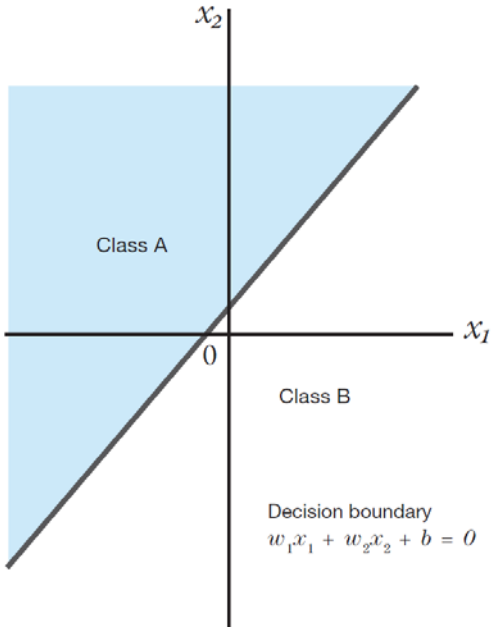


Рисунок 3.2 – Персептрон як межа прийняття рішення для двовимірних (2D) лінійно роздільних даних.

Рівняння цієї гіперплощини задається рівнянням 2.1 з k = 1. Вихідна характеристика перцептрона [34] задається таким чином:

$$\mathrm{h}(x) = \mathrm{sgn}(z) = \begin{cases} +1, коли \quad z > 0 \\ \\ -1, коли \quad z < 0 \end{cases}.$$

Щоб розширити мережу та сформувати класифікатор більш ніж двох класів, необхідно додати потрібні перцептрони. Зауважте, що sgn(·) – це функція активації, відома як функція сигнума.

3.3 Багатошаровий персептрон

Багатошаровий персептрон (MLP) – це тип нейронної мережі, яка побудована з використанням кількох персептронів, що утворюють

багатошарову мережу. Вона складається з вхідного шару, одного (або кількох) прихованих шарів та вихідного шару, як показано на рисунку 2.3 .

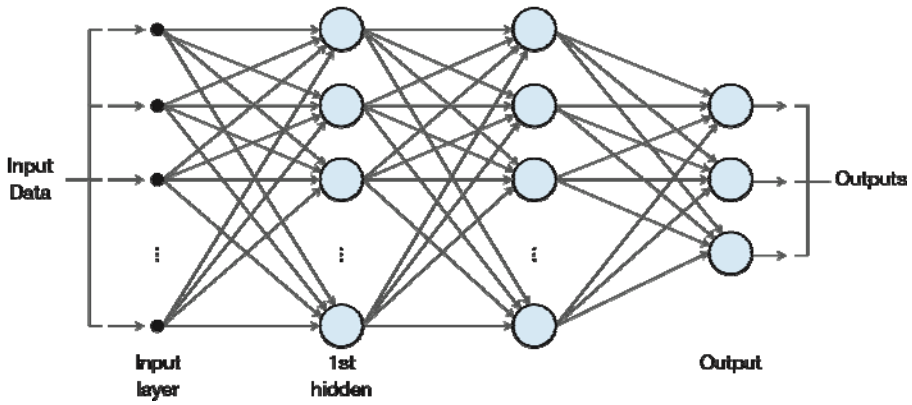


Рисунок 3.3 – Приклад багатошарового персептрона з двома прихованими шарами

Навчання MLP базується на правилі навчання з корекцією помилок, відомому як алгоритм зворотного поширення помилки. Цей алгоритм складається з прямого та зворотного проходу.

При прямому проходженні вхідні дані подаються на вхідний шар, після чого відбувається пряме поширення через приховані шари з фіксованими синаптичними вагами до вихідного шару для створення набору вихідних даних. Це поширення ілюструється на прикладі MLP з двома прихованими шарами, показаному на рисунку 2.3 з одним зразком даних.

У зворотному проході навчання відбувається, коли синаптичні ваги коригуються для мінімізації похибки. Воно починається з обчислення похибки фактичних вихідних відповідей мережі, з прямого проходу, з мітками відповідних вхідних зразків. Ця похибка потім поширюється назад шар за шаром через мережу, доки не досягне вхідного шару. Для зображення цього процесу буде використано структуру нейронної мережі на рисунку 2.3 та відповідні позначення.

Зрештою, синаптичні ваги та зміщення мереж можна ітеративно оновлювати за допомогою алгоритму оптимізації, такого як пакетний градієнтний спуск, який використовує всі зразки даних у навчальному наборі

3.4 Згорткова нейронна мережа

Згорткова нейронна мережа (ЗНМ) – це тип глибокої нейронної мережі, що спеціалізується на, але не обмежується, завданнями обробки зображень. Основні особливості полягають у тому, що ЗНМ зазвичай складається з більш ніж двох прихованих шарів та використовує розподіл ваги. Використання розподілу ваги в CNN призводить до того, що вона має значно меншу кількість параметрів для навчання та потребує менше часу на навчання порівняно з MLP такої ж глибини.

Оригінальна реалізація CNN складається з низки згорткових, об'єднуючих (субвибіркових) та повністю зв'язаних шарів. CNN, по суті, використовує методи обробки сигналів для автоматичного вилучення ознак. Згорткові шари містять згорткові фільтри з різними коефіцієнтами для створення різних ознак еквівалентності перекладання. Об'єднуючі шари містять нелінійні фільтри для вилучення найважливіших ознак у незмінній до перекладання структурі.

ЗГЗ використовує згорткові та об'єднуючі шари на початку своєї архітектури для вилучення ознак вхідних зображень. Потім повністю зв'язані шари, які є стандартним одновимірним вхідним MLP, використовуються для прогнозування виходу вхідного зображення. Приклад архітектури ЗГЗ показано на рисунку 3.4 .

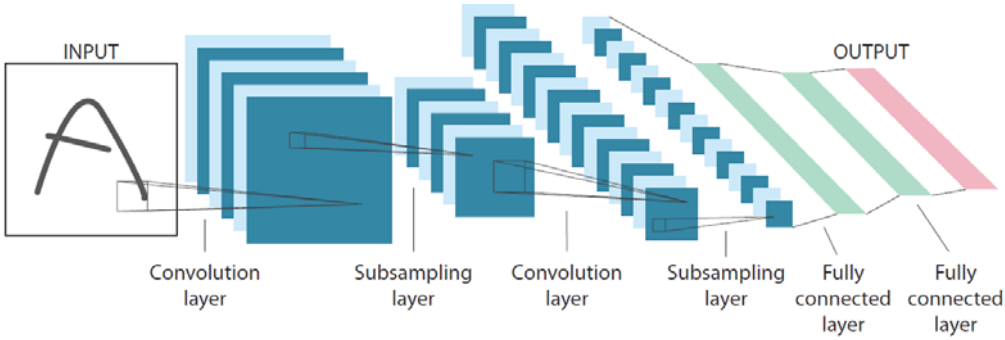


Рисунок 3.4 – Приклад архітектури CNN

3.5 Структури та параметри CNN

Існують різні структури та кількість параметрів, що використовуються в CNN для навчання.

Оригінальна структура CNN більше підходить для задачі класифікації, де вона використовує згорткові шари як екстрактори ознак та повністю зв'язані шари як кінцевий класифікатор. CNN також може бути структурована для виконання інших завдань. Найпоширенішими структурами CNN є стискаючі, стискаючо-розширювальні та багатоетапні структури.

ЗНС, розроблена для задачі класифікації, зазвичай використовує структуру скорочення, де роздільна здатність карт ознак зменшується в міру заглиблення мережі, де карти ознак з найнижчою роздільною здатністю є вхідними даними для повністю зв'язаних шарів. ЗНС, що використовується для класифікації, зазвичай вимагає, щоб ознаки були високоінваріантними для прийняття остаточного рішення. Структура включає методи або компоненти, які можуть допомогти інтегрувати просторово інваріантні властивості у високо рівневі особливості.

ЗНС, розроблена для задачі сегментації, зазвичай використовує структуру, що скорочується та розширюється. Частина структури, що скорочується, використовується для вилучення ознак, тоді як частина, що розширюється, використовується для високорівневого відображення ознак на початкову вхідну роздільну здатність для попіксельних прогнозів. ЗНС, що використовується для сегментації, зазвичай вимагає, щоб ознаки були високо еквівалентними, тобто збереженої просторової інформації. Структура зазвичай включає методи або компоненти, які можуть допомогти зберегти просторову інформацію ознак.

ЗНС, розроблена для складнішого завдання, такого як завдання виявлення об'єктів, іноді використовує кілька алгоритмів або мереж для обробки різних операцій завдання. Кожен алгоритм зазвичай розроблений та оптимізований для певної операції в рамках завдання, щоб досягти вищої загальної продуктивності порівняно з продуктивністю одного алгоритму, розробленого для повного вирішення завдання. Наприклад, ЗНС алгоритму вилучення області інтересу та мережі класифікації зазвичай призводить до кращої продуктивності класифікації порівняно з однією мережею класифікації.

Кількість параметрів, що використовуються в CNN, що навчаються, залежить від трьох факторів: глибини, розташування компонентів та кількості фільтрів, що використовуються в кожному згортковому шарі.

Глибока згорткова нейронна мережа (ЗНС), дозволяє ефективно спостерігати за більшою площею вхідного зображення на відміну від поверхневої ЗНС. Зі збільшенням глибини мережі встановлюється краще розуміння просторового зв'язку пікселя (або ознаки) та його оточення. Однак збільшення глибини мережі зазвичай призводить до збільшення кількості параметрів, що навчаються.

Згорткова глибока мережа (ЗГЗ) зазвичай будується з набору різних компонентів. Залежно від конструкції, ЗГЗ може мати значно менше параметрів, що навчаються, ніж інша ЗГЗ, побудована з того ж набору компонентів. Наприклад, ЗГЗ, побудована з двох згорткових шарів 3×3×256, складених разом, а потім двох згорткових шарів 1×1×1, має значно більшу кількість параметрів, що навчаються, порівняно з ЗГЗ, побудованою з тих самих компонентів, розташованих по черзі, тобто двох пар згорткових шарів 3 × 3 × 256 та згорткового шару 1 × 1 × 1, складених разом. Таким чином, кількість параметрів, що навчаються, що використовуються в ЗГЗ, залежить не лише від використовуваних компонентів, але й від розташування компонентів.

Як обговорювалося раніше, ЗГН використовує згорткові шари для вилучення ознак. Кожен згортковий шар складається з певної кількості фільтрів, які визначають тип ознак, що вилучаються з вхідних даних. Чим більша кількість фільтрів у кожному згортковому шарі, тим більша здатність мережі вирішувати завдання. Однак, чим більша кількість фільтрів, тим більша кількість параметрів, що навчаються, в мережі.

Таким чином, глибина, розташування компонентів та кількість фільтрів, що використовуються в кожному згортковому шарі, повинні ретельно регулюватися користувачем відповідно до завдання та доступних ресурсів, оскільки це може призвести до дорогого використання пам'яті та збільшення часу обчислень.

3.6 Компоненти CNN

У цьому розділі ми представимо компоненти, що використовуються для побудови CNN. Зокрема, розглянуті компоненти включають згортковий шар, шар об'єднання, шар підвищеної дискретизації, пропуск з'єднань, шар випадання, нормалізацію та активаційний шар.

Згортковий шар використовує набір згорткових фільтрів для виконання операцій зі створення карт ознак залежно від ядра фільтра. Форма ядра визначає форму та розмір вхідної області, тоді як значення ядра визначають ваги на вході. Форма ядра зазвичай заздалегідь визначена в архітектурі мережі, тоді як значення ядра ініціалізуються відповідно до алгоритмів ініціалізації, та коригуються під час процесу навчання. Отримані карти ознак є ознаками вхідних даних, такими як градієнти (ребра).

У згорткових нейронних мережах операцію згортки вхідного зображення та ядра згорткового фільтра можна виразити наступним чином:

$$y(i,j) = \sum_m \sum_n x(i+m, j+n) k(m,n) ,$$

де y – матриця вихідного зображення, x – матриця вхідного зображення, k – матриця ядра фільтра, а m та n – індекси рядка та стовпця ядра фільтра відповідно, причому центр ядра позначено як (0,0).

У згорткових мережах згортковий шар може бути реалізований за допомогою чотирьох гіперпараметрів, тобто кількості фільтрів, розміру ядра, довжини кроку та доповнення. Кількість фільтрів, яка відповідає кількості карт ознак, створених у кожному шарі, визначає ємність мережі. Розмір ядра визначає кількість вхідних даних у кожному напрямку, які обробляються одночасно. Довжина кроку визначає кількість рядків і стовпців, на які зміщується фільтр після кожного обчислення. Доповнення стосується операції додавання нулів по периметру вхідних даних для збереження роздільної здатності шару на виході. Застосування доповнення в згортковому шарі, з доповненням чи без нього, можна визначити за допомогою гіперпараметра доповнення. Згортковий шар без заздалегідь визначеної довжини кроку та доповнення зазвичай стосується згорткової операції з довжиною кроку 1 та застосуванням доповнення.

Ілюстрацію операції згортки 3 × 3 з довжиною кроку 1 та без доповнення показано на рисунку 3.5 , де вхідні дані згортаються за допомогою фільтра згортки 3 × 3 для отримання вихідних даних, тобто карти ознак.

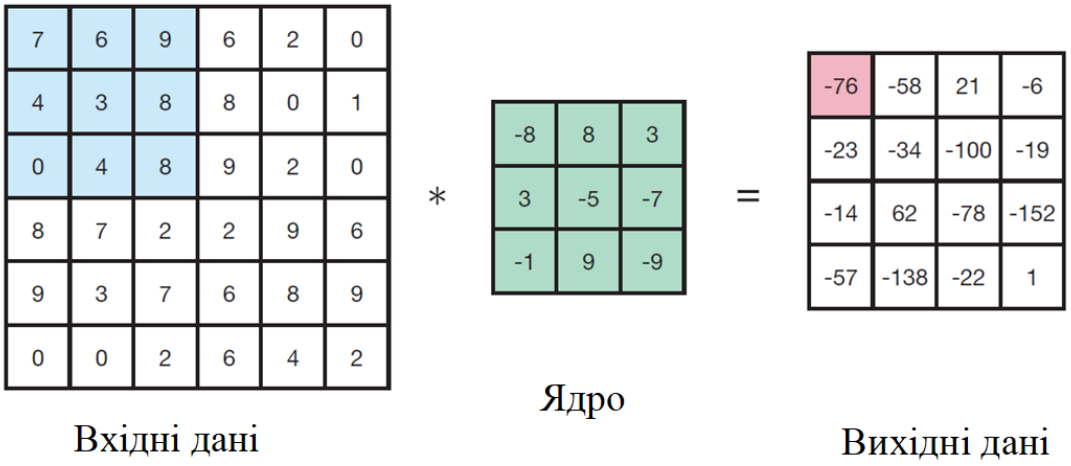


Рисунок 3.5 – Приклад операції згортки 3 × 3 з довжиною кроку 1 та без доповнення

Шари об'єднання використовуються для зменшення розмірності вхідних даних та введення трансляційних інваріантностей у мережу за допомогою процесу зниження частоти дискретизації. Найпоширенішим типом шару об'єднання, що використовується в CNN, є шар максимального об'єднання. Це тип нелінійного фільтра, який використовується для вилучення значущих ознак з попереднього шару шляхом вибору найбільш цінної ознаки з кожної області ядра ковзного фільтра. Шари об'єднання на основі лінійних фільтрів, такі як усереднений шар об'єднання або згортковий шар зі сходами (згортковий шар з довжиною кроку більше 1), також можуть використовуватися замість або разом із шарами максимального об'єднання в певних застосуваннях.

У звичайних нейронних мережах (CNN) шар об'єднання реалізовано з трьома гіперпараметрами, тобто розміром ядра, довжиною кроку та доповненням. Як правило, шар об'єднання 2 × 2 з довжиною кроку 2 та без доповнення використовується в CNN для виконання зниження дискретизації в 2 рази. Ілюстрація операцій об'єднання 2 × 2 max, average та min показано на рисунку 2.6 , де відповідно витягуються максимальне, середнє та мінімальне значення ознаки з кожної області ядра ковзного фільтра.

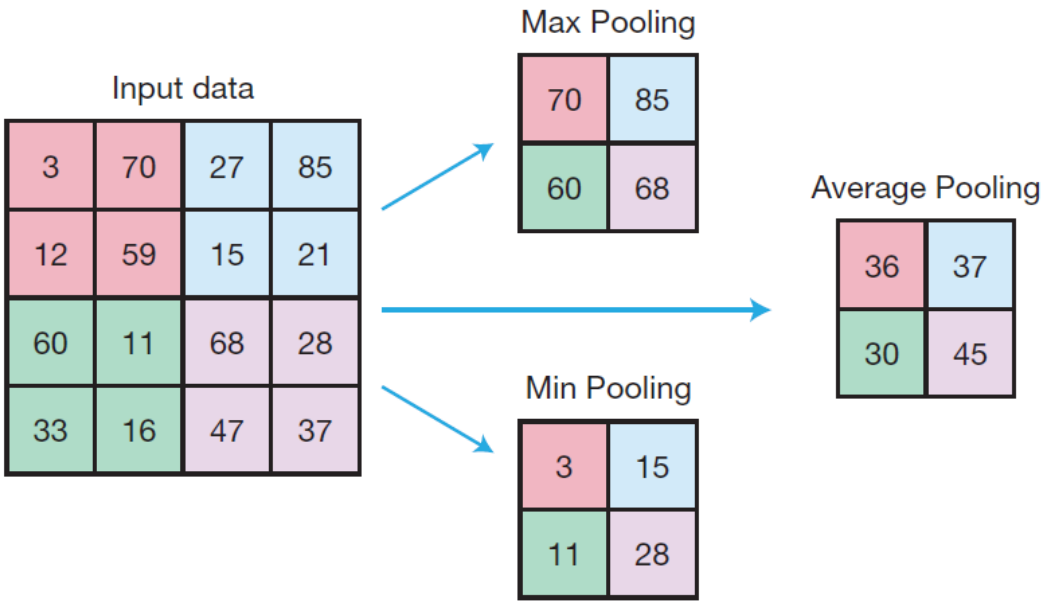


Рисунок 3.6 – Приклади операцій об'єднання 2 × 2 max, average та min

Шари підвищеної дискретизації використовуються для збільшення розмірності вхідних даних. У цифровій обробці зображень підвищена дискретизація традиційно виконується за допомогою методів інтерполяції. Інтерполяція – це метод оцінки, який використовується для отримання значення точки даних на основі значень точок даних навколо неї.

Найпростішим методом підвищення дискретизації є інтерполяція найближчого сусіда. Метод найближчого сусіда вибирає найближчу точку даних до точки даних, яку потрібно оцінити, а потім використовує її значення як оціночне значення без урахування значень інших точок. Однак інтерполяція найближчого сусіда зазвичай призводить до створення чітких меж.

Іншим поширеним методом інтерполяції є білінійна інтерполяція. Вона передбачає виконання лінійної інтерполяції у двох напрямках (наприклад, горизонтальному та вертикальному). Для оцінки необхідної точки даних використовується середньозважене значення кількох заздалегідь визначених сусідніх точок даних. Отримане зображення має більш гладкі краї порівняно з методом інтерполяції найближчих сусідів.

У звичайних нейронних мережах (CNN) шар підвищеної дискретизації на основі інтерполяції реалізовано з одним гіперпараметром, тобто розміром. Розмір визначає коефіцієнт підвищеної дискретизації для інтерполяції, яка має

бути виконана в кожному напрямку. Приклади операцій найближчого сусіда та білінійної інтерполяції розміром 2 × 2 показано на рисунку 3.7 .

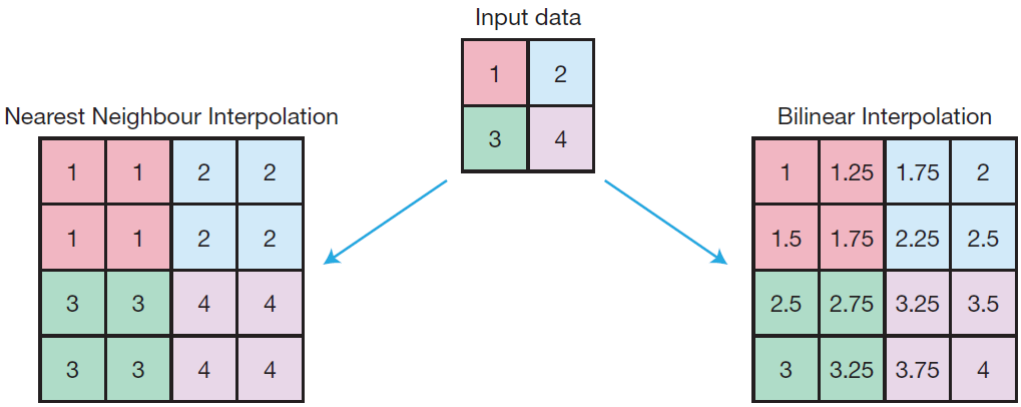


Рисунок 3.7 – Приклад операцій інтерполяції 2 × 2 найближчого сусіда та білінійної інтерполяції

Транспонована згортка – це метод підвищеної дискретизації, який можна навчати. На відміну від методів підвищеної дискретизації на основі інтерполяції (наприклад, метод найближчого сусіда, білінійна інтерполяція), транспонована згортка дозволяє навчати свій параметр разом з іншими параметрами мережі шляхом зворотного поширення. Транспонована згортка виконує підвищену дискретизацію з коефіцієнтом f , виконуючи згортку з дробовим вхідним кроком f1 , де f – ціле число. Для виконання нелінійної підвищеної дискретизації можна використовувати комбінацію транспонованого згорткового шару та нелінійної функції активації.

У згорткових мережах гіперпараметри, необхідні для транспонованого згорткового шару, подібні до згорткового шару. Транспонований згортковий шар без заздалегідь визначеної довжини кроку та відступів зазвичай відноситься до транспонованого згорткового шару з довжиною кроку 1 та без відступів.

Ще один популярний метод підвищення дискретизації, який можна навчати, називається згорткою зі зміною розміру.

Він використовує комбінацію інтерполяційних та згорткових шарів у вигляді стеку з інтерполяції найближчих сусідів розміром f × f та згорткових шарів розміром f × f , де f позначає коефіцієнт підвищеної дискретизації.

для виконання збільшення дискретизації в 2 рази кращим є використання згортки зі зміною розміру 2×2 або транспонованої згортки.

Глибокі згорткові мережі (CNN) виявилися дуже успішними у вилученні важливих ознак за допомогою згорткових шарів. Однак глибокі мережі схильні до проблеми деградації, коли точність навчання мережі зростає, насичується, а потім швидко знижується після того, як мережа починає збіжатися.

Пропускні з'єднання спочатку використовувалися для включення фреймворку залишкового навчання для вирішення проблеми деградації в глибоких мережах. Фреймворк навчання використовує пропускне з'єднання з шару до пізнішого шару $\ell+n$, щоб замінити вихідна функція $F(\boldsymbol{a}^{(\ell)})$ з функцією $H(\boldsymbol{a}^{(\ell)}) = F(\boldsymbol{a}^{(\ell)}) + \boldsymbol{a}^{(\ell)}$, як показано на рис. 3.8.

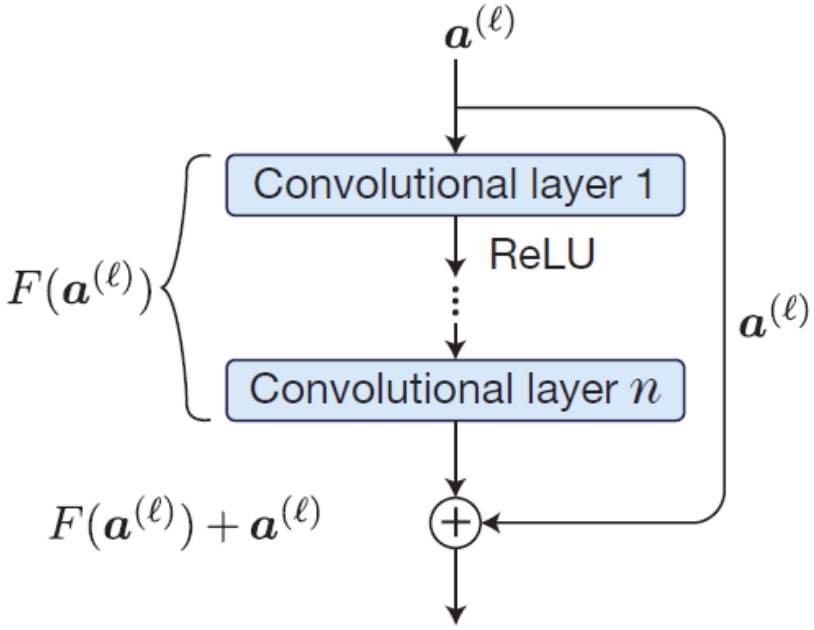


Рисунок 3.8 – Приклад реалізації пропускного з'єднання для включення фреймворку залишкового навчання

Причиною використання фреймворку навчання залишків є запобігання більшим помилкам навчання глибокої мережі, ніж відповідної поверхневої мережі, шляхом спрощення навчання будь-яких надлишкових шарів у глибокій мережі, наприклад, n згорткових шарів на рисунку 3.8 , для одиничного відображення. Використовуючи навчання залишків, розв'язувачі можуть спростити апроксимацію одиничних відображень, примусово наближаючи ваги функції залишку, $F(\boldsymbol{a}^{(\ell)})$, до нуля, якщо одиничні відображення є оптимальними. Це допоможе мінімізувати додаткову складність, що вноситься будь-якими

надлишковими нелінійними шарами в глибокій мережі, коли оптимальне рішення для завдання може бути досягнуто за допомогою відповідної поверхневої мережі. Шар поелементного підсумовування використовується в кінці пропускного з'єднання, таким чином зберігаючи розмірність вихідного шару фіксованою, що не додає ні додаткових параметрів, ні обчислювальної складності.

Пропуск з'єднань також можна використовувати для об'єднання ознак одного шару з іншим за допомогою операції конкатенації. Ідея полягає у використанні інформації попереднього шару в пізнішому шарі для досягнення кращої продуктивності. Його також можна використовувати в поєднанні з підвищеною дискретизацією для об'єднання підвищених прогнозів для формування точного результату попіксельної сегментації.

Видалення нейронів – це метод регуляризації, який використовується на етапі навчання для запобігання перенавчанню. Ідея полягає в тому, щоб навчати різні моделі та використовувати середнє значення прогнозів для покращення узагальнення мережі. Операція виконується шляхом випадкового видалення нейронів з визначеним коефіцієнтом випадання, pd , під час навчання, таким чином ваги мережі налаштовуються на основі різної зв'язності нейронів у мережі. Зауважте, що коефіцієнт випадання – це значення від 0 до 1. Під час тестування ваги кожного нейрона множаться на ймовірність утримання, $pr = 1 - pd$, для наближення середніх прогнозів різних навчених моделей. Цей метод може покращити результати навчання нейронних мереж.

Приклад застосування відсіву в нейронній мережі показано на рис. 3.9 (стандартна нейронна мережа та після застосування відсіву).

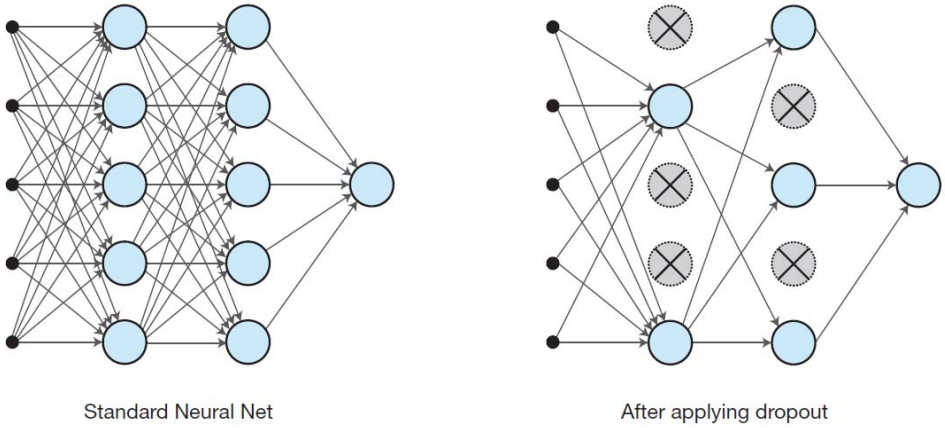


Рисунок 3.9 – Приклад нейронної мережі без відсіву (ліворуч) та з відсівом (праворуч)

Нормалізація в CNN – це метод, що використовується для підтримки однакового масштабу ознак у кожному шарі. Два найпоширеніші типи шарів нормалізації в CNN відомі як шари локальної відповіді та шари пакетної нормалізації.

Шар локальної нормалізації відгуку нормалізує вхідний сигнал по локальних областях, наприклад, 3 × 3. Його метою є сприяння узагальненню мережі. Основна ідея локальної нормалізації відгуку була натхненна латеральним гальмуванням, виявленим у реальних нейронах, тобто гальмуванням активації сусідніх нейронів, спричиненої збудженими нейронами. Однак це фіксований алгоритм, тому його гіперпараметри неможливо налаштувати протягом усього процесу навчання.

Шар пакетної нормалізації нормалізує вхідні дані шару шляхом віднімання середнього значення та ділення на стандартне відхилення кожної міні-партії вхідних даних (тобто підмножини навчального набору даних). Ця міні-пакетна нормалізація вносить певний шум до даних і призводить до певної форми регуляризації. Ця операція також дозволяє використовувати вищі швидкості навчання для пришвидшення процесу навчання. Однак застосування методу нормалізації до вхідних даних кожного шару може змінити представлення вихідних вхідних даних, наприклад, нормалізація вхідних даних сигмоподібної функції може обмежити вхідні дані в межах лінійної області сигмоподібної функції, отже, не використовуючи її нелінійну характеристику.

Тому в шарі пакетної нормалізації реалізовано додаткові параметри для керування масштабом та зміщенням нормалізованого значення, які необхідно вивчити разом з іншими параметрами мережі. Ці додаткові параметри дозволяють відновити вихідне значення, якщо воно дає кращі результати, ніж нормалізоване значення.

Активаційні шари зазвичай застосовуються після згорткових шарів, щоб вирішити, чи слід активувати певний нейрон чи ні. Робота згорткового шару, а потім шару активації в CNN виражається наступним чином

$$\boldsymbol{a}_j^{(\ell)} = \varphi\left(\left(\sum_{i=1}^{n} \mathbf{k}_{j,i}^{(\ell)} * \boldsymbol{a}_i^{(\ell-1)}\right) + b_j^{(\ell)}\right),$$

де n – кількість вхідних даних з попереднього шару, b – зміщення кожного згорткового фільтра, $k_{i,j}^{(l)}$– ядро кожного згорткового фільтра в згортковому шарі, яке з'єднує і -ту вхідну карту ознак з попереднього шару, а $\phi(\cdot)$ – функція активації, що використовується в активаційний шар.

Нелінійні активаційні функції, наприклад, сигмоїдна, Tanh, ReLU, витікаючий ReLU (LReLU) або параметричний ReLU (PReLU), зазвичай застосовуються по всій мережі, щоб мережа могла апроксимувати більшість нелінійних функцій.

Сигмоїдна функція відображає вхід у вихідний діапазон від 0 до 1. Вона базується на логістичній функції та виражається як:

$$f(x) = \frac{1}{\left(1 + e^{-x}\right)}.$$

Функція Танха (тобто гіперболічна тангенсна функція) відображає вхідний сигнал у вихідний діапазон від -1 до 1. Вона також базується на логістичній функції та визначається як:

$$f(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}}.$$

Логістична функція — це класична функція активації, яка спочатку використовувалася через її подібність до швидкості активації біологічного

нейрона. Вона зазвичай використовується завдяки своїй диференційовній властивості, що робить її придатною для алгоритму навчання зворотного поширення. Однак її властивість насичення, яка часто використовується для меж прийняття рішень, також викликає те, що відомо як зникнення градієнтів під час навчання.

Випрямлена лінійна одиниця, ReLU, є ненасиченою активаційною функцією, яка долає проблему зникнення градієнтів шляхом виключення експоненціальних членів. Вона визначається як

$$f(x) = \begin{cases} x & \text{if } x > 0 \\ 0 & \text{if } x \leq 0. \end{cases}$$

На відміну від сигмоїдної -функції активації, де кожен результат зваженої функції суми призводить до активованого нейрона, ReLU обмежує активацію нейрона, коли зважена сума менша або дорівнює нулю. Це спрощує мережу та зменшує час обчислення під час процесу навчання, що робить її найбажанішою функцією активації для прихованих нейронів. Однак ReLU має проблему. Оскільки градієнт функції ReLU в негативній області x дорівнює нулю, як тільки нейрон неактивний, він не буде активований знову протягом процесу навчання на основі градієнтного спуску, створюючи те, що відомо як проблема вмирання ReLU.

Витікаючий ReLU (LReLU) пропонує рішення проблеми вмираючого ReLU, встановлюючи градієнт негативної області на мале постійне значення c , тобто

.

$$f(x) = \begin{cases} x & \text{if } x > 0 \\ cx & \text{if } x \leq 0 \end{cases}$$

Параметричний ReLU (PReLU) перетворює мале константне значення c на параметр α , який можна навчати, щоб градієнт негативної області можна було відповідно навчити.

Хоча LReLU та інші модифіковані функції активації, наприклад PReLU, частково вирішують проблему згасання ReLU та демонструють переваги над ReLU, багато сучасних мереж сегментації все ще використовують ReLU як

функції активації по всій мережі, оскільки це дає задовільні результати та простіша в реалізації функція.

Для останнього шару нейронної мережі вибір функції активації залежить від завдання. Для виконання лінійної регресії часто використовується лінійна функція активації, яка створює вихідний сигнал, пропорційний вхідному сигналу без будь-яких перетворень. Для виконання класифікації сигмоподібна функція часто використовується в задачі класифікації з кількістю класів до двох, тоді як функція активації softmax зазвичай використовується для задачі класифікації з кількома класами, тобто більше двох класів. Функція активації softmax, також відома як нормалізована експонента, використовується для нормалізації вихідного сигналу мережі до розподілу ймовірностей за кількістю вихідних класів і задається формулою:

$$f(\mathbf{x})_i = \frac{e^{x_i}}{\sum_{j=1}^{K} e^{x_j}},$$

де і позначає елемент вхідного вектора x , а K позначає кількість вихідних класів.

3.7 Архітектури CNN

Архітектури CNN можуть бути розроблені для різних завдань комп'ютерного зору. Ці завдання можна в основному розділити на такі категорії: класифікація зображень, виявлення об'єктів та сегментація зображень.

Основна ідея класифікації зображень полягає в тому, щоб класифікувати зображення в один із набору класів відповідно до найважливіших ознак зображення. Це основа завдань виявлення об'єктів та сегментації зображень. Основними проблемами завдання класифікації є мінливість об'єктів через точку зору та внутрішньокласові дисперсії. Приклад завдання класифікації зображень показано на рисунку 2.10 .

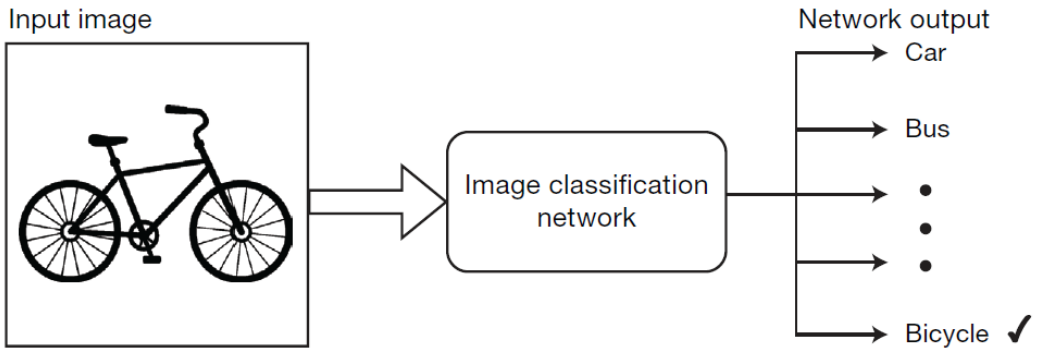


Рисунок 3.10 – Приклад завдання класифікації

Оригінальна реалізація згорткової зв'язної мережі (CNN) відома як LeNet-5. Вона була розроблена для класифікації зображень, зокрема, для задачі розпізнавання рукописних цифр. Вона складається зі згорткових, об'єднаних та повністю зв'язаних шарів. LeNet-5 складається з семи шарів і має загалом 60 тисяч параметрів, що підлягають навчанню. Вона приймає вхідні дані розміром 32 × 32, потім застосовує функцію активації Тана до кожного згорткового шару та використовує об'єднання усереднення для виконання зниження дискретизації.

Ще однією ранньою ЗНС, розробленою для задачі класифікації зображень, є AlexNet. Вона виграла конкурс ILSVRC 2012 року з класифікації основних об'єктів, присутніх на зображеннях, приблизно на 10 000 класів об'єктів. Архітектура AlexNet складається з восьми шарів і має загалом 60 мільйонів параметрів, що навчаються. Головною особливістю AlexNet є те, що вона використовує активацію ReLU, локальну нормалізацію відгуку та регуляризацію відсіву у своїй архітектурі. AlexNet вимагає фіксованого розміру вхідного зображення 224×224. Для навчання використовується кілька методів доповнення даних, таких як перетворення зображень, дзеркальне відображення та зміна інтенсивності RGB-каналів, щоб досягти низького рівня помилок. Для тестування мережа робить прогноз десяти обрізаних та доповнених зображень з вихідного зображення 256 × 256 та усереднює прогнози для отримання остаточного прогнозу.

Наступним проривом у CNN є мережа VGG. Мережа VGG використовує малі ядра фільтрів для побудови глибших мереж. Її найпродуктивніша мережа, VGG-19, має архітектуру 19 шарів і має загалом 144 мільйони параметрів для навчання. Решта її архітектури та методів реалізації були запозичені з AlexNet.

Іншим варіантом CNN є GoogLeNet. Ця архітектура перемогла у конкурсі класифікації зображень ILSVRC 2014 року. Архітектура GoogLeNet складається з 27 шарів і використовує загалом 6,8 млн параметрів, що навчаються. Її основна відмінність від вищезгаданих архітектур CNN полягає в тому, що вона використовує комбінацію згорткових фільтрів кількох розмірів паралельно, відому як початковий модуль. Кожен початковий модуль обробляє вхід модуля з згортковими фільтрами кількох розмірів і створює об'єднаний вихід цих фільтрів. Кожен з цих модулів дозволяє одночасно комбінувати різні рівні ознак. Цей початковий модуль також застосував у своїй архітектурі згорткові шари 1 × 1 для зменшення розмірності карти ознак, щоб зменшити загальну кількість параметрів, що навчаються. GoogLeNet також замінив повністю зв'язані шари на шар усереднення, щоб зменшити кількість параметрів, що навчаються.

Незважаючи на успіх GoogLeNet з початковими модулями, ReSNet виграв конкурс ILSVRC 2015 року. Він повернувся до оригінальної архітектури CNN, яка складається зі згортки, пулінгу та повністю зв'язаних шарів. Однак, він ввів пропуски з'єднань, щоб полегшити ідею наявності глибших мереж. Пропуски з'єднань надають моделі можливості повного використання глибшої архітектури або просто поверхневого аналога з фреймворком залишкового навчання. Він також використовує згорткові шари 1×1 у своїй архітектурі для зменшення розмірності карти ознак, що дозволяє побудувати 152-шарову мережу із загальною кількістю 60 мільйонів навчальних параметрів. Шар пакетної нормалізації (BN) використовується для оптимізації навчання глибшої мережі.

Після успіху ResNet з пропущеними з'єднаннями, DenseNet був розроблений шляхом модифікації навчальної структури ResNet. Замість використання додавання на пропущених з'єднаннях від ранніх до пізніших шарів було використано конкатенацію. DenseNet складається з 250 шарів і має загалом 15,3 млн параметрів, що навчаються. Мотивований ResNet, він використовує згортку 1 × 1 перед кожним згортковим шаром 3 × 3 для зменшення розмірності, щоб мінімізувати кількість параметрів, що навчаються.

Короткий виклад основних атрибутів CNN для класифікації зображень наведено в таблиці 3.1 .

Таблиця 3.1 – сновні атрибути CNN для класифікації зображень

Мережа	Основні атрибути
LeNet-5	Поєднання згортки, пулінгу та повністю зв'язаних шарів
AlexNet	Активація ReLu, нормалізація локальної відповіді, регуляризація відсіву
VGG	Глибша структура мережі
GoogleLeNet	Згортка 1x1, об'єднання глобальних середніх значень, початковий модуль
ReSNet	Пропустити зв'язки для формування залишкової навчальної структури
DenseNet	Щільний блок (пропустити з'єднання з усіх наступних шарів)

Виявлення об'єктів є складнішим завданням, оскільки воно поєднує складність двох завдань, тобто класифікації зображень та локалізації об'єктів. Класифікація зображень вимагає моделі для класифікації вхідного зображення в один із заздалегідь визначеного набору класів. Локалізація об'єктів вимагає моделі для створення обмежувальних рамок на виході, щоб вказати розташування та розмір об'єкта на зображенні.

Приклад завдання виявлення об'єктів показано на рисунку 2.11 , де мережа виявлення об'єктів створює обмежувальну рамку та мітку класу для кожного виявленого об'єкта (0 ,..., 4) на вхідному зображенні. Вихідна даними обмежувальної рамки зазвичай є початкова точка рамки в координатах x та y , за якою йдуть ширина, w , та висота, h рамки. Кожна обмежувальна рамка позначається як (x , y , w , h).

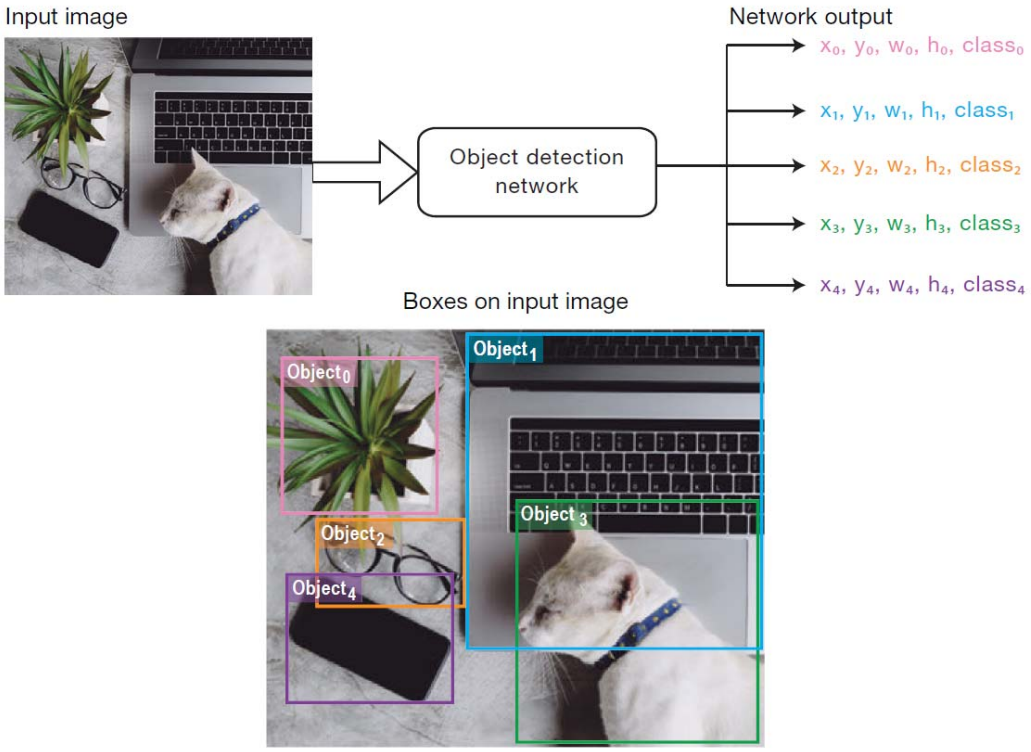


Рисунок 3.11 – Приклад завдання виявлення об'єктів

Одним із найдавніших алгоритмів виявлення об'єктів є R-CNN, який еволюціонував до Fast R-CNN, а потім до Faster R-CNN. Абревіатура R-CNN розшифровується як Regions with CNN features (Регіони з ознаками CNN). Як випливає з назви, це метод, заснований на CNN для виявлення об'єктів. Два методи, R-CNN та Fast R-CNN, спираються на алгоритм вибіркового пошуку для надання пропозицій щодо регіонів та попередньо навчену модель CNN (AlexNet використовувався в R-CNN, а VGG-16 - у Fast RCNN) для вилучення ознак. У R-CNN ознаки з CNN використовуються іншим алгоритмом машинного навчання, який називається машиною опорних векторів (SVM), та моделлю лінійної регресії для виконання класифікації та прогнозування обмежувального прямокутника відповідно [63]. SVM - це класичний алгоритм машинного навчання, який шукає найближчі зразки з різних класів, тобто опорні вектори, та використовує їх для формування найкращої розділювальної гіперплощини для прогнозування або класифікації нових зразків даних. Однак, у Fast R-CNN, SVM та модель лінійної регресії замінюються повністю зв'язаними шарами з softmax та лінійними функціями активації для класифікації та прогнозування обмежувальних рамок відповідно. Швидкість та

характеристики виявлення Fast R-CNN ще більше покращуються у Faster R-CNN, де алгоритм вибіркового пошуку замінюється простою згортковою мережею, яка називається мережею пропозицій регіонів (RPN), а Fast R-CNN зберігається як мережа виявлення. RPN складається зі згорткового шару $n \times n$, за яким йдуть два згорткові шари 1×1 з softmax та лінійною функціями активації відповідно. Як RPN, так і мережу виявлення у Faster R-CNN можна навчити використовувати спільні згорткові ознаки, отримані з попередньо навченої CNN, наприклад, VGG-16.

Мотивована складними конвеєрами R-CNN та недостатньою швидкістю Fast та Faster R-CNN у виявленні об'єктів у реальному часі, YOLO [49] була створена як перша мережа, яка може працювати в реальному часі з високою точністю. На відміну від сімейства RCNN, які складаються з двох етапів, тобто генерації блоку пропозицій регіону та класифікації, YOLO - це одноетапна мережа, яка прогнозує обмежувальні блоки та ймовірності класів об'єктів у вхідному зображенні за одну оцінку, звідси й назва "Ви дивитеся лише один раз" (YOLO). Архітектура YOLO натхненна GoogLeNet, де початкові модулі замінюються блоками, що містять згортковий шар 1×1, за яким йде згортковий шар 3×3. Вся архітектура складається з 30 шарів і має загалом 271,7 млн параметрів, де більшість параметрів, що навчаються, використовуються в повністю зв'язаних шарах. YOLO [49] вимагає фіксованого розміру вхідного зображення 448×448 для виявлення. Він використовує метод випадання як метод регуляризації та використовує Leaky ReLU замість ReLU як функцію активації, яка безпосередньо відповідає згортковим шарам по всій його архітектурі. Для останнього шару прогнозування використовується лінійна функція активації.

Короткий виклад основних атрибутів CNN для виявлення об'єктів наведено в таб. 3.2 .

Таблиця 3.2 – Основні атрибути CNN для виявлення об'єктів

Мережа	Основні атрибути
R-CNN	Ідея прогнозування обмежувальної рамки для локалізації об'єктів
Fast R-CNN	Використання нейронної мережі як моделі лінійної регресії для прогнозування обмежувального прямокутника
Faster R-CNN	Каскадне використання CNN для завдання виявлення об'єктів
YOLO	Нова архітектура CNN для одночасного прогнозування обмежувальних рамок та ймовірностей класів

3.8 Архітектури CNN для сегментації зображень

Сегментація зображення – це процес розділення зображення на кілька сегментів для полегшення аналізу. У звичайних нейронних мережах (CNN) сегментація зображення зазвичай називається процесом присвоєння мітки класу кожному пікселю зображення з попередньо визначеного набору класів, що призводить до створення попіксельної маски зображення як результату. Таким чином, результат CNN, що використовується для сегментації зображення, є точнішим за розміром та розташуванням порівняно з результатом CNN, що використовується для виявлення об'єктів.

Приклад завдання сегментації зображення показано на рисунку 2.12 , де кожному пікселю вхідного зображення присвоюється один клас із заздалегідь визначеного набору класів (наприклад, у цьому випадку людина, кінь, собака, земля та небо). Вихід мережі – це попіксельна маска зображення, що складається з різних міток класів, у цьому випадку вона візуалізується різними кольорами: людина – помаранчева, кінь – фіолетовий, собака – червоний, земля – зелена, а небо – блакитне.

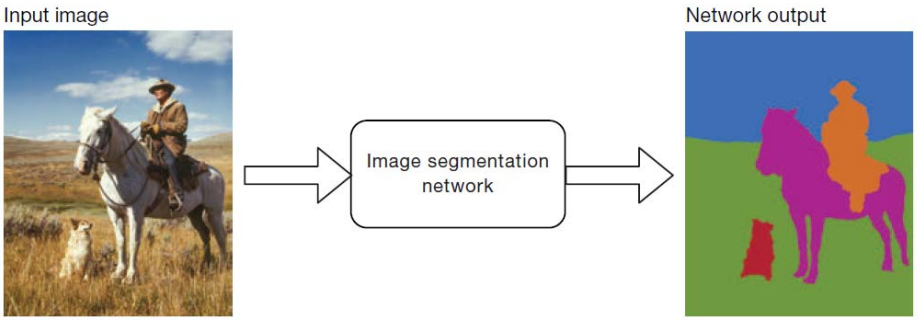


Рисунок 3.12 – Приклад завдання сегментації зображення

Однією з найуспішніших CNN для сегментації зображень є Fully

Згорткова мережа (FCN) побудована на основі архітектури класифікаційної мережі, наприклад, AlexNet, VGG-16 або GoogLeNet, шляхом видалення останнього шару мережі та перетворення повністю зв'язаних шарів на згорткові шари. Згортковий шар 1×1 з фіксованою кількістю згорткових фільтрів, що представляє кількість попередньо визначених класів, використовується для виконання класифікації кожної ознаки в кожному місці розташування ознаки. Цей класифікаційний шар 1×1 може бути використаний для виконання прогнозів при різних роздільних здатностях карти ознак шляхом додавання його до кількох вихідних місць розташування по всій мережі. Кожна вихідна карта, створена з набору зменшених карт ознак, має вихідну роздільну здатність нижчу, ніж початкова вхідна роздільна здатність, тобто грубий вихід. Транспонований згортковий шар з довжиною кроку, що дорівнює коефіцієнту підвищеної дискретизації, потім використовується для підвищеної дискретизації вихідних карт. Найпродуктивніша архітектура FCN – це FCN-8s, яка має довжину кроку 8 та класифікаційний шар 1 × 1 з транспонованим згортковим шаром, доданим у трьох різних місцях. Вихідні карти з підвищеною роздільною здатністю потім об'єднуються для формування піксельно-щільних виходів сегментації з дрібними деталями. Її головною характеристикою є нова архітектура сегментації зображень, отримана шляхом пропускання повністю зв'язаних шарів класифікаційної мережі, а потім об'єднання прогнозів з карт ознак з кількома роздільними здатностями з транспонованими згортковими шарами, які можна навчити для виконання підвищеної роздільної здатності. Найпродуктивніша FCN-8s базується на VGG-16 і має загалом 134 млн

параметрів, що навчаються. Ілюстрація архітектури FCN показана на рисунку 3.13 .

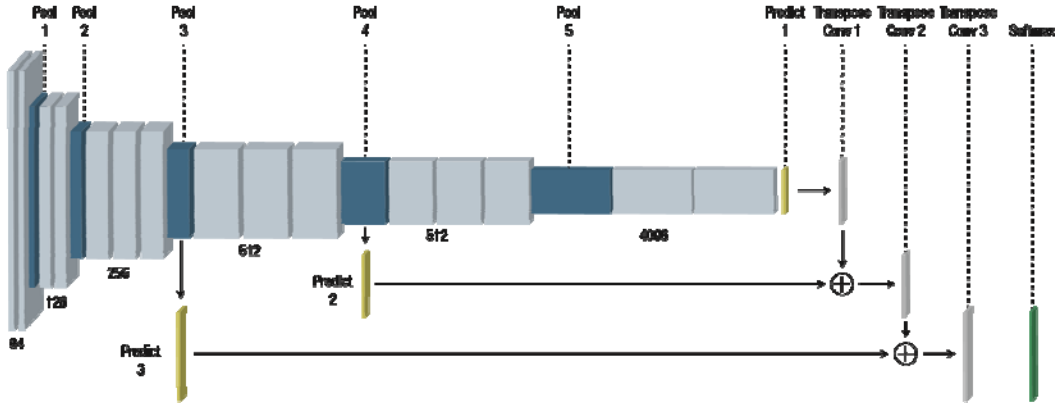


Рисунок 3.13 – Ілюстрація архітектури FCN

Ще одна згорткова мережа (CNN), що використовується для сегментації зображень, відома як U-Net. Ця мережа є модифікацією та розширенням FCN. U-Net складається з двох класифікаційних структур CNN, що утворюють U-подібну структуру, що стискається та розширюється. Частина, що стискається, використовується для вилучення ознак, тоді як частина, що розширюється, використовується для поширення контекстної інформації на шар з вищою роздільною здатністю (відображення ознак). U-Net також використовує пропускне з'єднання для об'єднання ознак від частини, що стискається, до частини, що розширюється, щоб допомогти з відображенням ознак. Компоненти U-Net подібні до FCN, за винятком методу підвищення роздільної здатності. Вона використовує згортку зі зміною розміру 2 × 2 замість транспонованої згортки. Архітектура U-Net складається з 23 згорткових шарів і має загалом 31 мільйон навчальних параметрів.

Ілюстрацію архітектури U-Net показано на рисунку 2.14 .

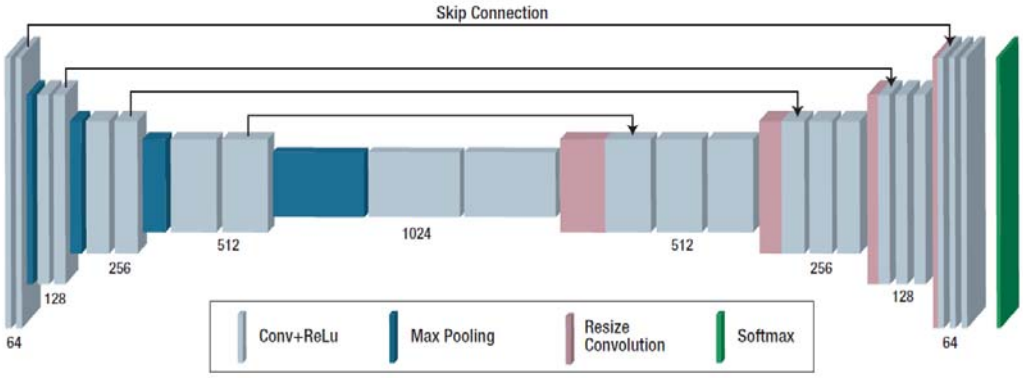


Рисунок 3.14 – Ілюстрація архітектури U-Net

Наступною розробкою є V-Net, що являє собою 3D CNN, створену шляхом заміни 2D згорткових шарів U-Net на 3D згорткові шари. Вона схожа за структурою на U-Net, проте приймає об'ємні дані. V-Net використовує подібний метод підвищення дискретизації до FCN, реалізує залишкове навчання у своїй архітектурі, як ResNet, виключає регуляризацію з відсіванням та використовує PReLU замість ReLU. Вона була розроблена для виконання об'ємної сегментації медичних зображень, оскільки медичні зображення часто мають 3D-природу.

Зведений огляд основних атрибутів CNN для сегментації зображень наведено в таблиці 2.3 .

Таблиця 3.3 – Основні атрибути CNN для сегментації зображень

Мережа	Головний атрибут
FCN	Нова архітектура CNN для сегментації зображень
U-Net	U-подібна мережева структура для сегментації зображень
V-Net	Тривимірна версія U-Net для виконання об'ємної сегментації

Структура, що використовується в сучасних ЗНН, залежить, перш за все, від завдання – чи то для класифікації, сегментації, чи для більш складного завдання, такого як виявлення об'єктів. ЗНН, розроблена для завдання класифікації або регресії, зазвичай використовує структуру для інтеграції просторових інваріантних властивостей у високорівневі ознаки для здійснення прогнозування. ЗНН, розроблена для сегментації, зазвичай використовує структуру, яка підходить для відображення високорівневих ознак назад до початкової вхідної роздільної здатності для сегментації. ЗНН, розроблена для більш складного завдання, такого як виявлення об'єктів, зазвичай використовує кілька алгоритмів/мереж для виконання різних операцій завдання.

Структури LeNet-5, AlexNet, VGG, GoogLeNet, ReSNet та DenseNet були розроблені та оптимізовані для завдань класифікації. У той час як FCN, U-Net та V-Net використовують структури мережі, яка спочатку не була розроблена та оптимізована відповідно до потреб завдання сегментації. FCN компенсує неоптимізований дизайн, використовуючи комбінацію кількох прогнозів з різною роздільною здатністю ознак, тоді як U-Net та V-Net використовують стек із двох класифікаційних мереж для формування структури, що стискається та розширюється, для відображення з поступово вищою роздільною здатністю від карт ознак з низькою роздільною здатністю до початкової вхідної роздільної здатності. Для завдання виявлення об'єктів R-CNN, Fast R-CNN та Faster R-CNN використовують багатоетапну структуру для застосування кількох традиційних алгоритмів та/або мереж для обробки різних операцій завдання. Однак більшість мереж, спроектованих з багатоступеневими структурами, зазвичай повільніші та трудомісткіші у використанні порівняно з мережами, спроектованими з одноступеневою структурою, наприклад, YOLO.

З таблиці 2.4 видно, що кількість параметрів, що навчаються, значно варіюється від 60 тис. до 297 млн. незалежно від структури мережі. Кількість змінюється залежно від глибини мережі, архітектури мережі та кількості згорткових фільтрів, що використовуються на кожному шарі мережі.

3.9 Зведення компонентів CNN

Компоненти, що використовуються в сучасних ЗНС, представлені в таблиці 2.5. Хоча кількість різних компонентів у сучасних мережах досить

обмежена, видатної продуктивності в різних застосуваннях все ще можна досягти за допомогою відповідних комбінацій цих різних компонентів та їх гіперпараметричних варіацій.Деякі з сучасних ЗНС покладаються на використання кількох мереж (наприклад, UNet, V-Net), кількох алгоритмів (наприклад, сімейство R-CNN) або великої кількості параметрів, що навчаються (наприклад, VGG, YOLO) для досягнення видатної продуктивності. В результаті, архітектура CNN часто має надлишкові компоненти та параметри.

3.10 Методи сегментації на базі нейронних мереж

Методи сегментації на основі нейронних мереж включають архітектури U-Net, FCN, Mask R-CNN, а також ансамблеві та гібридні підходи. Вони дозволяють виділяти об’єкти на зображеннях з високою точністю, використовуючи глибокі згорткові мережі та сучасні стратегії комбінування результатів.

Розглянемо класичні архітектури. Fully Convolutional Networks (FCN): перші моделі, що замінили повнозв’язні шари на згорткові, дозволяючи отримувати піксельні карти сегментації.

U-Net: симетрична архітектура з енкодером та декодером, популярна в медичній діагностиці та біології. Використовує skip-зв’язки для збереження деталей.

SegNet: подібна до U-Net, але з акцентом на збереження індексів пулінгу для точнішої реконструкції.

Розглянемо розширені моделі. Mask R-CNN: поєднує детекцію об’єктів і сегментацію, дозволяючи виділяти окремі об’єкти на сцені.

DeepLab (v3, v3+): застосовує атріальні згортки (dilated convolutions) для захоплення контексту на різних масштабах.

PSPNet (Pyramid Scene Parsing Network): використовує багаторівневий контекст для сегментації складних сцен.

Розглянемо ансамблеві та гібридні методи. Ансамблеві моделі: комбінують передбачення кількох нейронних мереж для підвищення точності та стійкості результатів. Гібридні підходи це поєднання CNN з трансформерами (наприклад, Segmenter, Swin-Transformer) для кращого врахування глобального контексту.

Таблиця 3.4 – Порівняння методів

Метод	Особливості	Переваги	Недоліки
FCN	Базова CNN для сегментації	Простота, швидкість	Менш точна деталізація
U-Net	Енкодер-декодер зі skip-зв’язками	Висока точність, медичні застосування	Вимагає багато даних
SegNet	Використання індексів пулінгу	Краще відновлення структури	Повільніше навчання
Mask R-CNN	Сегментація + детекція	Виділення окремих об’єктів	Високі обчислювальні витрати
DeepLab	Dilated convolutions	Контекст на різних масштабах	Складність налаштування
PSPNet	Pyramid pooling	Сегментація складних сцен	Високі ресурси

Недоліками є наступне:

- Висока обчислювальна складність — потрібні GPU та великі обсяги пам’яті.

- Необхідність великих датасетів для якісного навчання.

- Чутливість до шуму та артефактів у даних.

- Проблема узагальнення – моделі можуть працювати добре на тренувальних даних, але гірше на нових.

4 РЕЗУЛЬТАТИ ДОСЛІДЖЕНЬ СЕГМЕНТАЦІЇ ЗОБРАЖЕНЬ

Сегментація зображення залишається складним завданням з таких причин, як низька контрастність зображенн, розмиті межі. Крім того, значні варіації зовнішнього вигляду зображення, наприклад, форми та розмірів, у різних положеннях поперечного перерізу вздовж осі також сприяють труднощам автоматичної сегментації.

4.1 Дані для оцінки продуктивності сегментації

Для оцінки продуктивності запропонованої суміжної мережі порівняно з двома найсучаснішими двовимірними CNN U-Net та FCN-1 використовую набори даних з MICCAI 2017.

LiTS Challenge, що складається зі 130 професійно маркованих КТ-сканів. Набори даних та сегментації LiTS Challenge MICCAI 2017 надаються шістьма медичними центрами з кількох клінічних місць по всьому світу. Набори даних надходять у двох пакетах розміром 28 та 102 КТ-скани. Оскільки скани надходять з різних медичних центрів, деякі параметри сканування також відрізняються, наприклад, розміри сканування, розмір вокселів та орієнтація сканування. Кожне сканування складається з різної кількості 2D-зрізів, починаючи від 42 до 1026 зрізів. У цих наборах даних 2D-зріз складається з 512 × 512 пікселів. Розмір вокселів 2D-зрізів, як по ширині, так і по висоті, варіюється від 0,557 до 1 мм, тоді як товщина зрізу варіюється від 0,45 до 6 мм. Орієнтація сканування здійснюється в 3 різних анатомічних системах координат: ліва передня верхня (LAS), права передня верхня (RAS) та ліва задня верхня (LPS).

Архітектура 2D CNN, запропонована в цьому розділі, не залежить від високої дисперсії товщини зрізу, оскільки кожен із шарів обробки працює з 2D-даними. Для орієнтації сканування можна використовувати доповнення даних для вивчення різних орієнтацій або анатомічних систем координат. Однак немає потреби ускладнювати навчання, оскільки їх усі можна перевпорядкувати та перевести в певні системи координат. Тут ми використовуємо систему координат LAS.

Перед навчанням та тестуванням дані готуються шляхом зміни форми заданих двовимірних зображень зрізів розміром 512 × 512 до 224 × 224, щоб вони відповідали вхідному шару моделі. Далі область живота маскується за допомогою фільтра виявлення країв та операції мічення зв'язних компонентів, щоб виключити фон, створений різними сканерами. Нарешті, зображення покращуються розтягуванням контрасту та вирівнюванням гістограми, перш ніж центруватися за допомогою віднімання середнього.

4.2 Навчання

Зі 130 доступних сканів, 46 сканів використовуються для навчання, валідації та тестування. Для навчання 26 сканів взято з партії з 28, при цьому 2 випадково вибрані набори даних використовуються як валідаційний набір. Тестовий набір складається з 2 сканів, що використовувалися для валідації, та 18 сканів, випадково вибраних з 102.

Тестування проводиться двічі партіями по 10 сканувань, тобто 2 спліт-тести (Split 1 та Split 2). Це дозволяє оцінити можливості узагальнення кожної з мереж на наборах даних від невидимих сканерів.

Для навчання використовується ініціалізація He для ініціалізації ваг , а оптимізатор Adam використовується зі швидкістю навчання 10−4 та бінарною функцією перехресної ентропії.

Як видно з таблиці 4.1, запропонована суміжна мережа має найменшу кількість параметрів, що підлягають навчанню, та найшвидший час навчання за епоху порівняно з U-Net та FCN-1.

Таблиця 4.1– Порівняння продуктивності мережі

Мережа	Час навчання
U-Net	1931 секунд
FCN-1	790 секунд
Adjacent Network	211 секунд

4.3 Результати досліджень

У цій задачі сегментації використовую оцінки на основі перекриття , тобто коефіцієнт подібності Дайса (DSC) та коефіцієнт подібності Жаккара (JSC), як показники продуктивності для оцінки трьох мереж.

Метрика ДСК дорівнює подвоєній потужності перетину, у вокселях, між прогнозованою (P) та істинною (GT) областями, поділеній на суму потужностей областей P та GT, що визначається як

$$\mathrm{DSC} = \frac{2|P \cap GT|}{|P| + |GT|},$$

де | P | та | GT | – потужності областей P та GT відповідно.

Метрика JSC – це потужність кількості вокселів на перетині прогнозованої (P) та істинної (GT) областей, поділена на потужність кількості вокселів у їх об'єднанні, що визначається як

$$\mathrm{JSC} = \frac{|P \cap GT|}{|P \cup GT|}.$$

Результати кожного спліт-тесту показано на діаграмах типу «коробка-вуса» на рисунках 4.1 та 4.2.

З рисунків 4.1 та 4.2 видно, що суміжна мережа має найвищий максимальний DSC та JSC в обох спліт-тестах. Це показує здатність суміжної мережі перевершити продуктивність своїх конкурентів за обома показниками оцінки. Вона також досягає найвищого мінімального DSC та JSC у спліт-тесті на 2 порівняно з іншими. Хоча той/та/те мінімум спліт 1 тест рахунок з той/та/те суміжний мережа

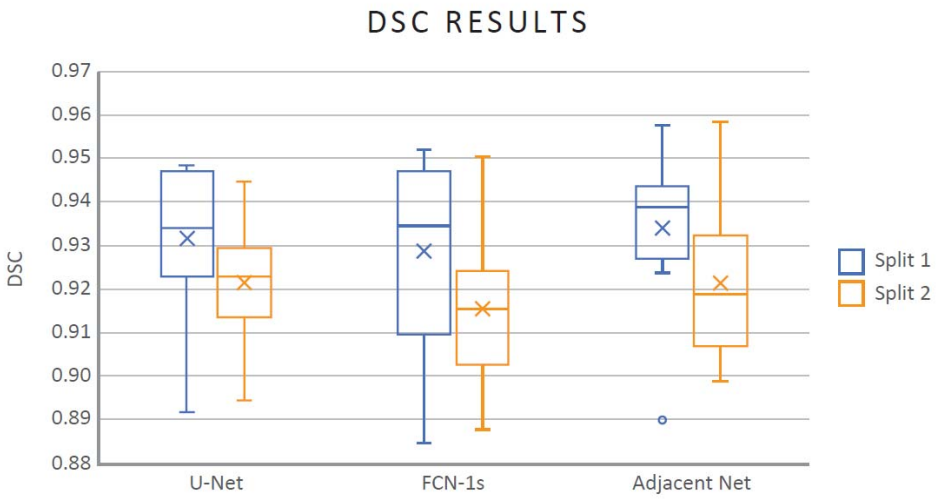


Рисунок 4.1 – Графік «коробка та вуса» для U-Net, FCN-1 та DSC суміжної мережі для тестів з розщепленням 1 та розщепленням 2.

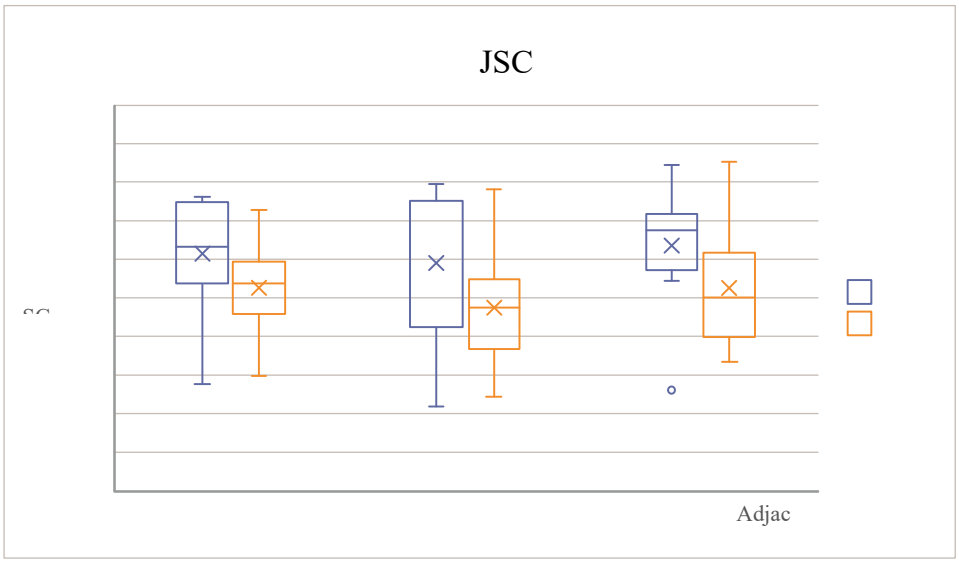


Рисунок 4.2 – Графік «коробка та вуса» для U-Net, FCN-1 та Adjacent Network JSC для тестів split 1 та split 2. Символ × у рамці вказує на середнє значення, а лінія в рамці вказує на медіане значення. ◦ представляє викид, рамка представляє діапазон , а «вуса» представляють верхню та нижню крайності, без урахування викидів.

Символ × у рамці вказує на середнє значення, а лінія в рамці вказує на медіанне значення. ◦ представляє викид, рамка представляє міжквартильний діапазон , а «вуса» представляють верхню та нижню крайності, виключаючи викиди.

не є найвищим, він приблизно такий самий, як у конкурентів (на 0,2% нижчий DSC, ніж у U-Net, та на 0,5% вищий DSC, ніж у FCN-1; на 0,3% нижчий JSC, ніж у U-Net, та на 0,8% вищий JSC, ніж у FCN-1). Рисунки 4.1 та 4.1 показують, що суміжна мережа працює порівняно добре як з точки зору DSC, так і з точки зору JSC, використовуючи при цьому значно менше параметрів, що піддаються навчанню.

Таблиця 4.2 – Порівняння продуктивності мережі

	Спліт 1	Спліт 2	Середнє	Спліт 1	Спліт 2	Середнє
U-Net	93.20	92,18	92,69	87.30	85,52	86,41
FCN-1	92,89	91,57	92,23	86,79	84,49	85,64
Adjacent Network	93,42	92,16	92,79	87,70	85,51	86,61

Загальні результати продуктивності зведені в таблиці 4.2, яка включає середнє значення (Avg) обох балів за спліт-тестами. Усі мережі, оцінені в порівнянні, досягають високих показників DSC та JSC. Запропонована суміжна мережа працює порівняно добре з іншими складнішими мережами, навіть незважаючи на те, що навчальна час значно менший (див. таблицю 4.2), що досягає дещо вищого середнього значення як у DSC (на 0,1% вище, ніж U-Net, та на 0,56% вище, ніж FCN-1), так і в JSC (на 0,2% вище, ніж U-Net, та на 0,39% вище, ніж FCN-1). У тесті Split 1 запропонована мережа досягає на 0,22% вищого DSC та на 0,4% вищого JSC, ніж U-Net. Однак у тесті Split 2 U-Net досягає на 0,01% вищих DSC та JSC, ніж запропонована мережа. З іншого боку, FCN-1 має трохи нижчі бали, ніж U-Net та запропонована суміжна мережа, в обох тестах.

Запропонована суміжна мережа використовує набагато меншу кількість параметрів, що навчаються, та менше пам'яті, хоча вона здатна працювати порівняно добре порівняно з сучасними мережами, такими як U-Net та FCN-1. Крім того, запропоновану мережу набагато легше навчати завдяки меншій кількості параметрів, що навчаються, та вона значно швидше обчислюється завдяки своїй ефективнішій архітектурі.

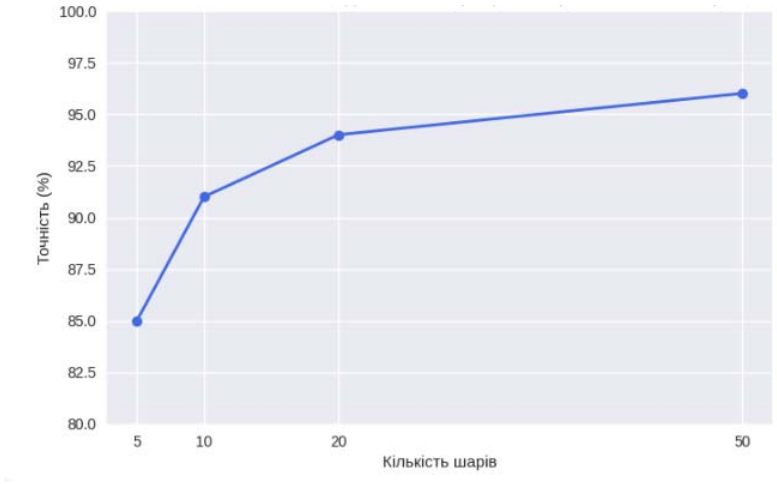


Рисунок 4.3 – Залежність точності від кількості шарів

На рис. 4.3 наведено залежність точності від кількості шарів для класичних CNN -архітектур (VGG, ResNet), де глибина покращує якість. Від 5 до 50 шарів точність стабільно зростає з 85% до 96%.

Це демонструє, що збільшення глибини мережі покращує якість до певної межі.

ВИСНОВКИ

У цій роботі виконано огляд методів в області сегментації. Автоматична сегментація є важливою галуззю досліджень у клінічних застосуваннях, оскільки вона мінімізує помилки, що виникають через упередженість, втому та мінливість експертів.

Основна увага приділяється інтелектуальним підходам, включаючи методи на штучних нейронних мереж. Вона дозволяє аналізувати більший обсяг даних з вищою швидкістю, точністю та узгодженістю в різних застосуваннях. Однак застосування нейронних мереж стало дорожчим з точки зору використання як часу, так і пам'яті в останні роки.

Розглянуто кілька архітектур з точки зору їхньої структури, кількості параметрів та компонентів, що навчаються, , які здатні ефективно досягати найсучасніших показників у різних завданнях сегментації.

Розглянуто особливості побудови зображення, а також методи попередньої обробки зображення для сегментації.

Наведено порівняння результатів, отриманих при використанні різних нейронних мереж.

Розроблено спеціалізоване програмне забезпечення для експериментального аналізу ключових показників ефективності запропонованого алгоритму. Показані результати оцінки продуктивності.

ПЕРЕЛІК ПОСИЛАНЬ

1. H. Bandyopadhyay, “Image segmentation: Deep learning vs traditional [guide].” [Online]. Available: https://www.v7labs.com/blog/image segmentation-guide.

2. Y. Guo, Y. Liu, T. Georgiou, and M. S. Lew, “A review of semantic segmentation using deep neural networks,” International journal of multimedia information retrieval, vol. 7, pp. 87–93, 2018.

3. S. Chavda and M. Goyani, “Scene level image classification: a literature review,” Neural Processing Letters, vol. 55, no. 3, pp. 2471–2520, 2023.

4. O. T.-C. Chen and C.-C. Chen, “Automatically-determined region of interest in jpeg 2000,” IEEE Transactions on Multimedia, vol. 9, no. 7, pp. 1333–1345, 2007.

5. A. M. Hafiz and G. M. Bhat, “A survey on instance segmentation: state of the art,” International journal of multimedia information retrieval, vol. 9, no. 3, pp. 171–189, 2020.

6. A. Kirillov, K. He, R. Girshick, C. Rother, and P. Dollar, “Panoptic seg mentation,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 2019.

7. S. Minaee, Y. Boykov, F. Porikli, A. Plaza, N. Kehtarnavaz, and D. Ter zopoulos, “Image segmentation using deep learning: A survey,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 44, no. 7, pp. 3523–3542, 2022.

8. M. Kass, A. Witkin, and D. Terzopoulos, “Snakes: Active contour models,” International journal of computer vision, vol. 1, no. 4, pp. 321–331, 1988.

9. L. Vincent and P. Soille, “Watersheds in digital spaces: an efficient algorithm based on immersion simulations,” IEEE Transactions on Pattern Analysis & Machine Intelligence, vol. 13, no. 06, pp. 583–598, 1991.

10. G. B. Coleman and H. C. Andrews, “Image segmentation by clustering,” Proceedings of the IEEE, vol. 67, no. 5, pp. 773–785, 1979.

11. L. Shafarenko, M. Petrou, and J. Kittler, “Automatic watershed segmen tation of randomly textured color images,” IEEE Transactions on Image Processing, vol. 6, no. 11, pp. 1530–1544, 1997.

12. N. Dhanachandra, K. Manglem, and Y. J. Chanu, “Image segmentation using k-means clustering algorithm and subtractive clustering algo rithm,” Procedia Computer Science, vol. 54, pp. 764–771, 2015.

13. A. Van Opbroek, M. A. Ikram, M. W. Vernooij, and M. De Brui jne, “Transfer learning improves supervised image segmentation across imaging protocols,” IEEE transactions on medical imaging, vol. 34, no. 5, pp. 1018–1030, 2014.

14. M. Majurski, P. Manescu, S. Padi, N. Schaub, N. Hotaling, C. Simon Jr, and P. Bajcsy, “Cell image segmentation using generative adversarial networks, transfer learning, and augmentations,” in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops, 2019, pp. 1114–1122.

15. S. Tammina, “Transfer learning using vgg-16 with deep convolutional neural network for classifying images,” International Journal of Scien tific and Research Publications (IJSRP), vol. 9, no. 10, pp. 143–150, 2019.

16. K. Fukushima, “Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position,” Biological cybernetics, vol. 36, no. 4, pp. 193–202, 1980.

17. A. Waibel, T. Hanazawa, G. Hinton, K. Shikano, and K. J. Lang, “Phoneme recognition using time-delay neural networks,” in Backprop agation. Psychology Press, 2013, pp. 35–61.

18. J. Long, E. Shelhamer, and T. Darrell, “Fully convolutional networks for semantic segmentation,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 3431–3440.

19. C. Sutton, A. McCallum et al., “An introduction to conditional random f ields,” Foundations and Trends® in Machine Learning, vol. 4, no. 4, pp. 267–373, 2012.

20. L.-C. Chen, Y. Yang, J. Wang, W. Xu, and A. L. Yuille, “Attention to scale: Scale-aware semantic image segmentation,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 3640–3649.

21. V. Badrinarayanan, A. Kendall, and R. Cipolla, “Segnet: A deep con volutional encoder-decoder architecture for image segmentation,” IEEE transactions on pattern analysis and machine intelligence, vol. 39, no. 12, pp. 2481–2495, 2017.

22. H. Noh, S. Hong, and B. Han, “Learning deconvolution network for semantic segmentation,” in Proceedings of the IEEE international conference on computer vision, 2015, pp. 1520–1528.

23. H.-S. Gan, M. H. Ramlee, A. A. Wahab, Y.-S. Lee, and A. Shimizu, “From classical to deep learning: review on cartilage and bone segmen tation techniques in

knee osteoarthritis research," Artificial Intelligence Review, vol. 54, no. 4, pp. 2445–2494, 2021.

24. A. Garcia-Garcia, S. Orts-Escolano, S. Oprea, V. Villena-Martinez, P. Martinez-Gonzalez, and J. Garcia-Rodriguez, "A survey on deep learn ing techniques for image and video semantic segmentation," Applied Soft Computing, vol. 70, pp. 41–65, 2018.

25. P. Jyothi and A. R. Singh, "Deep learning models and traditional automated techniques for brain tumor segmentation in mri: a review," Artificial intelligence review, vol. 56, no. 4, pp. 2923–2969, 2023.

26. H.-S. Gan, M. H. Ramlee, A. A. Wahab, Y.-S. Lee, and A. Shimizu, "From classical to deep learning: review on cartilage and bone segmen tation techniques in knee osteoarthritis research," Artificial Intelligence Review, vol. 54, no. 4, pp. 2445–2494, 2021