МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук
(повна назва)

Кафедра Програмної інженерії
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА

Пояснювальна записка

рівень вищої освіти другий (магістерський)

Дослідження методів сегментації та вбудови зображень для створення віртуальних сцен

(тема)

Виконав:
Студент 2 курсу, групи ІПЗм-19-1
Ушаков В.В.
(прізвище, ініціали)

Спеціальність 121- Інженерія програмного забезпечення
(код і повна назва спеціальності)

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

Керівник доц. Каук В.І
(посада, прізвище)

Допускається до захисту

Зав. кафедри _____ З.В. Дудар
(підпис) (прізвище, ініціали)

2021р.

Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук
(повна назва)

Кафедра Програмної інженерії
(повна назва)

Рівень вищої освіти другий (магістерський)

Спеціальність 121- Інженерія програмного забезпечення
(код і повна назва спеціальності)

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

Освітня програма Інженерія програмного забезпечення
(повна назва)

ЗАТВЕРДЖУЮ:
Зав.кафедри _____
(підпис)
« 26 » березня 2021 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

студента Ушакова Владислава
(прізвище, ім'я, по батькові)

1. Тема роботи Дослідження методів сегментації та вбудови зображень для створення віртуальних сцен
затверджена наказом університету від 26.03.2021 № 385
2. Термін подання роботи до екзаменаційної комісії 14 травня 2021р.
3. Вихідні дані до роботи *електронні ресурси за обраною тематикою, порівняльний аналіз методів семантичної сегментації, аналіз наборів даних COCO та Pascal VOC, середовища розробки PyCharm IDE, мова Python*
4. Перелік питань, що потрібно опрацювати в роботі *аналіз предметної області і постановка задачі, огляд методів сегментації екземплярів та згорткової нейронної мережі, аналіз математичних моделей згорткових мереж для вибору оптимальної, програмна реалізація методів семантичної сегментації, проведення*

експерименту, формування рекомендацій.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, слайдів, ілюстрацій *актуальність області дослідження, мета, постановка задачі, аналіз проблемної області, моделі з різними вимірами часу, розширена модель семантичної сегментації, етапи навчання згорткових мереж, вибір набору даних та алгоритму згорткової мережі, планування експерименту, програмна реалізація, результати, рекомендації, висновки.*

6. Консультанти розділів роботи

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата
Спецчастина	доц. Каук В.І.		12.05.21

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Аналіз проблемної області дослідження	25.01.21 – 11.02.21	виконано
2	Аналіз аналогів	7.02.21 – 11.02.21	виконано
3	Розробка постановки задачі	11.02.21 – 14.02.21	виконано
4	Дослідження методів комп'ютерного зору для пошуку об'єктів	14.02.21 – 20.02.21	виконано
5	Моделювання предметної області	15.02.21 – 21.02.21	виконано
6	Планування експериментальної частини дослідження	20.02.21 – 05.03.21	виконано
7	Аналіз алгоритмів згорткової нейронної мережі	05.03.21 – 25.03.21	виконано
8	Програмна реалізація засобів для експерименту	25.03.21 – 10.04.21	виконано
9	Проведення експериментів	05.04.21 – 14.04.21	виконано
10	Оформлення статті	10.04.21 – 21.04.21	виконано
11	Підготовка пояснювальної записки	01.04.21 – 30.04.21	виконано
12	Підготовка презентації та доповіді	30.04.21 – 03.05.21	виконано
13	Нормоконтроль	07.05.21 – 15.05.21	виконано
14	Рецензування	07.05.21 – 15.05.21	виконано
15	Занесення диплома в електронний архів	13.05.21	виконано
16	Попередній захист	15.05.21	виконано
17	Допуск до захисту у зав. кафедри	16.05.21	виконано

Дата видачі завдання 25 січня 2021р.
Студент

(підпис)

Керівник роботи доц. Каук В.І.

(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ / ABSTRACT

Кваліфікаційна робота магістра містить: 80 с., 20 рис., 1 табл., 12 джер.

КОМП'ЮТЕРНИЙ ЗІР, МОДЕЛЬ, РОЗПІЗНАВАННЯ ОБРАЗІВ, МАТУВАННЯ ЗОБРАЖЕННЯ, ОПТИМІЗАЦІЯ, OPENCV, R-CNN, TENSORFLOW, СЕГМЕНТАЦІЯ, PYTHON, ШТУЧНИЙ ІНТЕЛЕКТ, НЕЙРОННА МЕРЕЖА, НАБІР ДАНИХ.

Об'єктом дослідження є сегментація екземплярів та матування зображення. Предмет дослідження – алгоритми згорткової нейронної мережі на різних наборах даних для підготовки до матування зображення та вбудови віртуальних сцен.

Метою дослідження є аналіз існуючих рішень для сегментації екземплярів та алгоритмів згорткової нейронної мережі для матування зображень, їх вдосконалення, створення нових методів.

Дослідження, що будуть проведені, базуватимуться на результатах навчання моделей за готовими наборами даних, булевій алгебрі, згортковим нейронним мережам.

Методи розробки базуються на наступних мовах та технологіях: Python, Tensorflow, Caffe, open-cv, numpy.

У результаті роботи проаналізовано та досліджено різні методи сегментації та набори даних для отримання моделей, описано та досліджено алгоритми згорткової нейронної мережі для отримання тримапу зображення, сформовані загальні рекомендації щодо організації та роботи з сегментацією екземплярів та матування зображень.

COMPUTER VISION, MODEL, IMAGE RECOGNITION, IMAGE MATTING, OPTIMIZATION, OPENCV, R-CNN, TENSORFLOW, SEGMENTATION, ARTIFICIAL INTELLIGENCE, DATASETS.

The object of research is segmentation of copies and matting of the image. The subject of research is algorithms of convolutional neural network on different data sets for preparation for image matting and embedding of virtual scenes.

The aim of the study is to analyze existing solutions for segmentation of instances and algorithms of convolutional neural network for image matting, their improvement, creation of new methods.

The research that will be conducted will be based on the results of training models on ready-made data sets, Boolean algebra, convolutional neural networks.

Development methods are based on the following languages and technologies: Python, Tensorflow, Caffe, open-cv, numpy.

As a result, various segmentation methods and data sets for modeling were analyzed and researched, algorithms of convolutional neural network for image trimming were described and studied, general recommendations for organization and work with instance segmentation and image matting were formed.

Я, Ушаков Владислав Віталійович, студент гр. ІПЗм-19-1, здобувач вищої освіти на другому (магістерському) рівні кафедри «Програмна інженерія», заявляю: моя кваліфікаційна робота на тему «Дослідження методів сегментації та вбудови зображень для створення віртуальних сцен», що буде представлена в екзаменаційну комісію для публічного захисту, виконана самостійно, в ній не містяться елементи плагіату і вона може бути опублікована в електронному архіві відкритого доступу ElAr KhNURE. Всі запозичення з друкованих та електронних джерел мають відповідні посилання.

Я ознайомлена з діючим положенням «Про протидію академічному плагіату в ХНУРЕ», згідно з яким виявлення плагіату є підставою для відмови в допуску кваліфікаційної роботи до захисту та застосування дисциплінарних заходів.

ЗМІСТ

ВСТУП

Кожен день мільйони людей спілкуються з рідними та друзями, працюють або просто навчаються використовуючи онлайн відеозв'язок. Це стало невід'ємною частиною життя майже кожного з нас у період пандемії. Стрімкий ріст компаній котрі спеціалізуються на відеоконференціях таких як: Zoom, Skype, Google Hangouts та інших спонукає та потребує розвитку програмного забезпечення для проведення та підтримки таких відеоконференцій.

Сфера комп'ютерного зору та машинного навчання в останні часи невпинно зростає. В першу чергу це пов'язано з удосконаленнями апаратного забезпечення, що дає змогу обробляти великий обсяг інформації та даних набагато швидше. Коли комп'ютерний зір почав формуватися як поле в 1960-х роках, його метою було спробувати імітувати системи людського зору і попросити комп'ютери розповісти нам, що вони бачать, автоматизуючи процес аналізу зображень. Починаючи з 2010 року ми спостерігали прискорення вдосконалення методів та технологій глибокого навчання. Завдяки глибокому навчанню ми тепер можемо запрограмувати суперкомп'ютери, щоб вони тренувались, самовдосконалювались з часом і надавали частину цих можливостей бізнесу як онлайн-додатки, наприклад, хмарні програми.

Усе це дало різкий штовчок для розвитку комп'ютерного зору разом із штучним інтелектом. Багато великих компаній таких як Facebook або Microsoft витрачають багато часу та великі ресурси для вивчення цієї галузі. Вони розробляють велику кількість програмного забезпечення, або просто вдосконалюють існуючі додатки з використанням комп'ютерного зору для вирішення найрізноманітніших завдань.

На сьогоднішній день комп'ютерний зір має широке використання у найрізноманітніших галузях . Наприклад широке розповсюдження він набув у галузі медицини, де за його допомогою можна виявити діагноз, або знайти проблему за рентгенівським знімком або будь яким іншим фото та відео

матеріалом. Розпізнавання образів вже стало справжнім проривом в медицині - у багатьох випадках комп'ютери помічають речі, які пропускають навіть найдосвідченіші лікарі. Вони виступають своєрідними помічниками, чия «технічна» думка підтверджує гіпотезу лікаря або дає привід для більш глибоких досліджень.

Іншою галуззю є фізика, де розповсюдження комп'ютерний зір набув для вимірювання та аналізу тепловипромінювання або електромагнітного випромінювання. Але частіше за все комп’ютерний зір йде поряд з штучним інтелектом та машинним навчанням. Найрозповсюджені галузі це розпізнавання образів та навчальні методи. За допомогою штучного інтелекту можливо робити такі вимірювання та виконувати завдання які раніше і уявити неможливо було. Методи машинного навчання в рамках комп’ютерного зору швидко покращують розуміння комп'ютером зображень на високому рівні, тим самим відкриваючи нові можливості для виконання завдань, які раніше вимагали втручання людини вручну

Машинне навчання дозволяє не тільки виконувати різноманітні завдання комп’ютерного зору за запрограмованим алгоритмом, він може самовдосконалюватись за рахунок великого обсягу даних для самонавчання. Машинне навчання сьогодні стало більш популярним завдяки постійно зростаючому обсягу даних, вдосконаленим алгоритмам та вдосконаленню обчислювальної потужності та зберігання.

Машинне навчання покращило бачення комп’ютера щодо розпізнавання та відстеження. В останні роки спостерігається збільшення кількості досліджень щодо виявлення об'єктів, сегментації екземплярів зображень, відстеження відеооб'єктів, виявлення відеооб'єктів, семантичної сегментації відео та сегментації відеооб'єктів.

У даній магістерській роботі планується розглянути проблеми сегментації об’єктів та вбудови зображення. На сьогоднішній день головна проблема сегментації об’єктів є обмежена кількість моделей для навчання, та недосконалі алгоритми сегментації об’єктів та вбудови зображень. У той же час основним

обмеженням машинного навчання є загальна відсутність анотованих даних. Це обмеження стає ще більш значущим, коли мова йде про глибоке навчання, оскільки його ефективність зазвичай пропорційна кількості наявних анотованих даних. Тому метою даної роботи є виявлення кращих методів сегментації для вирішення проблеми вбудови зображень, та, можливо, вдосконалення існуючих алгоритмів для більш точного результату роботи цих двох методів.

1 АНАЛІЗ ПРОБЛЕМНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ

Комп’ютерний зір та штучний інтелект відносно молода галузь комп’ютерних наук, але незважаючи на це в останні роки вона набуває все більшого розвитку. Розпізнавання образів, предметів, речей та іншого застосовується у різноманітних галузях для вирішення багатьох людських проблем. Розвиток цього напрямку зацікавлює багатьох людей та компаній. Навіть найбільші компанії світу таки як Facebook, Instagram та Google у багатьох своїх сервісах та додатках активно використовують штучний інтелект та комп’ютерний зір.

Одночасно з розвитком комп’ютерного зору набирає оберти один з його напрямків сегментація зображень. Є декілька типів сегментації зображень. Спочатку займались розпізнаванням об’єктів. Його основним завданням було розпізнавання об’єктів на зображенні, виділення границь цього об’єкта за допомогою рамки. Далі розвитку набуло семантичне сегментування, коли на зображення кладуть маску та знаходять об’єкти попіксельно. У цьому методі кожному пікселю присвоюють один з класів, таким чином можно виявити різні класи об’єктів. Після цього об’єднали ці два методи для пошуку сегментованих об’єктів та розрізняти їх не лише по класу а й по екземплярам одного класу.

Рання робота над сегментацією екземплярів була виконана Вінном та Шоттоном. Підготовлено одинарний класифікатор на піксель для передбачення частини об’єкта. Потім цим частинам пропонувалося підтримувати просторовий порядок, що характерно для екземплярів, використовуючи асиметричні попарні потенціали в умовному випадковому полі (CRF).

Однак сегментація екземплярів стала більш поширеною після “Одночасного виявлення та сегментації” (SDS) роботи Харіхарана та співавт. Ця система базувалася на R-CNN пайплайнах. Генеруються регіональні пропозиції, класифікувались на категорії об’єктів за допомогою Convolutional Neural Network (CNN) перед застосуванням обмежувальної регресії як постобробки. Потім у

цьому обмеженні була проведена конкретна сегментація для одночасного виявлення та сегментування об'єкта.

Крім того виконуються численні кроки пост-обробки, такі як «проекція суперпікселя» та переформатування. Дай подивився на деякі з цих питань, розробивши один наскрізний тренувальну мережу, яка генерує пропозиції, створює передній план маски з цих пропозицій, а потім класифікує ці маски. Ця мережа може розглядатися як розширення кінцевого фреймворку Faster RCNN[1], яка генерує пропозиції та класифікує їх.

Окремим шляхом також були розроблені алгоритми які не потребують детекторів об’єктів. Чжан разом з колегами працював над сегментуванні екземплярів автомобілів, передбачаючи порядок глибини кожного пікселя на зображенні. На відміну від попереднього виявлення ґрунтуючись на підходах, цей метод аналізував усі випадки на зображенні одночасно (а не окремі пропозиції) із формулюванням на основі MRF. Крім того, хоча цей метод не використовує виявлення об'єктів, він є мережею для навчання і припускає максимум дев’ять автомобілів на зображенні. Прогнозування всіх випадків у зображення одночасно (а не класифікація особої категорії об’єктів) вимагає, щоб модель могла обробляти змінну кількість екземплярів виводу на зображення. В результаті була запропонована для цього завдання періодична нейронна мережа (RNN). Однак ця модель стосувалася лише однієї категорії об’єктів.

Один з напрямків використання сегментації екземплярів є image matting. Ці сфери доповнюють один одного, спочатку сегментуються екземпляри для знаходження переднього фону та заднього, а потім ці дані використовуються для чіткого отримання зображення без зайвого фону.

Матування зображень є ключовим методом редагування зображень та відео та композиції. З огляду на вхідне зображення та тримап, що визначають фон, передній та невідомий регіони, застосовується матування зображення для оцінки альфа-матовості всередині невідомої області, щоб чітко відокремити передній план від фону. Нещодавно багато методів, заснованих на глибокому навчанні (Xu et al. 2017; Lu et al. 2019; Hou and Liu 2019; Cai et al. 2019), досягли значного

вдосконалення порівняно з традиційними методами (Wang and Cohen 2007; Gastal and Oliveira 2010; Sun et al. 2004; Levin, Lischinski, and Weiss 2007; Grady et al. 2005). Ці методи глибокого навчання (Xuet al. 2017; Lu et al. 2019; Hou та Liu 2019) здебільшого беруть цілі зображення та пов'язані з ними цілі тримапи як вхідні дані, і використовують глибокі нейронні мережі, такі як VGG (Si-monyan and Zisserman 2014) та Xception (Chollet 2017) як основи їх мережі.

Однак ці методи можуть зазнати невдачі при роботі з входами високої роздільної здатності (HR). Матування зображень часто застосовується до HR-зображень розміром, наприклад 5000 × 5000 або навіть вище, у реальних програмах. Через апаратні обмеження, такі як пам'ять графічного процесора, зображення HR не можуть бути оброблені безпосередньо попередніми методами глибокого навчання. Дві загальноприйняті стратегії адаптації цих методів - це вибіркова обробка вхідних даних (He, Sun та Tang 2010) або тривіальний висновок на основі патчів. Перша стратегія призводить до втрати найдрібніших деталей, а друга спричиняє нерівномірність. Крім того, зображення HR можуть мати більші або навіть зовсім невідомі області в межах програми. Це також вимагає від моделей розуміння контекстної інформації з далеких латок для успішного матування.

Метою семантичної сегментації є отримання точного висновку шляхом передбачення міток для кожного пікселя зображення. Кожен піксель позначається класом відповідно до об'єкта або регіону, в якому він укладений. У цьому напрямку сегментація екземплярів надає різні мітки для окремих екземплярів об’єктів, що належать до одного і того ж класу об’єктів. Таким чином, сегментацію екземпляра можна визначити як завдання пошук одночасного рішення виявлення об’єкта, а також семантичної сегментації.

Сегментація на основі частин продовжує еволюцію цього дослідження, розкладаючи кожен з сегментованих об’єктів на відповідні під компоненти.

Виявлення людини - це гаряча точка в технологіях комп’ютерного зору, яка відіграє важливу роль у транспортному засобі помічник водіння та відеоспостереження. Виявлення виявлення людини є продовженням людини

виявлення, що є більш складним завданням, головним чином тому, що люди мають різну позу і тіло зовнішність, а фон людського середовища, як правило, складний та деякі інші зовнішні фактори, такі як зміна світла та оклюзія. Традиційні методи виявлення в основному засновані на моделюванні фону та статистичних даних навчання, на результати діяльності якого сильно впливає опис особливостей та класифікатор людини. Основними ознаками людських тіл на зображенні, що описується, є Haar та напрямок градієнта гістограма, і основними класифікаторами, що використовуються, є нейронна мережа, підтримка векторної машини та підсилення.

1.1 Опис проблеми

Ідея семантичної сегментації полягає у розробці техніки / алгоритму, яка добре працює в двох сферах, що забезпечує кращу точність сегментації та кращу ефективність сегментації. Краща точність сегментації охоплює точну локалізацію та розпізнавання об'єктів на зображеннях / кадрах, в результаті чого можна виділити велику різноманітність категорій, пов'язаних з об'єктами в реальному сценарії (тобто кращу відмінність), а також випадки об'єктів, що належать до одного класу і є предметом до внутрішньо класових змін зовнішнього вигляду, можуть бути локалізовані та розпізнані (тобто краща стійкість).

Підвищення ефективності сегментації відноситься до обчислювальних витрат алгоритму сегментації. Це стосується ефективних обчислювальних витрат у режимі реального часу, таких як прийнятні вимоги до пам'яті / зберігання та менший навантаження на процесор. Одним з важливих компонентів детектора об'єктів для сегментації є хороша репрезентація характеристик, що має першочергове значення при виявленні об'єктів. Раніше було докладено чимало зусиль для розробки місцевих дескрипторів (як SIFT та HOG) та для вивчення підходів (як Bag of Words (та Fisher Vector)) для того, щоб згрупувати та абстрагуватись дескрипторів у представлення високого рівня для появи дискримінаційних частин. Недоліком було те, що ці методи представлення

функцій потребували ручної роботи з тонкої техніки та великої кількості доменних знань. На відміну від цього, методи, засновані на глибокому навчанні (наприклад, Deep CNN), можуть навчитися потужним представлення функцій з різним рівнем абстракції із зображень. Згодом проблема представлення об’єктів була перенесена на розробку ефективніших мережевих архітектур та більш оптимізованих навчальних процедур.

Тенденція еволюції мережевої архітектури набуває все більшої глибини: AlexNet мав вісім шарів, VGGNet - 16 шарів, а нещодавно ResNet та DenseNet мають понад 100 шарів. Насправді саме VGGNet та GoogLeNet показали, що зі збільшенням глибини мережі можна збільшити потужність представлення мережі. Глибинні мережі, такі як AlexNet, OverFeat, ZFNet та VGGNet, мають надзвичайно велику кількість параметрів, хоча у них мало шарів. Це можна пояснити величезною кількістю параметрів, що надходять із повністю пов'язаних шарів. Нові мережі, такі як Inception, ResNet та DenseNet, хоч і мають велику глибину, мають набагато менше параметрів, уникаючи використання повністю пов'язаних шарів.

Глибокі детектори на базі CNN, такі як RCNN[2], Fast RCNN, Faster RCNN та YOLO, зазвичай використовують глибокі архітектури CNN, а згодом використовують функції верхнього рівня CNN для представлення об'єктів. Але є проблема. Виявлення об'єктів у різних масштабах є великою проблемою. З метою вирішення цієї проблеми детектор пройшов через піраміду зображень. Хоча такий підхід, як правило, призводить до більш точного виявлення, проте він страждає від обмежень часу виводу, а також від обчислювальних ресурсів, таких як пам'ять.

1.2 Постановка задачі

Незважаючи на велику кількість методів та алгоритмів семантичної сегментації та сегментації екземплярів неможливо однозначно вирішити який з методів кращий для того чи іншого типу завдання. У той же час є різні підходи матування зображення для різної якості зображення та різних вхідних даних. На

якість також можуть впливати коефіцієнти альфа, для більш чіткого виявлення переднього плану та фону. Якість сегментації залежить від багатьох факторів, найважливішими з котрих є:

– завдання яке ця модель повинна виконати;
– датасет для навчання моделі;
– обраний метод для сегментації;
– ресурси які будуть задіяні за для обробки моделі.

Виходячи з цього основною проблемою вибору алгоритму сегментації екземплярів є відсутність аналізу алгоритмів для кожного типу завдань та інших критеріїв. Усі методи або алгоритми були проаналізовані та досліджені на основі одного виду вхідних даних задля вирішення одного типу завдань. Це обумовлено тим що для різного завдання можуть підходити різні методи сегментації та набори даних. Різноманітні варіації та вдосконалення алгоритму под чітку задачу можуть значно покращити результат її роботи.

Тому в рамках цієї атестаційної роботи, будуть представлені результати дослідження ефективності методів сегментації екземплярів, будуть представлені результати обробки даних різними алгоритмам, проаналізоване навантаження кожної з моделей для отримання навченної нейронної мережі.

Метою даної роботи є дослідження методів сегментації екземплярів, формування емпіричних знань про ефективність методів та визначення факторів, пов’язаних з ефективністю використання кожної моделі для різних типів завдань. Аналіз роботи методів з різними параметрами та значеннями, використовування різних наборів даних.

Для досягнення мети дипломної роботи необхідно дослідити:

– швидкість обробки зображень та навчання нейронної мережі на одному наборі медіаданих;
– якість сегментування та обробки медіафайлів для різних видів завдань;
– кількість задіяних ресурсів системи для створення моделі;
– вибір методів сегментації для вирішення завдань з матування зображення.

Усі дослідження будуть проводитися використовуючи найпопулярніший на даний момент набір даних COCO[3]. Набір даних COCO - це широкомасштабний набір об’єктів виявлення, сегментації, виявлення ключових точок та субтитрів який складається з 328 тис. зображень. Цей набір даних ідеально підійде до нашого типу завдання. Також для аналізу інших алгоритмів сегментування будуть проаналізовані алгоритми з застосуванням інших наборів даних які можуть більш підходити для вирішення нашої пробелми. Це обумовлено тим що у нашому випадку нам не потрібно аналізувати всі об’єкти на зображенні, наше завдання з найбільшою точністю виявити один об’єкт, який буде виступати у ролі переднього плану.

2 МЕТОДИ КОМП’ЮТЕРНОГО ЗОРУ ДЛЯ ПОШУКУ ОБ’ЄКТІВ

Зображення в основному є двовимірною просторовою функцією координати, f (x, y) та амплітуда цієї функції при заданому координата дає значення інтенсивності зображення. Зображення може бути виражена як добуток функцій освітлення і роздуми.

$$f(x, y) = i(x, y) + r(x, y),$$

де i (x, y) - функція інтенсивності, а r (x, y) - функція відбивної здатності.

Цифрова обробка зображень - це застосування різних алгоритми на зображенні для поліпшення якості зображення шляхом видалення шуму та інших небажаних пікселів, а також до отримати більше інформації про зображення. Серед різноманітних технік обробки зображень зображення сегментація є дуже важливим кроком для аналізу даного зображення.

Існують різні види обробки зображень. Для рішення нашого завдання мі будемо розглядати методи пошуку об’єктів на зображенні. Основними методами для рішення цього завдання є семантична сегментація, розпізнавання об’єктів, класифікація зображень та сегментація екземплярів.

2.1 Згорткова нейронна мережа

Перш ніж розглядати методи сегментації та розпізнавання образів слід розглянути згорткову нейронну мережу, тому що вона використовується майже у кожному алгоритму сегментації.

Згорткові нейронні мережі широко використовуються в програмах розпізнавання зображень машинного навчання. Свертові нейронні мережі надають перевагу перед мережами зворотного зв'язку, оскільки вони здатні враховувати розташування елементів.

Згорткова нейронна мережа (ConvNet / CNN) - це алгоритм глибокого навчання, який може приймати вхідне зображення, призначати важливість (зважувати ваги та упередження) різним аспектам / об'єктам на зображенні та мати можливість диференціювати один від іншого[4]. Попередня обробка, необхідна в ConvNet, набагато нижча порівняно з іншими алгоритмами класифікації. Хоча в примітивних методах фільтри розробляються вручну, при достатній підготовці, ConvNets має можливість вивчати ці фільтри / характеристики.

Архітектура ConvNet аналогічна структурі зв’язку нейронів в мозку людини і натхненна організацією зорової кори. Окремі нейрони реагують на подразники лише в обмеженій області зорового поля, відомому як рецептивне поле. Колекція таких полів перекривається, щоб охопити всю зорову зону.

ConvNet здатний успішно фіксувати просторові та тимчасові залежності в зображенні за допомогою відповідних фільтрів. Архітектура краще підходить до набору даних зображення завдяки зменшенню кількості задіяних параметрів та повторному використанню ваг. Іншими словами, мережу можна навчити краще розуміти деталі зображення.

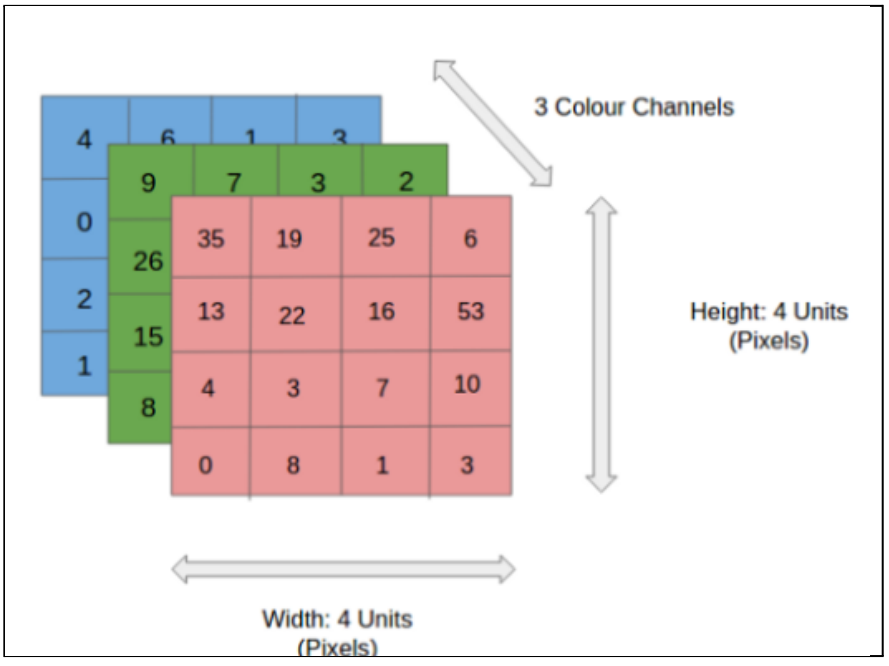

Рисунок 1 - Зображення 4x4x3

На малюнку 1 ми маємо зображення RGB, яке розділене трьома кольоровими площинами - червоною, зеленою та синьою. Існує ряд таких колірних просторів, в яких існують зображення - відтінки сірого, RGB, HSV, CMYK[5] тощо. Ви можете собі уявити, наскільки затратними у обчисленні будуть речі, як тільки зображення досягнуть розмірів, скажімо, 8K (7680 × 4320). Мета CNN полягає у зменшенні зображень у форму, яку легше обробити, не втрачаючи особливостей, які є критично важливими для отримання гарного прогнозу. Це важливо, коли ми хочемо розробити архітектуру, яка не тільки добре володіє методами навчання, але й масштабована до масивних наборів даних.

Згорткові нейронні мережі складаються з безлічі шарів штучних нейронів. Штучні нейрони, груба імітація їх біологічних аналогів, є математичними функціями, які обчислюють зважену суму кількох входів і виводять значення активації (Рисунок 2).

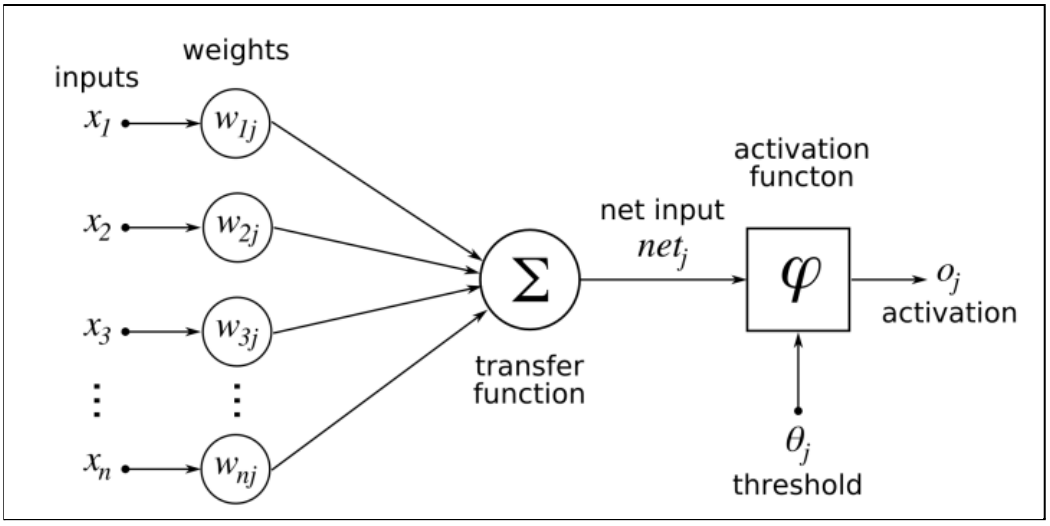

Рисунок 2 - Згорткова нейронна мережа

Поведінка кожного нейрона визначається його вагою. При подачі значень пікселів штучні нейрони CNN виділяють різні візуальні особливості.

Коли ви вводите зображення в ConvNet, кожен з його шарів генерує кілька карт активації. Карти активації виділяють відповідні особливості зображення. Кожен з нейронів бере в якості вхідного сигналу ділянку пікселів, множить значення кольорів на його вагу, підсумовує їх і запускає через функцію активації.

На основі карти активації остаточного рівня згортки рівень класифікації видає набір оцінок достовірності (значення від 0 до 1), які вказують, наскільки ймовірно, що зображення належить до “класу”. Наприклад, якщо у вас є ConvNet, який виявляє котів, собак і коней, результат остаточного шару - це можливість, що вхідне зображення містить будь-яку з цих тварин.

2.2 Семантичне сегментування

Якщо спростити, то мета семантичної сегментації - це зробити кольорове зображення RGB (висота × ширина × 3) або якщо розглядати зображення у градації сірого (висота × ширина × 1) і вивести карту сегментації, де кожен піксель містить мітку класу, представлену у вигляді цілого числа (висота × ширина × 1) як зображено на рисунку 3.

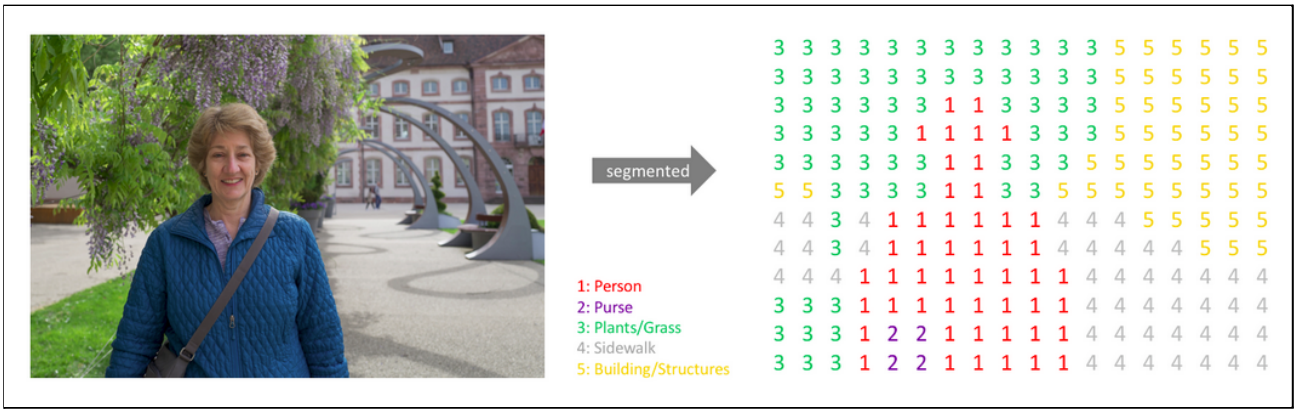

Рисунок 3 - Попіксельна сегментація зображення

Прогноз можна згорнути в карту сегментації (як показано на першому зображенні), взявши argmax кожного піксельного вектора по глибині. Ми можемо легко оглянути ціль, наклавши її на спостереження[6]. Коли ми накладаємо одне джерело нашої цілі (або передбачення), це називається маскою, яка висвітлює області зображення, де присутній конкретний клас.

У загальноприйнятій термінології семантична сегментація є основною метою класифікації кожного пікселя зображення, див. Рисунок 4. Подібно

класифікації та локалізації, семантична сегментація не розрізняє об’єкти, а лише класи. Це означає, що якщо на зображенні присутні кілька автомобілів, усі пікселі, що належать автомобілям, будуть класифіковані до одного класу. Хоча семантична сегментація не може відлічати об'єкти, вона все одно може бути корисною для аналізу та сегментування об’єктів які знаходяться на відео, оскільки результати сегментації потенційно можуть бути використані для встановлення границь людини та інших об’єктів.

Семантична сегментація є завданням позначення кожного пікселя зображення як певного класу.

Рисунок 4 - Семантична сегментація об’єктів

Зазвичай на зображенні з різними об’єктами ми хочемо знати, який піксель до якого об’єкта належить, Наприклад, на зображенні на відкритому повітрі ми можемо сегментувати дороги, пішоходів, вуличні ліхтарі тощо.

Семантична сегментація відрізняється від виявлення об’єктів, оскільки вона не передбачає будь-яких обмежувальних полів навколо об’єктів. Ми не класифікуємо різні екземпляри одного і того ж об’єкта. Семантична сегментація відрізняється від сегментації екземплярів, яка полягає в тому, що різні об’єкти

одного класу матимуть різні мітки. Наприклад на малюнку де зображено декілька людей кожен з них буде сегментований окремо та мати різний колір маски.

Наївний підхід до побудови архітектури нейронної мережі для цього завдання полягає у простому складенні декількох згорткових шарів (з однаковим відступом для збереження розмірів) і виведенні остаточної карти сегментації як на рисунку 5. Це безпосередньо вивчає відображення від вхідного зображення до його відповідної сегментації шляхом послідовного перетворення відображень об’єктів; однак зберегти повну роздільну здатність у всій мережі досить обчислювально.

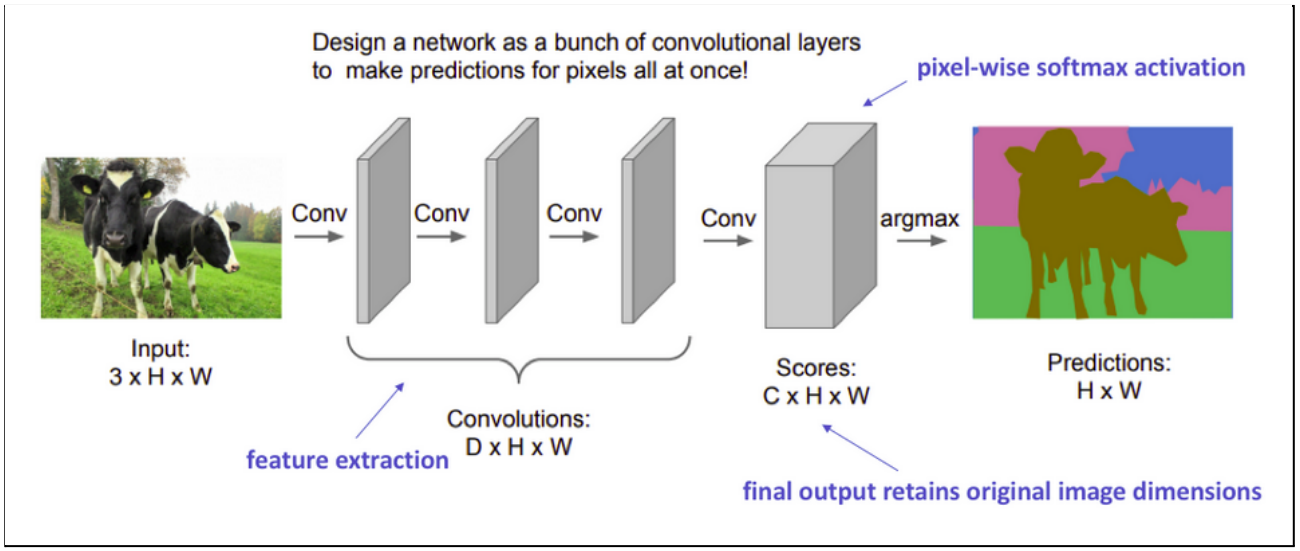

Рисунок 5 - Сегментація зображення за допомогою однорозмірних згорткових шарів.

Для глибоких згорткових мереж попередні шари, як правило, вивчають концепції низького рівня, тоді як пізні шари розробляють більш високі (та спеціалізовані) відображення функцій. Для того, щоб зберегти виразність, нам зазвичай потрібно збільшувати кількість функціональних карт (каналів) у міру заглиблення в мережу.

Це не обов'язково створювало проблему для завдання класифікації зображень, оскільки для цього завдання ми дбаємо лише про те, що містить зображення (а не де воно розташоване). Таким чином, ми могли б полегшити

обчислювальні затрати, періодично зменшуючи вибірку наших функціональних карт за допомогою об’єднання або поетапного здвигу (тобто стискаючи просторову роздільну здатність) без втрати якості результату. Однак для сегментації зображень ми хотіли б, щоб наша модель виробляла семантичне передбачення з повною роздільною здатністю.

Існує кілька різних підходів, які ми можемо використати, щоб збільшити роздільну здатність карти об’єктів. Тоді як операції об’єднання зменшують роздільну здатність шляхом підсумовування локальної області з одним значенням (тобто середнім або максимальним об’єднанням), операції „роз’єднання” збільшують роздільну здатність, розподіляючи одне значення у більш високу роздільну здатність.

Однак згортання зсувів на сьогоднішній день є найпопулярнішим підходом, оскільки вони дозволяють нам розробити готові до навчання нові вибірки. У той час як типова операція згортки буде приймати точковий добуток значень, що перебувають на даний момент у поданні фільтра, і створюватиме одне значення для відповідної вихідної позиції, згортка транспонування по суті робить протилежне. Для згортання транспонування ми беремо одне значення з карти об’єктів із низькою роздільною здатністю і множимо всі ваги у нашому фільтрі на це значення, проектуючи ці зважені значення на вихідну карту об’єктів (Рисунок 6).

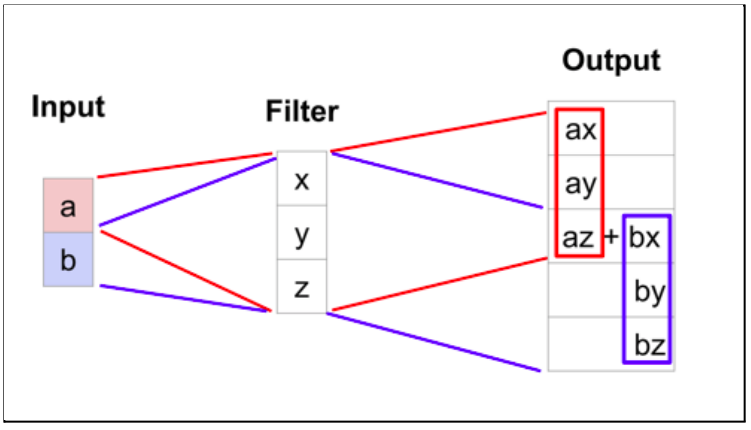

Рисунок 6 - Згортання за допомогою помноження ваг.

2.3 Розпізнавання об’єктів

Це друге завдання фокусується на створенні моделей, які, маючи зображення як вхідні дані, здатні не тільки просторово знаходити об’єкти з різних заздалегідь визначених класів, але й призначати правильний клас кожному з об’єктів, виявлених на зображенні. Для того, щоб знайти різні об'єкти на зображенні, використовуються координати найменшого прямокутника, що їх охоплює. Цей прямокутник відомий як "Bounding Box", оскільки він служить межею для встановлення місця просторового розміщення об'єкта[7].

На рисунку 7 ми можемо побачити результат виконання виявлення об’єктів на зображенні з трьома різними класами: людина, кінь та стілець. Протягом останнього десятиліття глибокі згорткові нейронні мережі здобули велику популярність у галузі виявлення об’єктів. Однак використання класичної Згорткової нейронної мережі із повністю зв’язаним шаром на кінці для виявлення різних об’єктів на зображенні це неможливо через дві основні перешкоди. По-перше, кількість об'єктів, які слід виявити в анімації, раніше невідома, що спричиняє мінливість довжини вихідного шару.

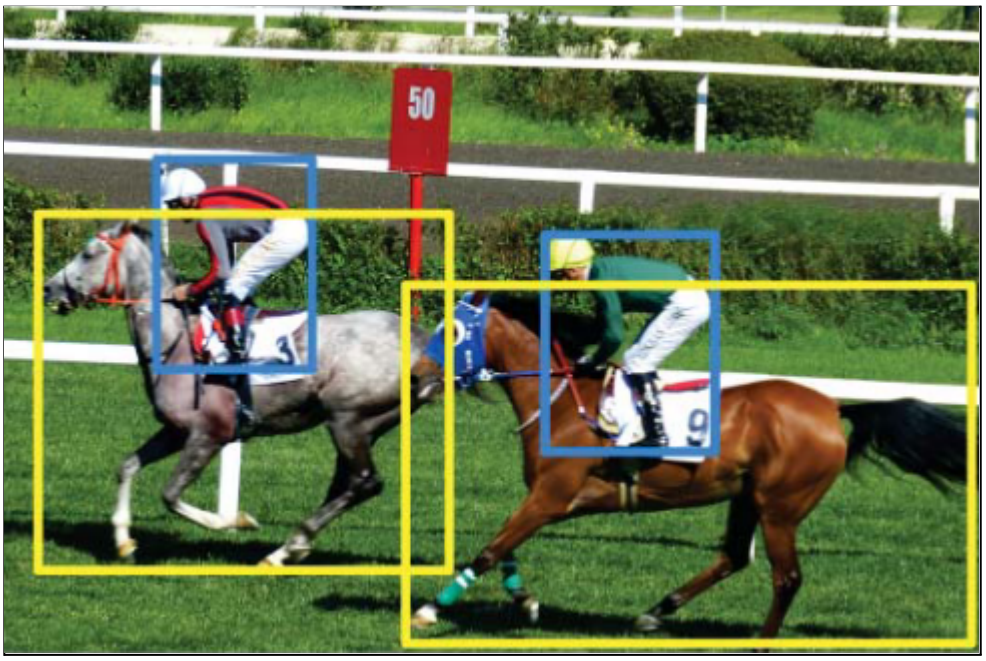

Рисунок 7 - Розпізнавання об’єктів

По-друге, навіть якщо ми вирішили застосувати згорткову нейронну мережу до кожної області, що представляє інтерес для зображення, щоб виявити, чи містить вона об’єкт, об’єкти, як правило, мають різні розміри та просторове розташування всередині картини, таким чином, доводиться вивчати величезну кількість регіонів, що цікавлять . Це робить його обчислювально недоцільним.

Однак дві різні групи методів змогли подолати ці недоліки байдужими способами - одноетапними та двоступеневими методами. Однокроковими методами є ті, що використовують CNN для пересилання вперед для того, щоб визначити розташування об'єктів, що цікавлять, тобто координати обмежувальних ящиків. Це стосується таких моделей, як YOLO, Multibox, AttentionNet, G-CNN або SSD. Ці моделі не потребують вироблення регіональних пропозицій, що робить їх простішими та швидшими. Однак це також спричиняє деякі проблеми у роботі, такі як труднощі при виявленні дрібних предметів або при спробі виконати інші завдання, такі як передбачення маски.

З іншого боку, ми маємо двоступеневі методи, такі як R-CNN, SSP-Net, FastR-CNN, FPN, Швидший R-CNN або R-FCN. Ці моделі використовують регіональний CNN як перший крок. CNN, що базується на регіоні, приймає в якості вхідного зображення зображення і виводить різні цікаві області, де можуть бути розташовані об'єкти на зображенні. Потім вектор з різними характеристиками, витягнутими з кожної запропонованої області, виступає вхідним сигналом для набору повністю зв’язаних шарів, які виводять класифікацію між різними класами та оцінку достовірності. Індекс конфіденційності допомагає скласти рейтинг з пропозиціями, тому враховуються лише найбільш впевнені в собі.

Райян Гіршик та співавт. запропонував R-CNN у 2014 році, щоб вирішити недолік необхідності вибору надмірної кількості регіонів, що представляють інтерес. Для цього R-CNN використовує вибіркову вибірку, як визначено в, щоб генерувати лише 2000 пропозицій регіону на зображення. Потім розмір кожної пропозиції регіону коригується та використовується як вхідна інформація світової

нейронної мережі. Результатом роботи цієї мережі є вектор 4096 ознак, який потім подається у SVM[8] для того, щоб визначити присутність або відсутність об'єкта в цій пропозиції регіону.

Ця модель зуміла перевершити попередні моделі більш ніж на 30% при застосуванні до набору даних PASCAL VOC 2012. Незважаючи на перевищення попередніх моделей, R-CNN мала три явні недоліки. По-перше, вивчення 2000 регіональних пропозицій на зображення призводить до того, що час навчання досить значний. По-друге, час тестування мережі перевищував 40 секунд на зображення, що перешкоджало використанню моделі в реальному часі.

Нарешті, алгоритм вибіркової вибірки був попередньо визначений, що спричинило недостатнє вивчення на цьому першому етапі. Автори R-CNN вирішили деякі недоліки, пропонуючи мережу Fast-RCNN. Їм вдалося зробити мережу швидшою, просто змінивши порядок шарів. Замість того, щоб образити CNN регіонами, що цікавлять, CNN отримує як вхідні дані ціле зображення, формуючи його карту функцій. Потім різні цікаві регіони вибираються безпосередньо з карти об’єктів. Структуруючи мережу таким чином, операція згортки виконується лише один раз, а не 2000 разів.

Незважаючи на вдосконалення, досягнуті Райяном Гіршиком, Шакінг Рен ще побачив місце для вдосконалення та запропонував мережу Faster R-CNN. Вони зрозуміли, що селективний пошук, процес, який використовується у Fast-RCNN та R-CNN з метою формування регіонів, що цікавлять, був повільним та трудомістким, тому вони вирішили запровадити власний алгоритм регіону. Замість використання селективного пошуку вони запропонували використовувати додаткову нейронну мережу, відому як регіональна мережа пропозицій, з метою виявлення регіональних пропозицій.

Сьогодні двоступеневі методи вважаються одним з базових методів виявлення об’єктів, оскільки вони перевершують інші підходи.

2.4 Класифікація зображень

Класифікація зображень - це місце, де комп’ютер може проаналізувати зображення та визначити „клас”, до якого воно потрапляє. (Або ймовірність того, що зображення є частиною «класу».) Клас по суті є міткою, наприклад, «машина», «тварина», «будівля» тощо.

Наприклад, ви вводите зображення вівці. Класифікація зображень - це процес комп’ютера, який аналізує зображення і каже вам, що це вівця. (Або ймовірність того, що це вівця.)

Для нас класифікація зображень не становить великої праці. Але це чудовий приклад парадоксу Моравека, коли мова заходить про машини. (Тобто речі, які ми вважаємо легкими, важкі для ШІ.)

Рання класифікація зображень спиралася на необроблені піксельні дані. Це означало, що комп’ютери розбиватимуть зображення на окремі пікселі. Проблема в тому, що дві картинки одного і того ж можуть виглядати дуже по-різному. Вони можуть мати різне тло, ракурси, пози тощо. Це спричинило проблему для комп’ютерів правильно «бачити» та класифікувати зображення як зображено на рисунку 8.

Рисунок 8 - Класифікація зображень

Поглиблене навчання - це тип машинного навчання; підмножина штучного інтелекту (ШІ), що дозволяє машинам вчитися на даних. Поглиблене навчання передбачає використання комп’ютерних систем, відомих як нейронні мережі. У нейронних мережах вхід фільтрує через приховані шари вузлів. Кожні ці вузли обробляють вхідні дані та передають свої результати наступному шару вузлів. Це повторюється, поки не досягне вихідного рівня, і машина надасть відповідь.

Існують різні типи нейронних мереж, засновані на тому, як працюють приховані шари. Класифікація зображень при глибокому навчанні найчастіше включає згорткові нейронні мережі або мережі CNN. У CNN вузли в прихованих шарах не завжди діляться своїми вихідними даними з кожним вузлом наступного шару (відомого як згорткові шари).

Поглиблене навчання дозволяє машинам ідентифікувати та вилучати особливості із зображень. Це означає, що вони можуть навчитися особливостям, які потрібно шукати на зображеннях, проаналізувавши безліч знімків. Отже, програмістам не потрібно вводити ці фільтри вручну.

2.5 Сегментація екземплярів

Це останнє завдання - природна еволюція семантичної сегментації та виявлення об’єктів. Виявлення об’єктів має на меті виявити різні об’єкти на зображенні, тоді як семантична сегментація має на меті передбачити клас кожного пікселя, створюючи маску для кожного присутнього класу. У сегментації екземплярів ідея полягає у вирішенні одночасно обох завдань, знаходженні екземплярів об’єктів з точністю на рівні пікселів. Кожен піксель зображення класифікується в одному із заздалегідь визначених класів, як у семантичній сегментації. Однак виявлення об'єкта також виконується. Це означає, що кожна отримана маска буде містити не всі об'єкти з одного класу, а лише один екземпляр одного класу. Різні об’єкти з того самого класу матимуть різні маски, як це видно на малюнку 9.

Рисунок 9 - Сегментація екземплярів

Це завдання дозволяє нам знаходити різні екземпляри одного класу, що відображаються на зображенні. Як і в попередніх завданнях, використовуючи

згорткові нейронні мережі як загальну базу, протягом багатьох років було запропоновано кілька фреймворків, спрямованих на вирішення цієї проблеми.

Харіхаран був одними з перших, що вирішували завдання сегментації екземпляра, називаючи це одночасним виявленням та сегментацією. Їх ідея полягала в тому, щоб розширити архітектуру R-CNN, щоб отримати маску кожного екземпляра, а не лише обмежувального вікна. Їх фреймворк розпочався з генерації різних пропозицій. Вони застосували Multiscale CombinatorialGrouping, щоб запропонувати 2000 пропозицій регіону на зображення.

На другому етапі CNN витягував особливості з усіх пропозицій різних регіонів. Потім була підготовлена машина підтримки вектора, використовуючи вилучені функції, отримуючи ймовірність для кожного із заздалегідь визначених класів. Нарешті, вони застосували не максимум придушення пропозицій. Таким чином, вони зберегли лише одну пропозицію щодо інстанції.

Через рік автори тієї ж статті вдосконалили свою модель. Вони зрозуміли, що використання лише виходу останнього шару як представлення ознак спричиняє втрату важливого просторового розташування. З іншого боку, попередні шари мали багатшу просторову інформацію. Щоб вирішити цей недолік, вони придумали концепцію гіперколонок: замість використання лише результату останнього рівня мережі кожен піксель визначався вектором активації всіх блоків CNN вище цього пікселя. Кроки, один початковий крок для регіональних пропозицій та секундний крок, коли ці пропозиції класифікуються.

Пінейро запропонував новий підхід до формування регіональних пропозицій. Його алгоритм, відомий як DeepMask, приймає в якості вхідного зображення частину зображення і генерує два різні виходи. З одного боку, один вихід - це маска класу. З іншого боку, оцінка, яка визначає ймовірність виправлення, що містить повний об'єкт із центром. Ядром моделі є ConvNet, яка використовує функцію витрат для оптимізації обох результатів одночасно. Ця модель зуміла зменшити кількість пропозицій, одночасно покращивши продуктивність, порівняно з попередніми моделями.

Як уже згадувалося раніше, повністю згорткові мережі виявилися дуже успішними при використанні для семантичної сегментації. Однак вони не надають інформації на рівні екземпляра.

Дай запропонував InstanceFCN, нову модель, засновану на FCN, здатну сегментувати різні екземпляри одного класу. Традиційний вихід FCN - це бальна карта з тим же розміром, що і вхідне зображення. Кожне значення вихідної матриці є результатом класифікації відповідного пікселя у вхідному зображенні. Автори цієї статті вводять поняття відносних позицій. У їх моделі кожен піксель вихідних даних є класифікатором відносних позицій екземплярів. Це означає, що класифікація проводиться не лише між класами, а й з використанням відносних позицій, таких як "ліва сторона" або "низ". Таким чином, кожен піксель класифікується з урахуванням того, є він частиною чи ні відносного положення екземпляра. Відносні позиції визначаються за допомогою сіток різного розміру. Завдяки використанню локальної когерентності (передбачення для одного і того ж пікселя кожного разу, коли ковзає вікно) не тільки обчислювальні витрати в цій моделі нижчі, ніж у DeepMask, це також вигідно від значного зменшення щодо кількості параметрів, які потрібно тренувати. Однак InstanceFCN також представив деякі явні недоліки.

InstanceFCN не враховував семантичні категорії, виконував сегментацію та виявлення на різних етапах, і рішення не було наскрізною моделлю. Застосовувані розсувні вікна мали фіксований розмір, і процес пошуку екземплярів у різних масштабах витрачав час. Лі з колегами запропонував повністю згорнуту сегментацію екземплярів (FCIS), першочергове рішення для завдання сегментації екземплярів. Видобуті функції та таблиці показників розподіляються між під завданням сегментації та виявлення, зменшуючи кількість параметрів. Більше того, він використовує пропозиції обмежувальної рамки замість розсувних вікон, роблячи процес ефективнішим. Незважаючи на вдосконалення, здійснені Лі, модель FCIS все ще мала деякі недоліки. Це показало деякі проблеми та неточності, коли два або більше екземплярів перекривались, створюючи хибні краї, навіть коли текстура фону є однорідною.

Інша галузь досліджень підходила до проблеми з іншої точки зору. Спочатку вони виконували семантичну сегментацію, а потім намагалися вирізати різні випадки для кожного класу. У Кириллов та співавт. запропонували Instance Cut. Їх структура мала два різні елементи. З одного боку, вони вирішили завдання семантичної сегментації, використовуючи стандартну CNN та отримавши інстанційно-агностичне рішення. З іншого боку, вони використовували інший CNN, щоб отримати кордони різних інстанцій. Потім вихід цих двох CNN поєднується.

Між різними перевагами цього фреймворку ми можемо виявити, що навчені дві різні моделі, одна для семантичної сегментації та інша для виявлення краю екземпляру. Це дозволяє уникнути необхідності мати справу з глобальними особливостями екземплярів різних класів. Це також допомагає зробити фреймворк більш модульним.

Досягнення семантичної сегментації або поля виявлення краю екземпляра можуть бути безпосередньо реалізовані в цьому рішенні. Основне обмеження цього методу полягає в тому, що об'єкти, які не пов'язані між собою, не можуть розглядатися як такі, що належать одній і тій же інстанції. Інші автори вирішили використати потенціал класичної техніки групування, відомої про перетворення вододілу. Ідея цього перетворення полягає в тому, що зображення у відтінках сірого можна розглядати як топографічну поверхню. Потім поверхня заливається з різних точок, розташованих у її мінімумах. Якщо ми не даємо воді, що надходить з різних джерел, зустрічатися, ми отримуємо сегментацію різних компонентів зображення.

Бай та ін. пропонували використовувати глибоку згорткову нейромережу для вивчення енергії перетворення з метою отримання сегментацій, що містять лише один примірник. Таким чином, різні екземпляри можна сегментувати, просто скорочуючи одиничні енергетичні рівні. Одним з основних обмежень цієї системи є неможливість розглядати ці висновки. Вони пропонували використовувати мережі послідовних групувань (SGN) з метою сегментації ефективності. SGN використовують конкатенацію нейронних мереж, кожна з них вирішує проблему

різної семантичної складності. Вони поділяють завдання сегментації екземпляра на більш легкі підзадачі, і кожна з них виконує іншу нейронну мережу. Метою першої мережі є групування пікселів уздовж різних піксельних рядків і стовпців на зображенні, передбачаючи розташування точок зупинки об'єкта. Потім ці групи утворюють сегменти рядків. Відрізки лінії є входом другої мережі. Його мета - згрупувати різні відрізки ліній у зв’язані елементи. Після цього другого кроку остання нейронна мережа об'єднує різні компоненти, що генерують маски екземплярів об'єкта. Двома основними обмеженнями цієї основи є низька продуктивність при сегментування невеликих екземплярів та випадкове сегментування декількох екземплярів разом, коли вони перекриваються.

Маска R-CNN була запропонована як продовження Faster R-CNN. Автори статті мали ідею додати додаткову гілку до архітектури швидшого R-CNN, як ми бачимо на рисунку 10 вихідна гілка була призначена для класифікації та регресії обмежувальної рамки, отримання в якості вихідних обмежувальних рам і міток класів об'єктів, які вони містять.

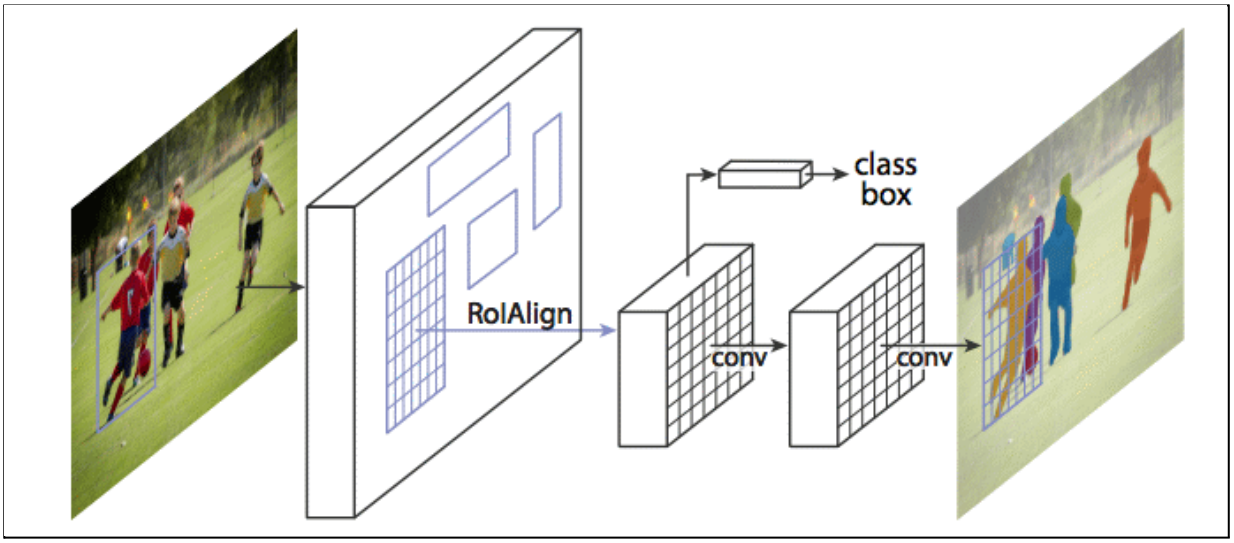

Рисунок 10 - Mask R-CNN

З іншого боку, нова додаткова гілка мала на меті передбачити маски сегментації для кожної області інтересу, надаючи як вихідну інформацію маску екземпляра, що міститься в кожному обмежувальному полі. Mask R-CNN вдалося

перевершити всі попередні найсучасніші моделі, що використовувались для завдання сегментації екземплярів при використанні набору даних COCO.

3 ОСНОВНІ МЕТОДИ ЗГОРТКОВОЇ НЕЙРОННОЇ МЕРЕЖІ

Перші згорткові нейронні мережі (CNN) були введені в 1980 році, коли Куні-хіко Фукусіма запропонував неокогнітрон. Неокогнітрон був натхненний зоровою нервовою системою хребетних у ссавців і є, мабуть, однією з перших запропонованих глибоких нейронних мереж. В останні роки варіанти CNN значно покращують сучасний рівень сегментації та виявлення зображень.

CNN потрапляє до категорії глибокі нейронні мережеві піски, як правило, мають ручну архітектуру для автоматичного вилучення функцій із повністю зв’язаними шарами в кінці для класифікації. Цю структуру можна побачити на малюнку 11, де показано CNN із згортковими шарами, перемежованими з об'єднуючими (субдискретизаційними) шарами. Звитки витягують особливості з допомогою зважених сум, де ваги засвоюються під час тренування. Підвибірка зменшує вибірку роздільної здатності карти особливостей за допомогою техніки, яка називається макс. Це спричиняє деяку втрату просторової роздільної здатності, але, в свою чергу, розширює поле сприйняття та робить мережу меншою, швидшою та менш схильною до надмірного пристосування. В якості субдискретизації виконується за допомогою заздалегідь визначених функцій, на етапі субдискретизації не відбувається навчання. Згорнутий і субдискретизаційний шари створюють особливості карт, що містять багаті на семантику дані для класифікації вхідного зображення.

У типовому CNN на малюнку 11, остання карта об'єктів сплющена і з'єднана з повністю з'єднаними шарами, що, як правило, класифікує вектор на вихід softmax. Цей тип CNN характерний для класифікації зображень. Для більш складних завдань, таких як семантична сегментація, виявлення об’єктів та сегментація екземплярів, потрібна більш складна архітектура. Складні архітектури, які мають значення для цього проекту, обговорюються наступних підрозділах.

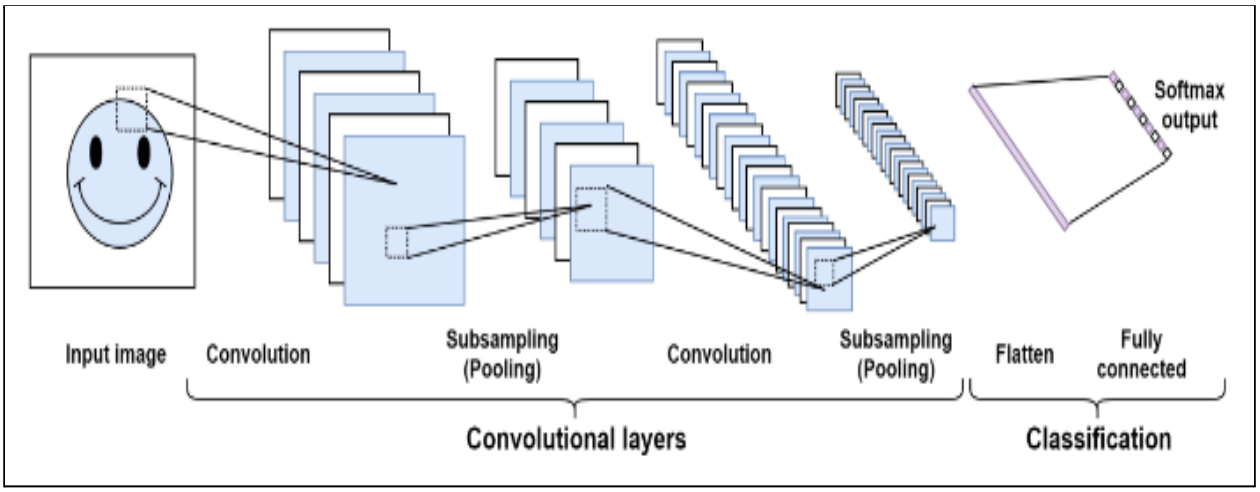

Рисунок 11 - Типова згорткова нейронна мережа

3.1 FCN - повністю згорнута нейронна мережа

FCN або повністю згорнуті мережі зазвичай використовуються для семантичної сегментації і вперше були введені Лонгом та співавторами у 2015 році. Що відрізняє FCN від нормального CNN, це те, що замість класифікації вектора ознак з MLP в кінці FCN "розшифровує" ознаки, вилучені згорткою та об'єднанням шарів на карту сегментації. Оскільки вилучення функцій у CNN є своєрідним кодуванням зображення в функції, FCN, по суті, є мережею кодування / декодування. Кодування (вилучення ознак) ефективно розмежовує класи, і існує вибірка, щоб відтворити ознаки на карті сегментації з однаковою роздільною здатністю вихідного зображення. Підвищення вибірки здійснюється за допомогою перенесеної згортки (іноді її називають розгорткою або згортками назад), яка, як випливає з назви, є оберненою до згорток. Оскільки в кінці FCN[9] немає повністю зв’язаного шару, а транспоновані конволюції є інваріантами розміру вхідного сигналу, FCN має практичну перевагу, оскільки можна взяти будь-який розмір зображення як вхідний.

Ваги, що впливають на проекції з транспонованих звивин, вивчаються під час тренувань, а це означає, що мережа по суті навчається декодувати (підвищуючи вибірку) своє власне кодування функцій (зменшення розмірів) з не глибокими шарами. Глибинні шари містять грубу та загальну інформацію, яка

добре підходить для класифікації чогось. Не глибокі шари містять дрібнозернисту локальну інформацію, яка добре вирішує, де знаходяться ці класифікації. Поєднуючи локальну та загальну інформацію, можна створити більш детальну карту сегментації в підбірці даних. див. рисунок 12 для візуалізації архітектури FCN.

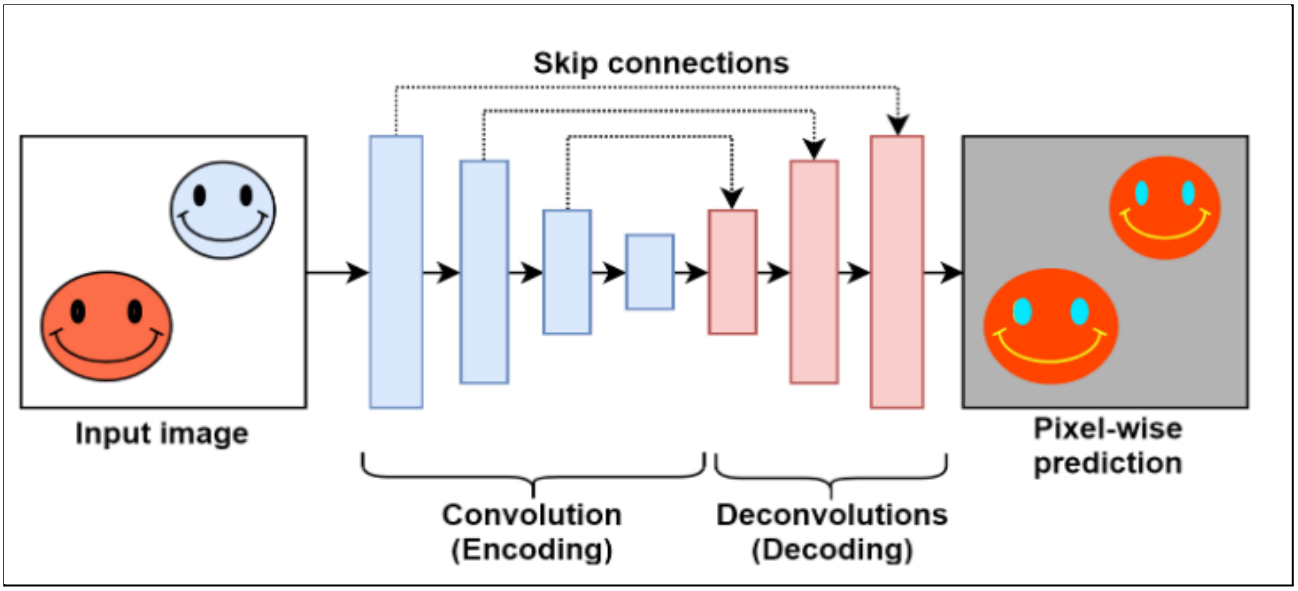

Рисунок 12 - FCN структура кодування та декодування

3.2 Faster R-CNN

Регіони з функціями CNN (R-CNN) - це мережі, призначені для виявлення об'єктів. Як оригінальний R-CNN, запропонований Гіршиком була обчислювальною дорогою мережею, докладено зусиль, щоб зробити більш ефективну версію. Перша ітерація отримала назву Fast R-CNN, що дозволило підвищити ефективність, щоб виставити регіональні пропозиції як нове вузьке місце. Це мотивувало наступну ітерацію, faster R-CNN, яка в основному є Fast R-CNN із підключеною до мережі мережею регіональних пропозицій (RPN). RPN створює регіональні пропозиції, які називаються регіоном інтересів (ROI), а Fast R-CNN передбачає, який клас повинен бути пов'язаний з ROI, а також обмежувальні рамки, щоб відповідати ROI об'єкту. RPN та Fast R-CNN далі обговорюються окремими пунктами.

3.2.1 RPN - Регіональна мережа пропозицій

Як уже встановлено, RPN пропонує ROI, які передаються Fast R-CNN. Це робиться шляхом розміщення фіксованих обмежувальних коробок, які називаються якорями, по всьому зображенню, а потім передбачається, чи є предмет всередині якоря. RPN також попередньо визначає дельта-значення для регулювання якоря відповідно до об'єкта. RPN вдається обчислити всі якорі обчислювально дешево, оскільки він використовує витяг функцій з "задньої кістки" (див. Малюнок 11). Остання карта функцій з магістралі передається на RPN, який аналізує її повністю згортково. Перший шар RPN - це згортковий шар а 3 × 3 з кроком 1 (більші кроки можна використовувати для меншої кількості якорів), який ковзає по карті об'єктів. Вихідні дані з першого шару потім розгалужуються на два різні згорткові шари 1 × 1, як це видно на малюнку 13. Один шар 1 × 1 призначений для класифікації, а один - для регулювання обмежувальної коробки. Рівень класифікації має дві тисячі фільтрів, а шар регресії обмежувальної рамки має 4 тисячі фільтрів, де визначається кількість якорів на точку. У цьому сенсі згортки 1 × 1 служать для вилучення значень довжини 4 або 2 тисячі із проміжного шару.

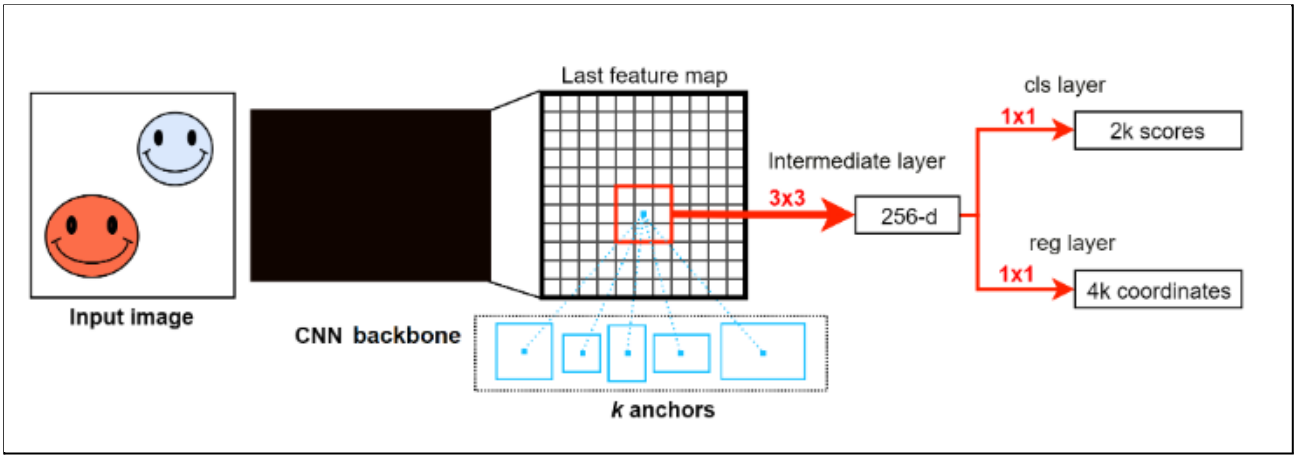

Рисунок 13 - Архітектура RPN

Потім класифікаційний шар проходить через шар softmax, щоб отримати 2 оцінки softmax на якір, один бал - це ймовірність якоря є фоном, а один із них - переднього плану. Регресійний шар, що обмежує, дає 4 бали за якір: Δxcenter, Δycenter, Δwidth, Δheight. Як можна побачити на малюнку 9, передбачення робляться один раз для кожної точки на карті об’єктів (якщо крок = 1). Це призводить до прогнозованих якорів W H k сумма, де W є шириною карти особливостей і висотою карти об'єкта. W та H можна додатково визначити за:

W = w/r та H = h/r,

де w та h вихідна ширина та висота вхідного зображення та r коефіцієнт субдискретизації магістральної архітектури.

Якорі для регіональної пропозиції зображені на малюнку 14. Якорі, класифіковані як передній план, - це передбачувані рентабельності інвестицій, але оскільки існує так багато якорів, багато з них перекриваються і передбачають один і той же об’єкт.

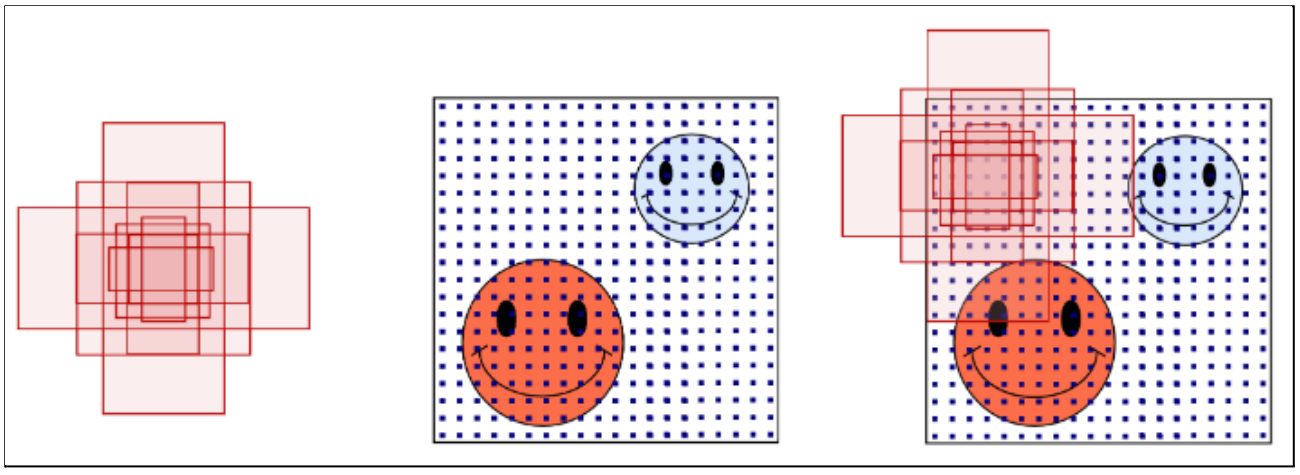

Рисунок 14 - Візуалізація якорів: Якіри (ліворуч), точки на зображенні (посередині), анкери в одній точці (праворуч).

Для зменшення надмірності якіри, що перекриваються, відфільтровуються, використовуючи не максимальне замовчування (NMS). НМС працює шляхом

сортування списку пропозицій за балами, а потім пропозицій із нижчим балом між пропозиціями з IOU (Пересічення понад об'єднаннями), що перевищує певний поріг викидаються. IOU між двома пропозиціями p_1 та p_2 визначається як:

$$IOU(p_1 p_2) = \frac{p_1 \cap p_2}{p_1 \cup p_2}$$

Після фільтрації нових країн-членів найвищі оціночні пропозиції зберігаються та надсилаються на Fast R-CNN. У статті "Швидше R-CNN" автори використовували N = 2000 для навчання та N = 1000 для умовиводу. RPN можна навчати індивідуально, але було показано, що підготовка його спільно з усією мережею, яке також називають наскрізним навчанням, покращує середню точність. RPN тренується з функцією журналу втрат для класифікації та гладкості для передбачення обмежувального вікна.

3.2.2 Fast R-CNN - Швидка згорткова нейронна мережа

ROI, отримані від RPN, є обмежувальними коробками різного розміру. Завдання Fast R-CNN - класифікувати кожну ROI та додатково налаштувати обмежувальну коробку відповідно до об'єкта. Це дуже схоже на те, що робить RPN з якорями, але замість того, щоб класифікувати, існує об'єкт чи ні, Fast R-CNN класифікує серед усіх доступних класів. Швидкий R-CNN починається з шару об’єднання ROI[10], який приймає запропоновані ROI для вилучення фіксованих карт об’єктів з останньої карти об’єктів на магістралі.

Об’єднання рентабельності інвестицій важливо, оскільки розміри рентабельності інвестицій можуть змінюватися, але класифікація вимагає фіксованого вхідного розміру, типового для повністю пов’язаних шарів. Фіксована карта об’єктів, створена шаром об’єднання ROI, відображається у вектор об’єкта за допомогою повністю зв’язаних шарів. Цей вектор використовується двома різними гілками, які паралельно виробляють м'який вивід усіх можливих класів

разом з 4 значеннями для кожного класу, що визначає уточнені координати для рентабельності інвестицій, щоб розмістити його навколо цього класу. Faster R-CNN навчається за допомогою функції втрати журналу для класифікації та згладжування L1 для передбачення обмежувального блоку. Це майже точно такі ж функції втрат, що використовуються для RPN, оскільки дві насправді виконують подібні завдання. Втрати дещо відрізняються, оскільки faster R-CNN обробляє декілька класів, порівняно з RPN, який обробляє лише об'єкт / відсутність об'єкта.

3.3 Маска R-CNN

Наприкінці 2017 року запропонували Mask R-CNN, який на той час був найсучаснішим для сегментації екземплярів. На сьогоднішній день Mask R-CNN, як і раніше, є однією з найпопулярніших моделей сегментації екземплярів, і її перемагають лише рішення, побудовані поверх неї. Mask R-CNN - це, по суті, розширення певної швидшої версії R-CNN, яка використовує магістраль Feature Pyramid Network (FPN), запропоновану Лінетом та іншими.

Розширення від Faster R-CNN до Mask R-CNN зустрічається на останньому етапі мережі, тій частині, яка передбачає класи та зміщення обмежувальної рамки. Тут Mask R-CNN додає додаткову паралельну гілку, яка передбачає маску сегментації, яка виконана з FCN. Оскільки прогнозування маски здійснюється паралельно з класифікацією та регресією обмежувального вікна, Mask R-CNN лише додає невелику накладну вартість, порівняно з більш швидкою R-CNN, і робить конвеєр відносно простим. Маска попередньо продиктована за допомогою FCN з розмірним висновком K × m × m для K-класів. Це означає, що маска завжди передбачається для кожного класу, а не тільки для класифікованої маски.

Потім мережа покладається на гілку класифікації, щоб вибрати правильну маску. Це розділення передбачення класу та передбачення маски дозволяє передбачати маски без змагань серед класів. Відповідно функція збитків для maskonly враховує правильну маску класу істинної землі. Тренування проводиться з функцією втрат, дуже подібною до ідентичної швидшій R-CNN, але з

додаванням втрати маски. Втрата маски визначається як середня втрата бінарної перехресної ентропії між передбачуваною та основною маскою істини.

У Faster R-CNN для вилучення карти об’єктів фіксованого розміру для рентабельності інвестицій використовувався рівень об’єднання ROI, ROI Pool. Через те, як був розроблений цей ROI Pool, мало місце невідповідність фактичної рентабельності інвестицій і представлення рентабельності на карті об'єктів. Хоча це не було великою проблемою для класифікації та доопрацювання обмежувальної рамки, воно виявилося шкідливим для точного прогнозування маски. Щоб пом'якшити це, автори Mask R-CNN запропонували ROI Align, новий шар, який правильно вирівняв карту об'єктів з ROI, використовуючи дволінійну інтерполяцію, замість квантування ROI до деталізації карти особливостей. На малюнку 15 показано розширення від Faster R-CNN до Маска R-CNN.

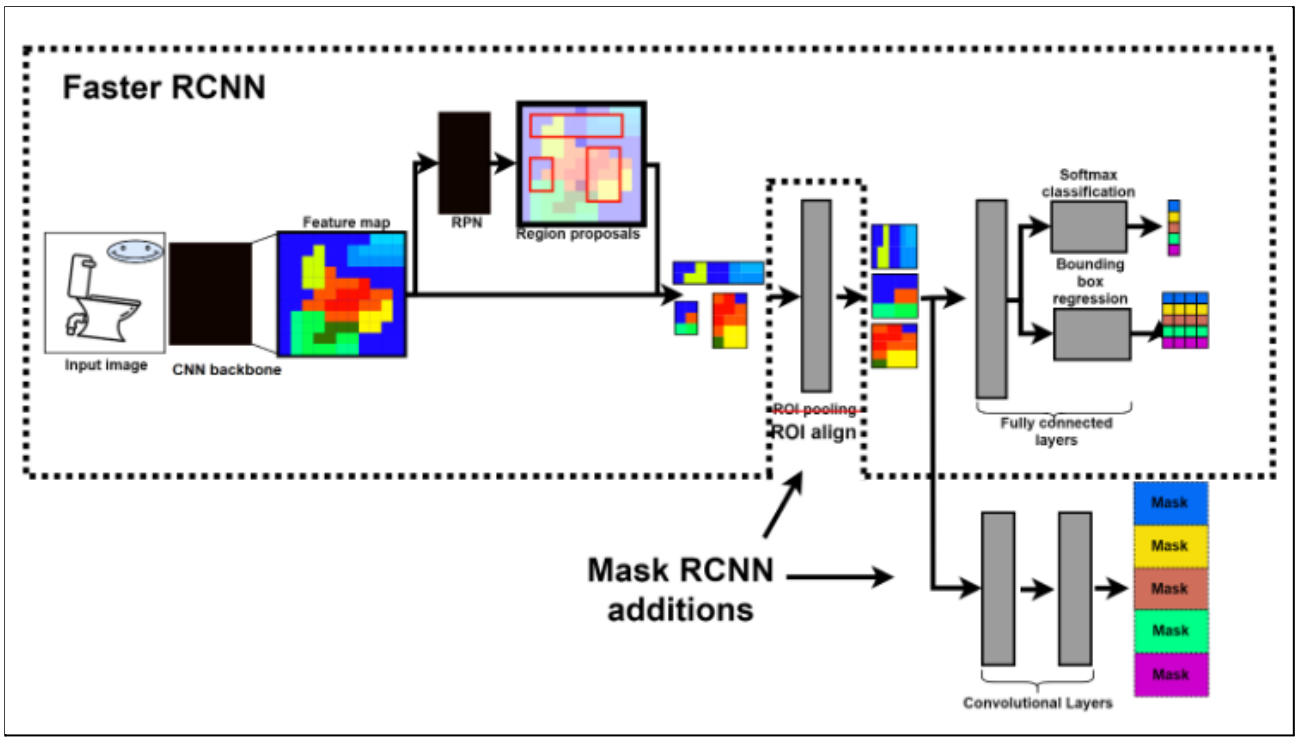

Рисунок 15 - Представлення Mask R-CNN щодо Faster R-CNN

3.4 FPN - Функція пірамідних мереж

Мережа пірамід об’єктів (FPN) - це екстрактор функцій, який приймає одномасштабне зображення довільного розміру як вхідне та виводить пропорційно розміщені карти об’єктів на декількох рівнях повністю згорнуто. Цей процес не залежить від основної конволюційної архітектури. Тому він виступає загальним рішенням для побудови пірамід об’єктів усередині глибоких згорткових мереж, що використовуються в таких завданнях, як виявлення об’єктів.

Будівництво піраміди передбачає шлях знизу вгору та шлях зверху вниз.

Шлях знизу вгору - це пряме обчислення магістральної системи ConvNet, яка обчислює ієрархію об’єктів, що складається з карт об’єктів у декількох масштабах із кроком масштабування 2. Для піраміди об’єкта для кожного етапу визначається один рівень піраміди. Результат останнього шару кожного етапу використовується як еталонний набір функціональних карт. Для ResNets ми використовуємо активації функцій, що виводяться останнім залишковим блоком кожного етапу.

Функціональна пірамідна мережа (FPN) була запропонована Ліном та співавторами та вирішувала проблему вилучення особливостей, яку демонструють звичайні CNN. Питання практично те саме, глибокі шари в CNN містять семантично сильну інформацію, але мають низьку просторову роздільну здатність. Цю просторову роздільну здатність можна знайти в неглибоких шарах, які містять точну та локальну інформацію, але з семантично слабкими особливостями. Запропонована FPN - це мережа, яка поєднує шари знизу вгору та зверху вниз, див. Рисунок 16. Шлях знизу вгору по суті такий самий, як і обчислення вперед типового CNN. Для кожного кроку в піраміді просторова розмірність зменшується на 1/2, оскільки підвідбір здійснюється з кроком (2,2).

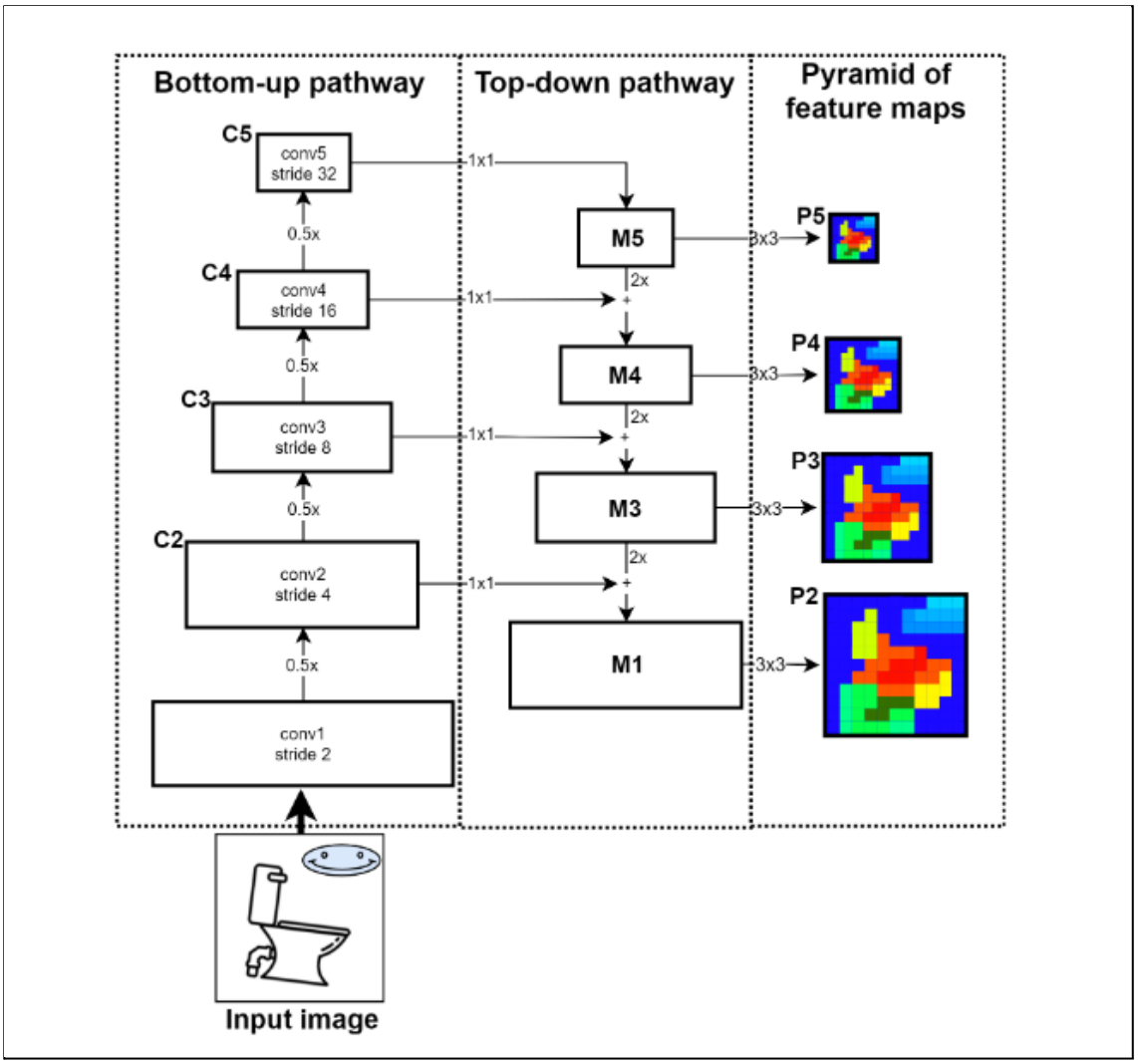

Рисунок 16 - Візуалізація FPN

Шлях зверху вниз побудований шляхом передискретизації карт функцій з підбіркою найближчих сусідів. Це підсумовується з бічним з'єднанням, яке проходить через звивини 1 × 1, щоб зменшити розміри каналу. Нарешті, для зменшення згладжування від передискретизації зроблено згортку 3 × 3. Коли FPN використовується як магістраль у Faster R-CNN або Mask R-CNN, усі шари (P2, P3, P4 і P5) використовуються в ROI Pooling / ROI Align для вилучення характеристик з різних рівнів, залежно від розміру рентабельності інвестицій.

4 МАТУВАННЯ ЗОБРАЖЕННЯ ДЛЯ ЗМІНИ ФОНУ

Матування зображення - це велике поле, але в цій роботі я зосереджуюсь на виділенні переднього та заднього планів. Отримавши зображення I, ми хочемо створити “ammate”(значення прозорості αна кожному пікселі, з αi∈ [0,1]), таким чином, щоб на кожному пікселі ми могли деконструювати значення кольору Ii у сумі двох зразків, один із кольору Fi переднього плану та один із а колір тла Bi. Потім ми беремо для отримання кольорового значення на кожному пікселі на нашому початковому зображенні.

$$I_i = \alpha_i F_i + (1 - \alpha_i) B_i$$

Ця проблема обмежується "ескізом", який користувач часто називає не пріоритетним, із зазначенням областей зображення, які, як відомо, знаходяться або на передньому плані, або на задньому плані. За достатньої кількості таких обмежень проблему можна визначити так щоб отримати точну та корисну матовість.

4.1 Огляд принципу роботи матування зображення

Передній план - це частина виду або картинки, яка є найближчою до вас, коли ви дивитесь на неї. Ми, люди, зазвичай добре розрізняємо об’єкти переднього плану на зображеннях від фону. Оскільки алгоритми комп’ютерного зору ставали дедалі кращими у вирішенні візуальних завдань, цілком природно, що ми хочемо розвантажити завдання розділення переднього плану на машини.

Ми знаємо, що передній план можна відокремити, застосовуючи алгоритми семантичної сегментації. Але це не означає що проблема практично вирішена. Основна відмінність між семантичною сегментацією та матуванням зображень полягає в тому, що в останньому ми хочемо, щоб наш результат був надзвичайно

точним і безперервним. Мітки на пікселі, які визначають, чи кожен конкретний піксель належить до переднього або заднього плану, вже недостатньо хороші для нас, оскільки для багатьох природних об’єктів, таких як волосся або хутро, відповідь може бути щось середнє. Більшість алгоритмів семантичної сегментації навчаються, не роблячи жодного акценту на настільки точних краях, тому ці моделі не можуть дати нам бажаного результату. Нам потрібне спеціальне рішення для нашого випадку, а саме, матування зображення.

Тепер давайте розглянемо алгоритм матування зображення SOTA, щоб побачити, як він працює.

Давайте розпочнемо підхід FBA Matting[11] на реальних зображеннях. Щоб застосувати алгоритм FBA Matting, нам спочатку потрібно створити тримап. Ми використовуватимемо попередньо навчений DeepLabV3 для створення маски сегментації з імовірностями кожного пікселя, що належить до класу переднього плану. Після цього ми використаємо ряд операцій розширення, щоб позначити пікселі меж та пікселі з низькою ймовірністю переднього плану як невідомі. На жаль, такий підхід може призвести до неточного матування. Ви можете побачити різницю між міченим та створеним тримапом на рисунку 17.

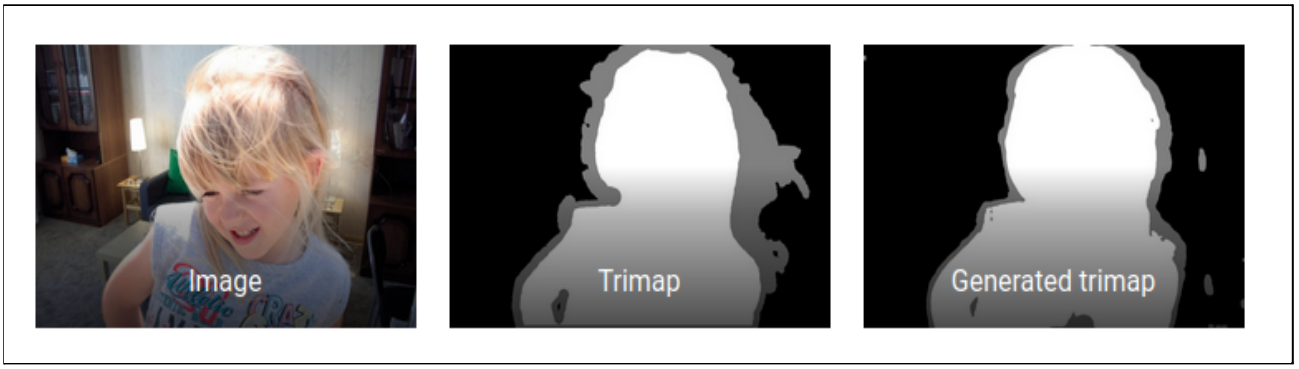

Рисунок 17 - Приклад роботи матування зображення

4.2 Математичне формулювання матування

Добре, зараз ми побачили, що матування може принести досить хороші результати, а тепер давайте розглянемо як матування працює. Математично питання сформульовано як рівняння складання:

$$C_i = \alpha_i F_i + (1 - \alpha_i) B_i$$

Якщо альфа дорівнює 1 для пікселя i, то це чистий піксель переднього плану. Розв’язування рівняння складання - це неправильно поставлене питання, оскільки ми маємо лише 3 рівняння для 7 невідомих. Протягом останніх кількох років декілька методів, що базуються на глибокому навчанні, підвищили рівень техніки в області матування зображень. Є багато успішних підходів, таких як глибоке матування зображення, IndexNet Matting, GCA Matting, щоб назвати лише декілька. Сучасний рівень техніки - це F, B, Alpha Matting. Далі розглянемо як працює тримап, який використовує матування зображення.

4.3 Проблема тримапу

Потрібно пам’ятати, що головним акцентом проблеми матування є дуже точне відокремлення переднього плану від фону. Таким чином, матування насправді мало цікавить, який тип об’єкта зображений на рисунку 18. Ця проблема відокремлена від фактичної семантичної сегментації, і тому багато алгоритмів матування вимагають сегментації маски або трімапу - як вхідні дані. В основному, тримап - це груба сегментація зображення на три типи регіонів: певний передній план, невідомий, певний фон.

Рисунок 18 - Тримап зображення

Тримап зазвичай поєднується з відповідним зображенням, а потім ця 4-канальна конкатенація використовується як вхід для моделі. Однак ця вимога накладає суворі обмеження щодо використання моделі. Слід якось заздалегідь створити тримап із зображення. Часто очікується наявність приємних вручну створених тримапів, щоб запустити алгоритм матування. На щастя, ми можемо використовувати сегментацію для досягнення цілей, щоб передбачити тримап за вихідним зображенням. Ми генеруємо тримап, використовуючи маску сегментації, створену попередньо навченим DeepLabV3[12].

5 ОПИС ЕКСПЕРИМЕНТАЛЬНИХ ДОСЛІДЖЕНЬ

5.1 Планування експерименту

Для нашої задачі було обрано декілька основних методів сегментації такі як: R-CNN, Fast R-CNN, Faster R-CNN та Mask R-CNN[13]. Усі ці методи використовують згорткову нейронну мережу як базовий рушій алгоритма, але кожен з методів так чи інакше доповнює спосіб отримання кінцевого результату.

Більшість алгоритмів сегментування спрямовані на знаходження різних класів об’єктів, але у нашому випадку ми зацікавлені лише у пошуку людини та лише однієї на нашому зображенні. Тому ми будемо використовувати декілька датасетів для нашої моделі, а також спробуємо використати лише під вибірку з датасету COCO.

Набір даних COCO, що означає “Загальні об’єкти в контексті”, являє собою набір складних високоякісних наборів даних для комп’ютерного зору, переважно найсучасніших нейронних мереж. Це ім'я також використовується для назви формату, що використовується цими наборами даних.

Формат цього набору даних автоматично розуміється удосконаленими бібліотеками нейронних мереж, наприклад Facebook Detectron2. Існують навіть інструменти, створені спеціально для роботи з наборами даних у форматі COCO, наприклад COCO-анотатор та COCO api. Розуміння того, як представлений цей набір даних, допоможе використовувати та модифікувати існуючі набори даних, а також створити власні. Зокрема, нас цікавлять файли анотацій, оскільки повний набір даних складається з каталогу зображень та файлу анотацій, забезпечуючи метадані, що використовуються алгоритмами машинного навчання.

Також розглянемо такий датасет як PASCAL VOC. Набір даних PASCAL Visual Object Classes (VOC) 2012 містить 20 категорій об’єктів, включаючи транспортні засоби, домогосподарства, тварин та інше: літак, велосипед, човен, автобус, машина, мотоцикл, поїзд, пляшка, стілець, обідній стіл, рослина в горщику, диван, Телевізор / монітор, птах, кішка, корова, собака, кінь, вівця та

людина. Кожне зображення в цьому наборі даних містить анотації сегментації на рівні пікселів, анотації обмежуючих полів та анотації класів об’єктів. Цей набір даних широко використовується як еталон для виявлення об’єктів, семантичної сегментації та класифікації. Набір даних PASCAL VOC розділений на три підмножини: 1464 зображення для навчання, 1449 зображень для перевірки та приватний набір тестування.

5.2 Тестування швидкості методів сегментації

Під час тестування ми проводимо вибірковий пошук на тестовому зображенні, щоб витягти близько 2000 пропозицій регіону (ми використовуємо «швидкий режим» selective search у всіх експериментах). Ми деформуємо кожну пропозицію та передаємо її через CNN в ордері для обчислення характеристик. Потім для кожного класу ми кожного разу витягуємо вектор об’єктів, використовуючи SVM, навчений для цього класу. Враховуючи всі забиті області на зображенні, ми застосовуємо узгоджене не максимальне придушення (для кожного класу самостійно), яке відхиляє область, якщо вона має перекриття перекриття-об'єднання (IoU) з перекриттям, вибраним регіоном більшим, ніж вивчений поріг.

Дві властивості роблять виявлення ефективним. По-перше, усі параметри CNN спільно використовуються між усіма категоріями. По-друге, вектори характеристик, обчислені CNN мають низькі розміри у порівнянні з іншими типовими підходами, такими як просторові піраміди з кодуванням візуально-навантажувальних кодувань. Наприклад, функції, що використовуються в системі виявлення UVA, на два порядки більші за ваші (360k проти 4k-мірних). Результатом такого обміну є те, що час, витрачений на компіляцію пропозицій та функцій регіону (13 с / зображення на графічному процесорі або 53 с / зображення на центральному процесорі) амортизується для всіх класів.

Тільки обчислення, специфічні для класу - це точкові добутки між характеристиками та коефіцієнтами SVM та не максимальне приглушення. На

практиці всі точкові добутки для зображення об’єднуються в єдиний матричний продукт. Матриця функцій типово 2000 × 4096, а матриця ваги SVM - 4096 × N, де N - кількість класів. Цей аналіз показує, що R-CNN може масштабуватися до тисяч класів об'єктів, не вдаючись до наближених технік, таких як хешування.

Навіть якщо було 100k класів, отримане матричне множення займає лише 10 секунд на сучасному багатоядерному процесорі. Ця ефективність не просто результат використання регіональних пропозицій та спільних функцій. Система UVA, завдяки своїм високомірним характеристикам, буде на два порядки повільніше, вимагаючи 134 ГБ пам'яті лише для зберігання 100 тисяч лінійних предикторів, порівняно з лише 1,5 ГБ для наших функцій нижнього розміру

Також цікаво протиставити R-CNN нещодавня робота від Dean. щодо масштабованого виявлення з використанням хешування DPM та піктограми. Вони повідомляють, що показник MAP становить близько 16% на VOC2007 під час виконання 5 хвилин на зображення при введенні 10k класів дистракторів. З нашим підходом, 10 тисяч детекторів можуть працювати приблизно за хвилину на центральному процесорі, і оскільки ніяких наближень не робиться, ПДК залишатиметься на рівні 59%.

Дотримуючись найкращих практик PASCAL VOC, ми визнали недійсними всі рішення щодо проектування та гіперпараметри набору даних VOLVO 2007. Для остаточних результатів наборів даних VOC 2010-12 ми точно налаштували CNN на тренування VOC2012 та оптимізували наші SVM виявлення на VOC 2012. Ми подали результати тестування на сервіс оцінки лише один раз для кожного з двох основних варіантів алгоритму (без регресії).

Також мі використали набір даних COCO, та частину з набору COCO з прикладами для тренування сегментації людини, що для нашого завдання більш пріоритетне. На рисунку 19 показано результат роботи методів при використанні набору даних VOC 2010.

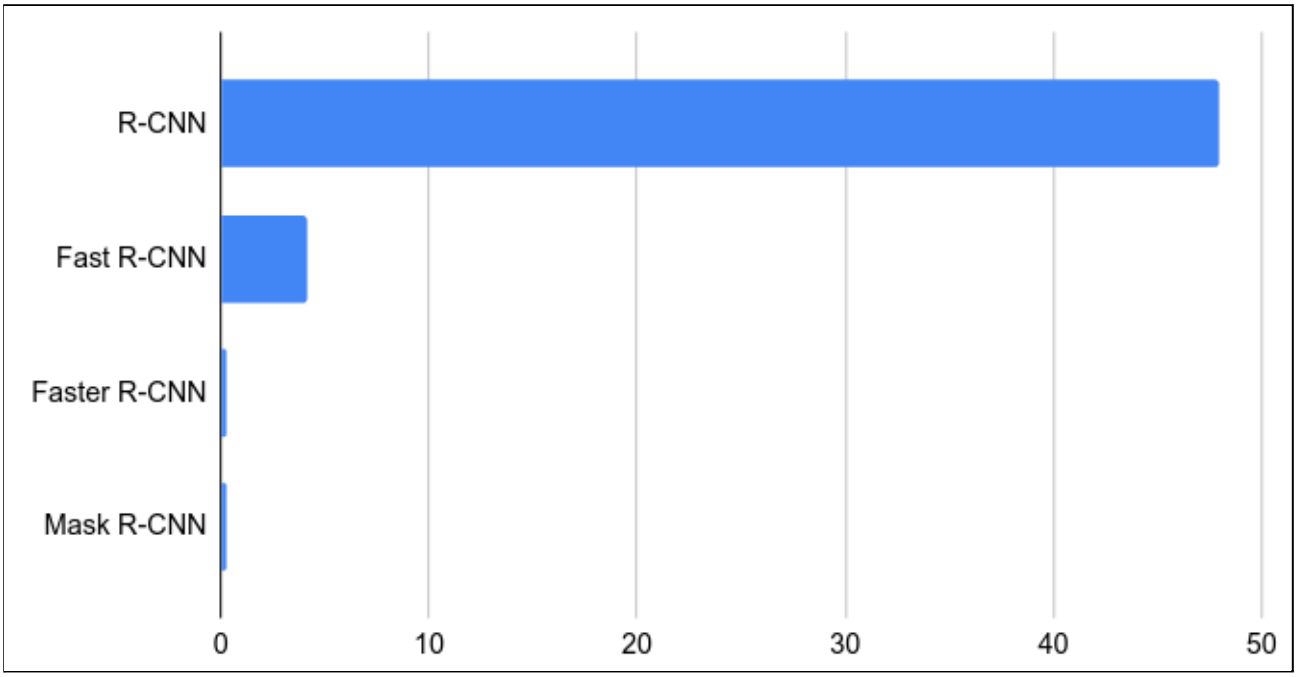

Рисунок 19 - Час роботи методів згорткової мережі на наборі даних VOC 2010

Як ми можемо побачити методи Mask R-CNN та Faster R-CNN набагато швидше впорались з завданням. В першу чергу це зумовлено тим, що нам не потрібно кожного разу подавати пропозиції 2000 регіонів до згорткової нейронної мережі. Натомість операція згортки виконується лише один раз для кожного зображення, і з нього генерується карта об’єктів.

Майже все алгоритми використовують вибірковий пошук для з’ясування пропозицій регіону. Вибірковий пошук - це повільний і трудомісткий процес, що впливає на продуктивність мережі. Тому Шаоцин Рен та співавтори придумали алгоритм виявлення об’єктів, який виключає вибірковий алгоритм пошуку і дозволяє мережі вивчати регіональні пропозиції.

Подібно до Fast R-CNN, зображення подається як вхід до згорткової мережі, яка забезпечує згорткову карту функцій. Замість використання алгоритму вибіркового пошуку на карті об’єктів для ідентифікації пропозицій регіону використовується окрема мережа для прогнозування пропозицій регіону. Потім прогнозовані пропозиції регіонів перероблюються за допомогою шару об’єднання RoI, який потім використовується для класифікації зображення в межах

запропонованої області та прогнозування значень зміщення для обмежувальних рам. На рисунку 20 ми можемо побачити результат роботи цих алгоритмів на іншому наборі даних COCO.

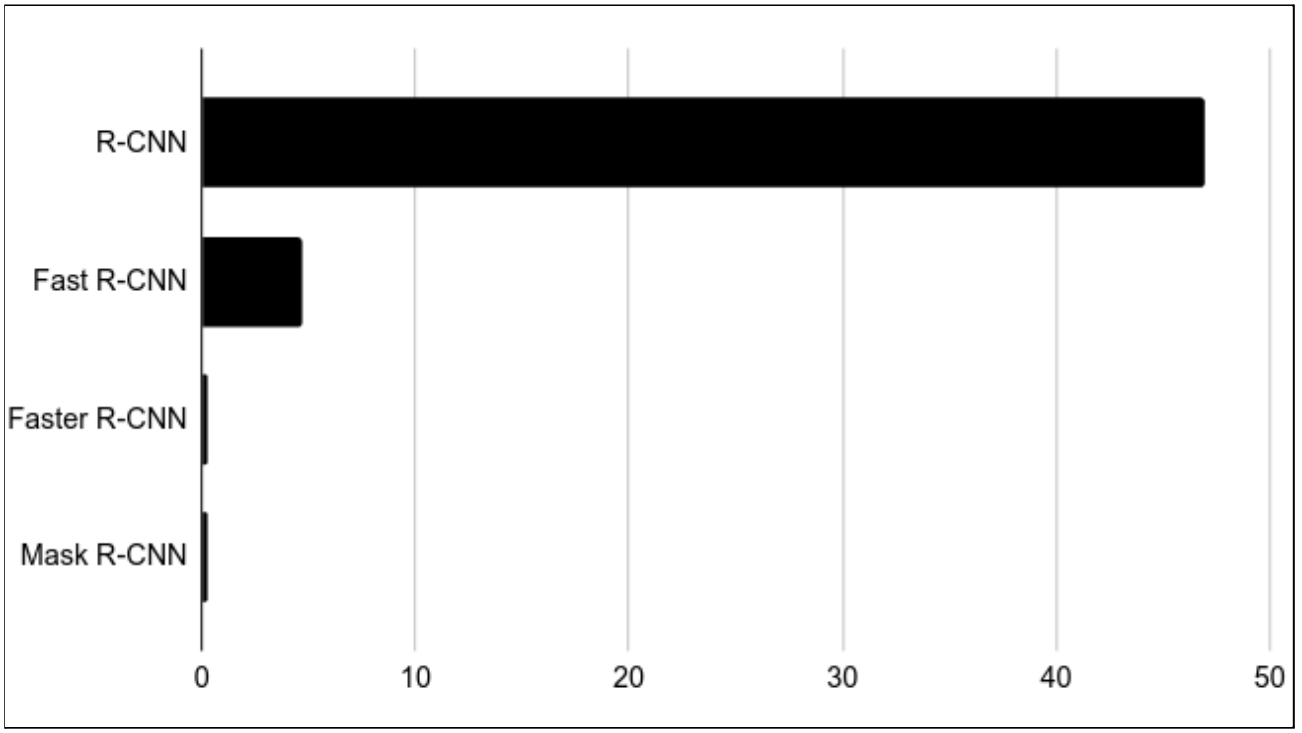

Рисунок 20 - Час роботи методів згорткової мережі на наборі даних COCO

Як і слідувало очікувати час не сильно змінився, тому що алгоритм той самий, різниця у зображеннях не повинна ніяк значно змінити швидкість обробки даних. Набір даних в першу чергу повинен впливати на якість знаходження макси об’єктів, але це ми розглянемо у наступному розділі.

5.3 Оцінка якості методів сегментації екземплярів

Перед початком експерименту потрібно зазначити що таке mAP або середня точність та як вона рахується. AP (Середня точність) - це популярна метрика для вимірювання точності детекторів об'єктів, таких як Faster R-CNN, SSD тощо. Середня точність обчислює середнє значення точності для значення відкликання

від 0 до 1. Це звучить складно, але насправді досить просто. Перед цим ми спочатку зробимо короткий підсумок точності, відкликання та IoU.

Точність вимірює, наскільки точними є ваші прогнози. тобто відсоток ваших прогнозів правильний.

Згадайте міри того, наскільки добре ви знаходите всі позитивні сторони. Наприклад, ми можемо знайти 80% можливих позитивних випадків у наших найкращих прогнозах K.

Ось їх математичні визначення:

$$Precision = \frac{TP}{Tp + FP}$$

$$Recall = \frac{TP}{Tp + FN},$$

де TP - позитивна істина, TN - негативна істина, FP - позитивна помилка, FN - негативна помилка.

IoU вимірює перекриття між двома межами. Ми використовуємо це для вимірювання того, наскільки наша передбачувана межа перекривається з основною істиною (межею реального об’єкта). У деяких наборах даних ми попередньо визначаємо поріг IoU (скажімо, 0,5), класифікуючи, чи є прогноз справжнім позитивним чи помилково позитивним.

Набір зображень було зроблено та підготовлено з використанням усіх RCNN (RCNN, FAST RCNN, FASTER RCNN та Mask RCNN). Зображення тренуються та перевіряються за допомогою caffe framework (c ++, python), а також у Matlab.

Конволюційна архітектура для швидкого вбудовування функцій, відома в народі як CAFFE, є глибоким скелетом навчання. Він має швидкість, сумісність, виражає потужність та модульність. Він написаний за допомогою мови програмування C ++ та Python і є BSD-ліцензованою структурою з відкритим кодом. CAFFE працює на таких операційних системах, як Mac OS, Windows, Linux тощо. Він допомагає значній кількості програм глибокого навчання в ідентифікації, розподілі та класифікації зображень. CAFFE може працювати з

графічними процесорами, а також з центральними процесорами, а також з низкою багатопроцесорних залежних обчислювальних бібліотек ядра, таких як Intel MKL, WDNN, NVIDIA, CNN, LSTM, RCNN та моделі нейронних мереж, що підтримуються CAFFE.

Замість того, щоб визначати їх як код, оптимізація представлена у вигляді простого тексту разом із науковим прогресом для еталонних моделей та кодів. CAFFE був розроблений Янгін Цзя під час навчання в докторантурі в Каліфорнійському університеті в Берклі. Але в даний час велика кількість людей працює над цим проектом у GitHub. Деякі характерні особливості кафе такі:

- фреймворк CAFFE має надзвичайну швидкість обробки, це найпростіший у промислових дослідженнях промислових та дослідницьких областях, він має можливість обробляти зображення з дуже високою швидкістю за допомогою одного nvidak 47, найшвидшого перетворення, зручного в цій області;
- фреймворк CAFFE підтримує стимулювання стимулів та конструктивну розробку для кодування, що забезпечує точність та узгодженість кодів з новими тенденціями у використанні. Співробітники та розробники у величезній кількості є ключовим фактором для підтримки та роботи цієї структури;
- фреймворк CAFFE отримав велику підтримку громадськості люди з усіх галузей пропонують стартапи академій промисловості дослідження та мультимедійний сектор є частиною цього;
- фреймворк CAFFE має основу для розширення, отже, легко просуває нові ідеї та реалізує їх.

Порівнюються наступні результати, як показано в таблиці 1. Найкращі показники були отримані з архітектурою Mask RCNN та з точністю ~ 80.00%.

Таблиця 1 Порівняння методів сегментації за mAP

	RCNN	Fast RCNN	Faster RCNN	Mask RCNN
Pascal VOC	67.1	77.3	78.1	80.1
COCO	68.2	78.2	78.4	81.4
COCO sub.	67.5	76.6	75.9	81.4

Як видно з результатів дослідження Mask RCNN має найвищий рівень якості серед усіх розглянутих методів. Також можна побачити що підбірка набору даних COCO виключно для виявлення людини не принесла великого покращення. а у деяких випадках результат і зовсім був гіршим, це може бути обумовлено тим, що частина набору даних COCO була ніяк доповнена іншими прикладами, у цьому випадку наша модель навчалася виключно з обмеженою вибіркою даних, що не може принести покращення якості.

Також якщо подивитись на різницю між наборами даних COCO та Pascal VOC можна зробити висновок, що набір даних COCO більш підходить для нашого типу завдання та приніс кращий результат. Усі набори даних мають гарний результат для свого типу завдання, але у нашому випадку COCO підійшов найкраще.

ВИСНОВКИ

На закінчення існує велика кількість підходів до машинного навчання для сегментування зображень. У цьому проекті Mask R-CNN був оцінений на зображеннях людини та довів, що він здатний сегментувати людину, виділяти передній фон від заднього та створювати якісний тримап для подальшого використання алгоритмом матування зображення. Мережа навчалася на обмеженому наборі даних, і результати хороші як для гарно освітлених фото, так и для зображень з низькою роздільною здатністю. Як і в будь-якому проекті машинного навчання, очікується, що навчання на більших наборах даних дасть кращі результати. Через обмежений обсяг даних було досліджено синтетичне генерування даних, яке допомогло мережі узагальнити оцінку зображень з різними стилями, ніж треновані. Очевидно також, що обмеження апаратного забезпечення, в першу чергу пам'яті графічного процесора, все ще обмежують бажані параметри навчання.

У результаті виконання кваліфікаційної роботи роботи було проведений докладний аналіз предметної області, проведено дослідження методів та фреймворків аналізу та сегментації об’єктів.

Був проведений аналіз різних методів, алгоритмів та підходів для класифікації та сегментації об’єктів. Розібрани переваги кожного з підходів та їх основні принципи. На основі аналізу інструментів було прийнято рішення яку з моделей використовувати для конкретного типу завдання.

В результаті аналізу було встановлено критерії ефективності методів семантичної сегментації та сегментації екземплярів, за якими можливо створити порівняльну характеристику алгоритмів та моделей. Було проаналізовано швидкісну складову моделей розпізнавання образів. Виявлено основні потенційні проблеми кожної з моделей та алгоритму, та їх основні переваги. Знайдені критерії за якими можно проводити тестування робочої системи за декількома характеристиками.

У ході роботи було проаналізовано такі завдання:

– семантична сегментація;

– виявлення об’єктів;

– класифікація зображень;

– сегментація екземплярів.

В результаті аналізу і розробки поставлену задачу було виконано в заданому обсязі.

Також у роботі було розглянуто декілька наборів даних, проаналізовано роботу методів сегментації з різними наборами даних. Найкращим з наборів виявився набір даних COCO. Він краще з усіх підходить саме до нашого типу завдань, а саме до сегментації людини для створення тримапу зображення.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Шелхамер Е. Д., Лонг Д. С., Даррел А. В. (2017). Fully convolutional networks for semantic segmentation. «Machine Intelligence Institute».
2. Mask r-cnn for object detection and instance segmentation on keras andtensorflow: https://github.com/matterport/Mask_RCNN (дата звернення: 28.02.2021).
3. Kirillov E., Levinkov E., Andres B., Savchynskyy B., and Rother C. (2017). Instancecut:from edges to instances with multicut. “IEEE Conference onComputer Vision and Pattern Recognition”.
4. Oleksandr Topchii, Oleksandr Samantsov, Oksana Mazurova, Mariia Shirokopetleva. A Study of Optimization Models for Creation of Artificial Intelligence for The Computer Game in The Tower Defense Genre. // Problem of Infocommunications. Science and Technology (PIC S&T′2020). 2020. Kharkiv, Ukraine.
5. Dana Angluin (1987). Queries and concept learning. “Machine Learning”
6. Deep Learning. URL: https://www.technologyreview.com/deep-learning (дата звернення: 05.03.2021)
7. Python. Документація. URL: https://docs.python.org/3.7/ (дата звернення: 01.03.2021).
8. Chalyi S., Leshchynskyi V., Leshchynska I. Method of constructing explanations for recommender systems based on the temporal dynamics of user preferences // «EUREKA: Physics and Engineering». 2020. No 3.-P. 43-50.
9. OpenCV Документація. URL: https://docs.opencv.org/master/d6/d00/tutorial_py_root.html (дата звернення: 10.03.2021).
10. Tensorflow Документація, URL: https://www.tensorflow.org/api_docs (дата звернення: 10.03.2021).
11. Postgres. Документація. URL: https://www.postgresql.org/docs/ (дата звернення: 12.03.2021).
12. Gorokhovatskyi V, Rusakova N, Tvoroshenko I. The application of image analysis methods and predicate logic in applied problems of magnetic monitoring, Volume

79, 2020 Issue 20. -Telecommunications and Radio Engineering. - P 1801-1811. DOI: 10.1615/ TelecomRadEng.v79.i20.30. 2020. 2711.-P. 518-528.

13. Splash of Color: Instance Segmentation with Mask R-CNN and TensorFlow. URL:https://engineering.matterport.com/splash-of-color-instance-segmentation-with-mask-r-cnn-and-tensorflow-7c761e238b46 (дата звернення: 12.04.2021).