

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет _____ Комп'ютерних наук _____
(повна назва)

Кафедра _____ Штучного інтелекту _____
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти _____ другий (магістерський) _____

_____ Дослідження та розробка методів аналізу рукописного тексту _____
_____ на основі гібридних нейромереж _____
(тема)

Виконав:
студент 2 курсу, групи _____ СШМ-20-3 _____
_____ Мосієнко Р.М. _____
(прізвище, ініціали)

Спеціальність 122 Комп'ютерні науки _____
(код і повна назва спеціальності)

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

Освітня програма Системи штучного інтелекту _____
(повна назва спеціалізації)

Керівник _____ проф. Філатов В.О. _____
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри _____
(підпис)

_____ В.О. Філатов _____
(прізвище, ініціали)

2022 р.

Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук
(повна назва)
Кафедра Штучного інтелекту
(повна назва)
Рівень вищої освіти другий (магістерський)
Спеціальність 122 Комп'ютерні науки
(код і повна назва)
Тип програми освітньо-наукова
(освітньо-професійна або освітньо-наукова)
Освітня програма Системи штучного інтелекту (СШІ)
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____

(підпис)

« _____ » _____ 20 ____ р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові Мосієнку Роману Миколайовичу
(прізвище, ім'я, по батькові)

1. Тема роботи Дослідження та розробка методів аналізу рукописного тексту на основі гібридних нейромереж

затверджена наказом університету від 24 березня 2022 р. № 414Ст

2. Термін подання студентом роботи до екзаменаційної комісії _____ 20 ____ р.

3. Вихідні дані до роботи Науково-технічні публікації, дані Інтернет-джерел та відомих наукових проектів

4. Перелік питань, що потрібно опрацювати в роботі _____

1) Аналіз предметної галузі

2) Вибір техніки для реалізації та її опис

3) Реалізація методу

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (п.5 включається до завдання за рішенням випускової кафедри) _____

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

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

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Отримання завдання на кваліфікаційну роботу	28.03.2022	виконано
2	Огляд та аналіз сучасного стану проблеми	29.03.2022–04.04.2022	виконано
3	Постановка задачі	05.04.2022	виконано
4	Класифікація технологій для розпізнавання рукописного тексту	06.04.2022–09.04.2022	виконано
5	Аналіз галузей застосування	10.04.2022–12.04.2022	виконано
6	Аналіз реалізованих існуючих технологій та підходів	13.04.2022–17.04.2022	виконано
7	Розробка системи	18.04.2022–23.04.2022	виконано
8	Підготовка пояснювальної записки	24.05.2022–01.05.2022	виконано
9	Надання пояснювальної записки на перевірку	02.05.2022	виконано
10	Рецензування	07.05.2022	виконано
11	Захист кваліфікаційної роботи		

Дата видачі завдання 28 березня 2022 р.

Студент _____
(підпис)

Керівник роботи _____
(підпис) _____ (посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка: 56 с., 12 рис., 2 дод., 22 джерела.

ІНФОРМАЦІЙНА СИСТЕМА, МЕТОД БІНАРІЗАЦІЇ, МЕТОД ГІЛОК І КОРДОНІВ, РОЗПІЗНАВАННЯ РУКОПИСНОГО ТЕКСТУ.

Об'єктом досліджень є розробка системи для розпізнавання рукописних символів

Предметом досліджень кваліфікаційної роботи є інформаційні технології і методи аналізу рукописного тексту.

Мета досліджень: провести огляд та дослідження предметної області, дослідити існуючі методи розпізнавання й аналізу рукописних текстів

Метою кваліфікаційної роботи є розробка системи для дослідження методів розпізнавання відсканованих зображень з рукописним текстом в автономному режимі

Методи дослідження – системний аналіз (для виділення окремих блоків для роботи з розробкою системи та дослідження алгоритмів розробки), аналогія (для дослідження роботи та визначення недоліків подібних систем прогнозування).

ABSTRACT

Explanatory note: 56 p., 12 figs., 2 ann., 22 sources.

**BINARIZATION METHOD, BRANCH AND BORDER METHOD,
HANDWRITTEN RECOGNITION, INFORMATION SYSTEM, TRANSPORT
TASK.**

The object of research is to develop a system for recognizing handwritten symbols

The subject of research qualification work is information technology and methods of analysis of handwritten text.

The purpose of research: to conduct a review and research of the subject area, to investigate existing methods of recognition and analysis of manuscripts

The purpose of the qualification work is to develop a system for the study of methods for recognizing scanned images with handwritten text in offline mode

Research methods: system analysis (to select individual blocks to work with system development and research development algorithms), analogy (to study the work and identify the shortcomings of such forecasting systems).

ЗМІСТ

Вступ.....	7
1 Аналіз предметної галузі	9
Огляд і аналіз сучасного стану проблеми	9
Випадки використання	10
Методи обробки й аналізу рукописного тексту.....	11
Постановка задачі	12
2 Вибір техніки для реалізації та її опис	13
Техніки реалізації систем для зчитування рукописного тексту.....	13
2.4 Опис обраного алгоритму	25
3 Реалізація методу	32
Вирішення задачі	32
Висновки	39
Перелік джерел посилання	40
Додаток А Текст програми.....	43
Додаток Б Відомість кваліфікаційної роботи.....	56

ВСТУП

Актуальність теми. Розпізнавання рукописних цифр та символів у сучасному світі стали дуже важливими завдяки їх практичному застосуванню в різних повсякденних сферах життєдіяльності. Це підтверджується тим, що в останні роки стрімко розробляються системи розпізнавання в різних галузях, де потрібна висока ефективність класифікації. Системи, які використовуються для розпізнавання рукописних літер, символів та цифр допомагають людям вирішувати складніші завдання, які в іншому випадку були б трудомісткими та дорогими. Гарним прикладом є використання систем автоматичної обробки, що використовуються в банках для обробки банківських чеків. Без них, банк зобов'язаний наймати окремих співробітників, які можуть бути не такими ефективними, як комп'ютеризована система обробки. Системи розпізнавання рукописного тексту можуть бути натхненні біологічними нейронними мережами, які дозволяють людям і тваринам навчатися і моделювати нелінійні і складні відносини. Це означає, що їх можна розвивати від штучної нейронної мережі. Мозок людини може розпізнавати різні об'єкти рукописного введення, наприклад цифри, літери та символи. Однак люди упереджені, тобто вони можуть інтерпретувати рукописні літери та цифри по-різному. Комп'ютеризовані системи, з іншого боку, є неупередженими і можуть виконувати дуже складні завдання, які вимагають від людей витратити багато своєї енергії та часу.

Метою дослідження є знаходження оптимального алгоритму для розпізнавання рукописного тексту.

Завдання дослідження:

- провести огляд та дослідження предметної області;
- дослідити існуючі методи розпізнавання, виявити їх переваги та недоліки;

– розробити програмний інструментарій для розпізнавання тексту на основі гібридних нейромереж.

Об’єктом досліджень є розробка системи для розпізнавання рукописних символів.

Предметом досліджень є технології та методи розпізнавання рукописних текстів на основі гібридних нейромереж.

1 АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ

Огляд і аналіз сучасного стану проблеми

Очікується, що до 2025 року розмір ринку оптичного розпізнавання символів (OCR) становитиме 13,38 мільярда доларів США, при цьому річний ріст на 13,7 %. Це зростання зумовлене швидкою оцифровкою бізнес-процесів з використанням OCR, щоб зменшити витрати на оплату праці та заощадити дорогоцінний час людини. Хоча OCR вважалось вирішеною проблемою, є один ключовий її компонент, розпізнавання рукописного тексту, який досі вважається складною постановкою проблеми. Висока різниця в стилях рукописного введення для різних людей і низька якість рукописного тексту в порівнянні з друкованим текстом створюють значні перешкоди для перетворення його в текст, зрозумілий комп'ютеру. Тим не менш, це важлива проблема для багатьох галузей, таких як охорона здоров'я, страхування та банківська справа.

Нещодавні досягнення в глибокому навчанні, такі як поява архітектур трансформаторів, прискорили прогрес у розпізнаванні рукописного тексту. Розпізнавання рукописного тексту називається інтелектуальним розпізнаванням символів (ICR) через той факт, що алгоритми, необхідні для розв'язання ICR, потребують набагато більшого інтелекту, ніж розв'язування загального OCR [3].

Проблеми в розпізнаванні рукописного тексту [3]:

- величезна мінливість і неоднозначність штрихів від людини до людини;
- стиль почерку окремої людини також змінюється час від часу і є непостійним;
- погана якість вихідного документа або зображення через погіршення з часом;

- текст у друкованих документах розміщений у прямій лінії, тоді як людям не потрібно писати рядок тексту по прямій на білому папері;
- скорописний почерк ускладнює розділення та розпізнавання символів;
- рукописний текст може мати змінний поворот праворуч, що на відміну від друкованого тексту, де весь текст розташований прямо;
- збір хорошого міченого набору даних для навчання недешевий у порівнянні з синтетичними даними.

Випадки використання Охорона

здоров'я та фармацевтика.

Оцифровка рецептів для пацієнтів є основною проблемою в галузі охорони здоров'я фармацевтичної промисловості. Наприклад, «Roche» щодня обробляє мільйони петабайт медичних PDF-файлів [11]. Ще одна сфера, де виявлення рукописного тексту має ключовий вплив, — це реєстрація пацієнтів та оцифрування форми. Додавши розпізнавання рукописного тексту до свого набору послуг, лікарні можуть значно покращити роботу користувачів.

Страхування.

Велика страхова галузь отримує понад 20 мільйонів документів на день, і затримка в обробці позовів може страшенно вплинути на компанію. Документ про позови може містити різні стилі рукописного введення, і чиста ручна автоматизація їх обробки повністю сповільнить конвеєр [11].

Банківська справа.

Люди виписують чеки на регулярній основі, і вони все ще відіграють важливу роль у більшості безготівкових операцій. У багатьох країнах, що розвиваються, нинішня процедура обробки чеків вимагає від працівника банку прочитати та вручну ввести інформацію, надану на чеку, а також перевірити такі записи, як підпис і дата [20]. Оскільки банк щодня повинен

обробляти велику кількість чеків, система розпізнавання рукописного тексту може заощадити витрати та години людської роботи.

Інтернет-бібліотеки.

Величезний обсяг історичних знань оцифровується шляхом завантаження сканованих зображень для доступу до всього світу. Але ці зусилля не дуже корисні, доки не можна ідентифікувати текст на зображеннях, який можна проіндексувати, запитувати та переглядати. Розпізнавання рукописного тексту відіграє ключову роль у оживленні середньовічних і 20-го століття документів, листівок, досліджень тощо [15].

Методи обробки й аналізу рукописного тексту

Методи розпізнавання рукописного введення можна в цілому розділити на два нижченаведені типи на рисунку 1.1.

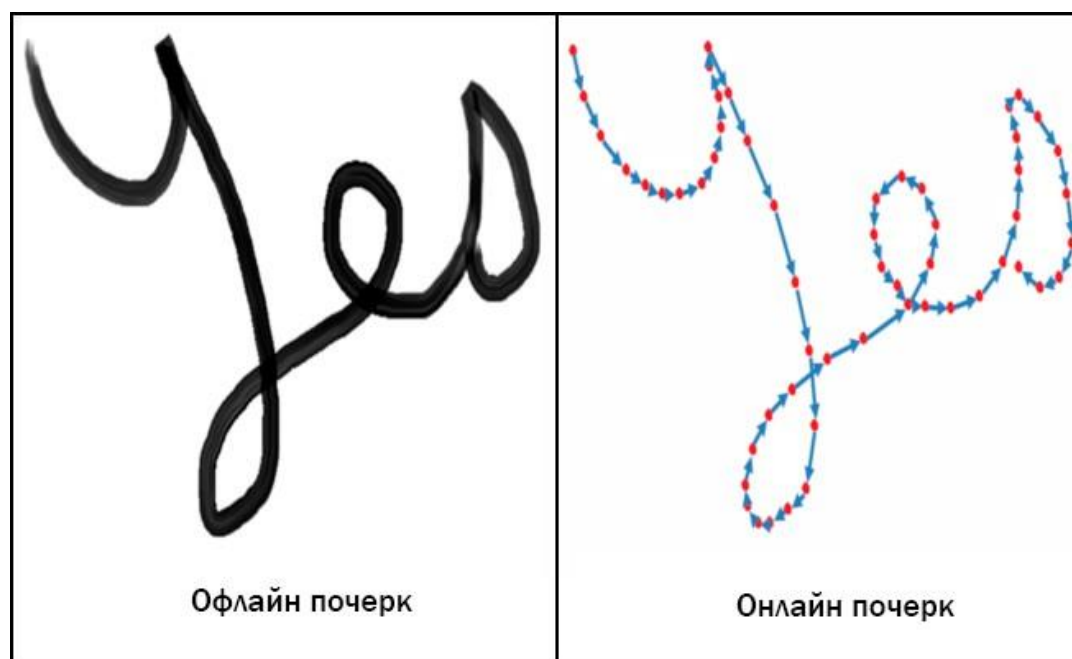


Рисунок 1.1 – Методи розпізнавання рукописного введення

Онлайн-методи: онлайн-методи включають цифрове перо або стилус і мають доступ до інформації про штрих, розташування пера під час написання тексту, як показано на правому малюнку вище (рисунок 1.1). Оскільки вони, як правило, мають багато інформації щодо потоку написаного тексту, їх можна класифікувати з досить високою точністю, і розмежування між різними символами в тексті стає набагато чіткішим.

Автономні методи: автономні методи передбачають розпізнавання тексту після його запису, а отже, вони не матимуть інформації про штрихи чи напрямки, задіяні під час написання, з можливим додаванням певного фонового шуму від джерела, тобто паперу.

У реальному світі не завжди можливо або доречно носити цифрову ручку з датчиками для захоплення інформації про штрих, тому завдання розпізнавання тексту в автономному режимі є набагато більш актуальною проблемою [16].

Постановка задачі

Постановка задачі.

Необхідно дослідити методи для аналізу рукописного тексту на основі гібридних нейромереж та розробити систему для розпізнавання рукописних символів.

Основною метою даної роботи є виділення ознак, отриманих методом бінаризації на основі гібридних нейромереж, для розпізнавання рукописних символів англійської мови. Розпізнавання рукописних зображень символів повинно бути здійснено за допомогою багат шарової штучної нейронної мережі як класифікатора. Деякі методи попередньої обробки, такі як витончення, видалення переднього та фонового шумів, кадрування та нормалізація розміру тощо, також використовуються для попередньої обробки зображень символів перед їх класифікацією.

Вхідними даними є – відскановане рукописне зображення символу.

2 ВИБІР ТЕХНІКИ ДЛЯ РЕАЛІЗАЦІЇ ТА ЇЇ ОПИС

2.1 Техніки реалізації систем для зчитування рукописного тексту

Початкові підходи до вирішення проблеми розпізнавання рукописного тексту включали методи машинного навчання, такі як приховані моделі Маркова (HMM), SVM тощо [17]. Після попередньої обробки початкового тексту виконується вилучення ознак для визначення ключової інформації індивідуального характеру, такої як петлі, точки перегину, співвідношення сторін тощо. Ці згенеровані функції тепер передаються в класифікатор, наприклад HMM, щоб отримати результати. Продуктивність моделей машинного навчання досить обмежена через фазу вилучення функцій вручну та обмежену здатність до навчання. Крок вилучення функцій відрізняється для кожної окремої мови і, отже, не масштабується. З появою глибинного навчання відбулося величезне покращення точності розпізнавання рукописного тексту. Найвидатніші техніки у сфері глибокого навчання для розпізнавання рукописного тексту.

Багатовимірні рекурентні нейронні мережі.

RNN/LSTM, як відомо, може працювати з послідовними даними, щоб ідентифікувати тимчасові закономірності та генерувати результати. Але вони обмежуються обробкою одновимірних даних і, отже, не будуть безпосередньо застосовні до даних зображень [18]. Щоб вирішити цю проблему, автори цієї роботи запропонували багатовимірну структуру RNN/LSTM, як показано на рисунку 2.1.

Товсті лінії показують зв'язки з поточною точкою (i, j). Зв'язки всередині прихованого шару повторюються. Пунктирні лінії показують смуги сканування, уздовж яких було переглянуто попередню точку, починаючи з верхнього лівого кута.

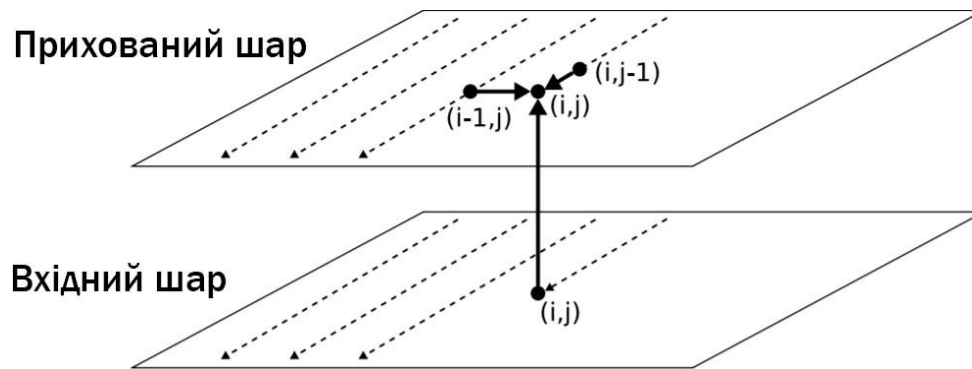


Рисунок 2.1 – Двовимірний MDRNN.

Нижче наведена різниця між загальною RNN та багатовимірною RNN. У загальному RNN, прихований рівень каже, що i отримує стан від попереднього прихованого рівня в час $i-1$. У багатовимірній RNN, наприклад, двовимірна RNN, прихований рівень (i, j) отримує стани від кількох попередніх прихованих шарів, тобто $(i-1, j)$ і $(i, j-1)$, і таким чином захоплює контекст з обох висоти і ширини зображення, що має ключове значення для чіткого розуміння локального регіону мережею. Він додатково розширений, щоб отримати інформацію не лише з попередніх, але й із майбутніх шарів, подібно до того, як BI-LSTM отримує інформацію від $t-1$ і $t+1$. Аналогічно, прихований шар 2D MDRNN і тепер може отримувати інформацію $(i-1, j)$, $(i, j-1)$, $(i+1, j)$, $(i, j+1)$, таким чином фіксує контекст у всіх напрямках.

Спочатку вхідне зображення збирається в блоки шириною 3 пікселі та висотою 4 пікселі, які потім скануються чотирма шарами MDLSTM (рисунок 2.2). Активації комірок у кожному шарі відображаються окремо, а стрілки в кутах вказують напрямок сканування. Далі активації MDLSTM збираються в коробки 4×3 і подаються на прямий шар блоків підсумовування (\tanh). Цей процес повторюється ще два рази, поки остаточні активації MDLSTM не будуть згорнуті до одновимірної послідовності та транскрибуються шаром CTC. У цьому випадку всі

символи позначено правильно, крім останнього другого, і правильна назва міста вибирається зі словника [18].

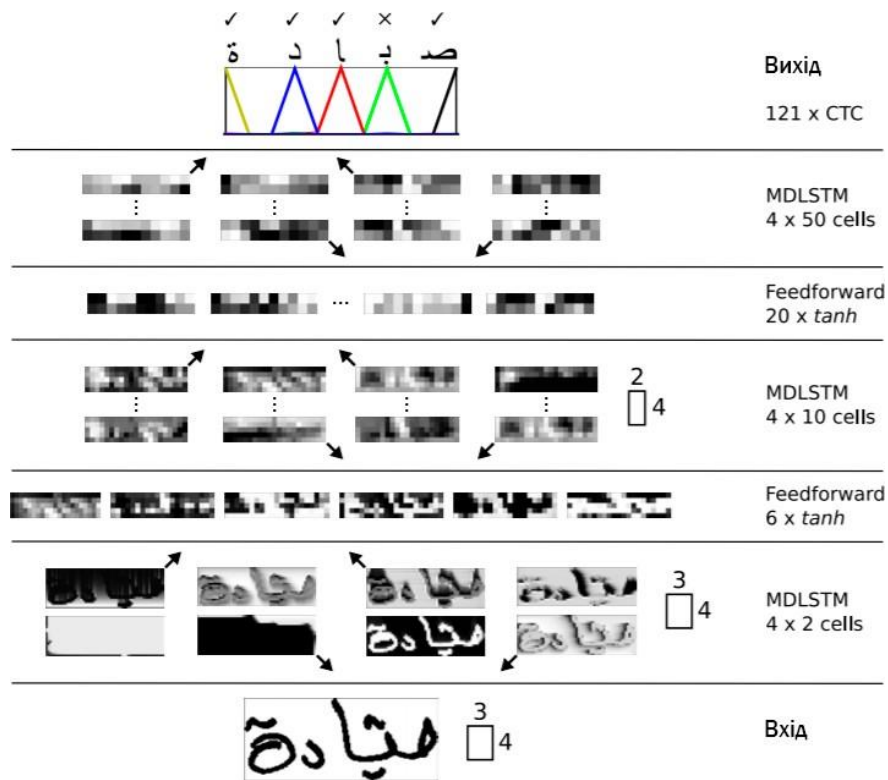


Рисунок 2.2 – Повна система розпізнавання

Connectionist Temporal Classification (CTC) – це алгоритм, який використовується для вирішення таких завдань, як розпізнавання мовлення, розпізнавання рукописного тексту тощо, де доступні лише вхідні дані та вихідна транскрипція, але не надаються деталі вирівнювання, наприклад, як певний частина у аудіо для мовлення або конкретна область на зображеннях для рукописного введення вирівнюється за певним символом. Прості евристики, такі як надання кожному символу однакової області, не працюватимуть, оскільки кількість місць, які займає кожен символ, залежить від почерку від людини до людини та час від часу.

Для нашого випадку використання розпізнавання рукописного тексту розглянемо області вхідного зображення для конкретного речення

як вхідні дані $X=[x_1, x_2, \dots, x^*T]$, а очікуваний вихід — $Y=[y_1, y_2, \dots, y^*U]$. Враховуючи X , ми повинні знайти точний Y . Алгоритм CTC працює, беручи вхідні дані X і даючи розподіл по всіх можливих Y , використовуючи які ми можемо зробити прогноз кінцевого результату.

CTC використовує основний символ щоб розрізнити повторювані символи та повторювані символи в області введення. Наприклад, певний символ може охоплювати кілька областей введення, і, таким чином, CTC виводить один і той же символ послідовно. Приклад: вхід james і вихід CTC jjaammmees. Остаточний результат виводиться шляхом згортання повторюваних виводів, і, отже, ми отримуємо Джеймса. Але тепер, щоб представити повторювані символи, які вимовляють «l» у привітанні, нам потрібно встановити розділення, і, таким чином, усі вихідні дані розділяються дефісом (-). Тепер результатом для hello може бути h-ee-l-l-ll-oo, який у разі згортання стане hello, а не helo. Більше інформації про те, як працює CTC, можна побачити тут CTC [4].

Декодуючи вихід CTC на основі простої евристики найвищої ймовірності для кожної позиції, можна отримати результати, які можуть не мати жодного сенсу в реальному світі. Щоб вирішити це, можна використовувати інший декодер, щоб покращити результати. Типи декодування:

- декодування найкращого шляху: це загальне декодування, яке було описано вище. У кожній позиції беремо вихід моделі і знаходимо результат з найбільшою ймовірністю;

- декодування пошуку променів: замість того, щоб брати один вихід із мережі щоразу, пошук променів пропонує зберегти декілька вихідних шляхів із найвищими ймовірностями та розширити ланцюжок за допомогою нових вихідних даних та шляхів відкидання з меншою ймовірністю, щоб зберегти розмір променя. Результати, отримані за допомогою цього підходу, більш точні, ніж за допомогою жадібного підходу;

– променевий пошук за допомогою мовної моделі: пошук променем дає точніші результати, ніж пошук у сітці, але все одно це не вирішить проблему отримання значущих результатів. Щоб вирішити це, ми можемо використовувати мовну модель разом із пошуком променя, використовуючи як ймовірності моделі, так і мовну модель для отримання кінцевих результатів [9].

Кодер-декодер і мережі уваги.

Моделі Seq2Seq, що мають мережі кодер-декодер, нещодавно стали популярними для вирішення завдань розпізнавання мовлення, машинного перекладу тощо, і, таким чином, були розширені для вирішення випадку використання розпізнавання рукописного тексту шляхом розгортання додаткового механізму уваги. Давайте обговоримо деякі фундаментальні дослідження в цій області.

Сканування та зчитування.

У цій основоположній роботі Scan, Attend and Read (SAR) автори пропонують використовувати модель на основі уваги для наскрізного розпізнавання рукописного тексту. Основним внеском дослідження є автоматична транскрипція тексту без сегментації на рядки як етап попередньої обробки, що дозволяє сканувати всю сторінку та давати результати. Для розпізнавання рукописного тексту, чергування шарів LSTM у чотирьох напрямках та підвибірки згорток (рисунок 2.3) [13].

SAR використовує архітектуру на основі MDLSTM, подібну до тієї, яку ми обговорювали вище, з однією невеликою зміною на останньому рівні. Після останнього лінійного шару, тобто останнього блоку Sum на малюнку вище, карти об'єктів згортаються у вертикальному вимірі, і для отримання результатів застосовується остаточно функція softmax (рисунок 2.4).

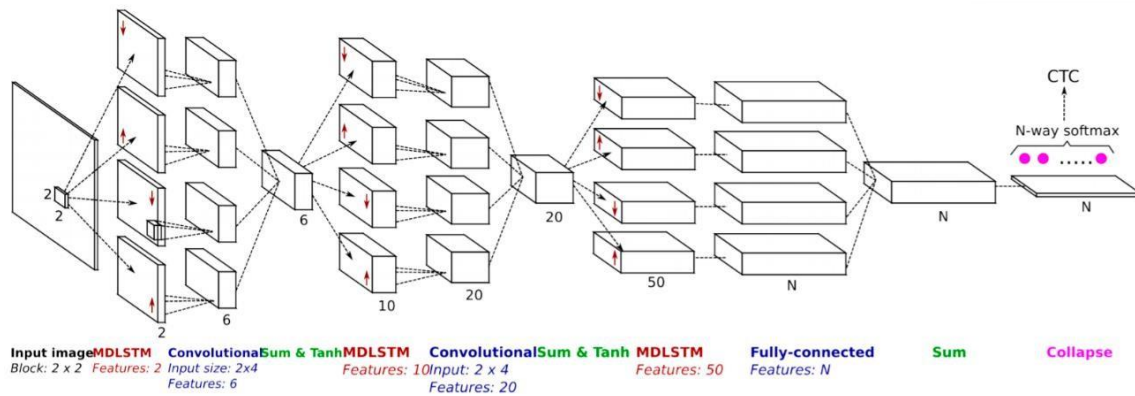


Рисунок 2.3 – MDLSTM-RNN

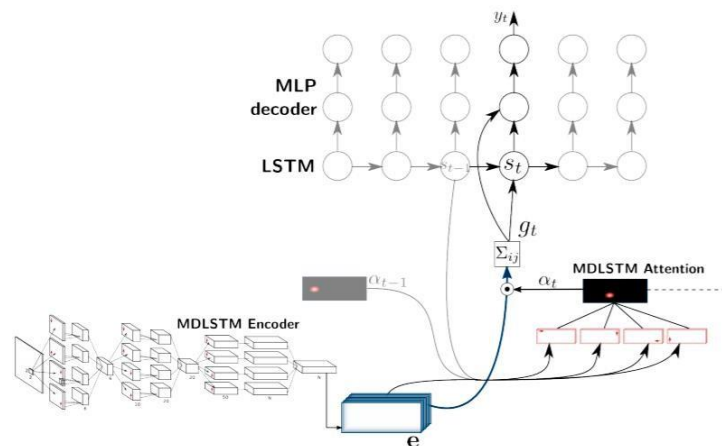


Рисунок 2.4 – Архітектура пропозицій

Згорання, стеження та пропис.

У цій роботі пропонується модель від послідовності до послідовності для розпізнавання рукописних слів. Запропонована архітектура складається з трьох основних частин: кодера, що складається з CNN і двонаправленого GRU, механізму уваги, присвяченого зосередженню на відповідних функціях, і декодера, утвореного односпрямованим GRU, здатним писати відповідне слово, персонаж за символом (рисунок 2.5) [18].

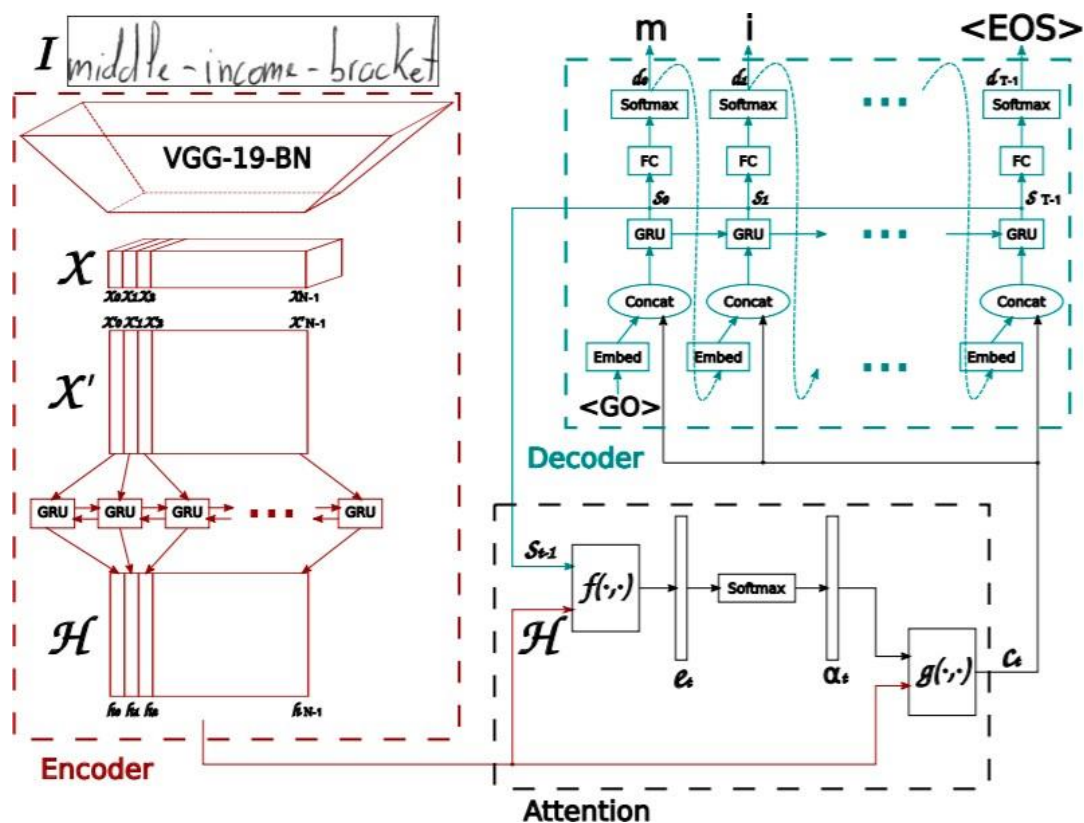


Рисунок 2.5 – архітектура моделі seq2seq з механізмом уваги

Кодер використовує CNN для вилучення візуальних функцій. Як екстрактор функцій використовується попередньо навчена архітектура VGG-19-BN. Вхідне зображення перетворюється на карту об'єктів X , яка потім змінюється на X' шляхом розбиття всіх каналів по стовпцях та їх об'єднання для отримання послідовної інформації. X' далі перетворюється в H за допомогою двонаправленого GRU [21]. GRU – це нейронна мережа, подібна до LSTM за своєю природою, і може захоплювати тимчасову інформацію.

Далі використовується модель уваги під час прогнозування вихідних даних з декодера. У статті розглядаються два різних типи досліджуваних механізмів уваги:

– увага на основі вмісту: ідея цього полягає в тому, щоб знайти подібність між поточним прихованим станом декодера та картою ознак із

кодера. Ми можемо знайти найбільш корельовані вектори ознак у карті ознак кодера, які можна використовувати для прогнозування поточного символу на поточному кроці часу. Детальніше про те, як працює механізм уваги, можна побачити тут «Увага»;

– увага на основі розташування: основним недоліком механізмів розташування на основі вмісту є те, що існує неявне припущення, що інформація про місцезнаходження вбудована у вихідні дані кодера. Інакше неможливо розрізнити вихідні символи, які повторюються з декодера. Наприклад, розглянемо слово «Charmander», символ а в ньому повторюється двічі, і без інформації про місцезнаходження декодер не зможе передбачити їх як окремі символи. Щоб пом'якшити це, поточний символ та його вирівнювання прогнозуються за допомогою вихідних даних кодера та попереднього вирівнювання [21].

Декодер являє собою односпрямований багатосаровий ГРУ. На кожному кроці часу t він отримує вхідні дані від попереднього кроку часу та вектор контексту від модуля уваги. Мультиномінальне декодування та згладжування міток досліджуються під час навчання для покращення можливості узагальнення.

Моделі трансформаторів.

Хоча мережі кодувальників-декодерів були досить хороші в досягненні результатів для розпізнавання рукописного тексту, вони мають вузьке місце в навчанні через задіяні шари LSTM, і, отже, не можуть бути розпаралеленими [13]. Останнім часом трансформери були досить успішними і замінили LSTM у вирішенні різноманітних мовних завдань. Давайте тепер обговоримо, як моделі на основі трансформатора можна застосувати для розпізнавання рукописного тексту.

Оскільки знання мови вбудовані в саму модель, немає потреби в будь-якому додатковому етапі постобробки з використанням мовної моделі, а отже, має можливість передбачати результати, які не є частиною словника. Для досягнення цього кодування тексту відбувається на рівні

символів, а не на рівні слова. Оскільки архітектура трансформатора дозволяє паралельно навчати модель для кожного регіону або персонажа, процес навчання значно спрощується.

Архітектура мережі складається з наступних компонентів [14]:

- візуальний кодер: для виділення відповідних функцій і застосування багатоголового візуального самоуваги до різних місць розташування персонажів;

- переписувач тексту: він виконує завдання введення тексту, кодування його, застосування самоуваги багатоголової мови та застосування взаємної уваги як до візуальних, так і до текстових функцій.

Візуальний кодер. Основа Resnet50 використовується для додаткових функцій, як показано на малюнку вище. Тривимірні карти об'єктів, виведені з Resnet50 F_c , передаються в модуль тимчасового кодування, який змінює форму на 2d, зберігаючи ту саму ширину i , отже, форму $(f \times h, w)$. Це подається в повністю підключений шар, щоб зменшити форму до (f, w) , і результуючий вихід є $F_{c'}$. Крім того, до $F_{c'}$ додається позиційне кодування TE, щоб зберегти інформацію про положення, як зазначено у статті «Трансформатор» Васвані. Більше інформації про те, як проектується архітектура трансформатора, можна побачити тут. Вихідні дані проходять через повністю підключений шар, щоб отримати остаточну карту об'єктів із формою (f, w) . Остаточний результат передається через багатоголовий модуль уваги з 8 головами, щоб отримати візуально насичену карту функцій.

Переписувач тексту. Введений текст передається через кодер, який генерує вбудовування на рівні символів. Ці вбудовування поєднуються з тимчасовим розташуванням, подібно до того, як у Visual Encoder за допомогою модуля Temporal Encoder. Цей результат потім передається до модуля самоуваги з кількома заголовками, який подібний до модуля уваги у візуальному кодері. Текстові об'єкти, створені разом із візуальними ознаками з візуального кодера, передаються до модуля взаємної уваги,

завданням якого є вирівнювання та комбінування вивчених функцій як із зображень, так і з тексту. Вихідні дані передаються через функцію softmax для отримання кінцевого результату [5].

При оцінці даних тесту транскрипції недоступні. Таким чином, лише початковий маркер < S > передається як вхід, а перший передбачений символ повертається в систему, яка виводить другий передбачений символ. Цей процес висновку повторюється в циклі до тих пір, поки не буде створено символ кінця послідовності < E > або коли буде досягнуто максимальної вихідної довжини N [5].

Генерація рукописного тексту.

Генерація рукописного тексту – це завдання створення справжнього рукописного тексту, яке, отже, може використовуватися для доповнення наявних наборів даних. Як ми знаємо, глибоке навчання вимагає багато даних для навчання, а отримання величезного корпусу позначених рукописних зображень для різних мов є громіздким завданням. Щоб вирішити це, ми можемо використовувати генеративні змагальні мережі для генерування навчальних даних. Давайте обговоримо тут одну з таких архітектур (рисунок 2.6).

ScrabbleGAN.

ScrabbleGAN дотримується підходу з напівнаглядом до синтезу рукописних текстових зображень, які є різноманітними як за стилем, так і за лексикою. Він має можливість створювати зображення різної довжини. Генератор також може маніпулювати отриманим стилем тексту, що дозволяє нам вирішувати, чи має текст бути скорописом, або сказати, наскільки товстим/тонким має бути штрих пера [16].

Архітектура складається з повністю згорткового генератора на основі BigGAN. Для кожного символу у вхідних даних вибирається відповідний фільтр, і всі значення об'єднуються разом, які потім помножуються на вектор шуму z , який керує згенерованим стилем тексту. Як видно вище, області, створені для кожного окремого символу,

перекриваються, що допомагає генерувати зв'язаний рекурсивний текст, а також забезпечує гнучкість різних розмірів символів [19]. Наприклад, *m* займає значну частину місця, тоді як *e* і *t* займають обмежену площу. Щоб зберегти однаковий стиль для всього слова або речення, вектор стилю z залишається незмінним для всіх символів.

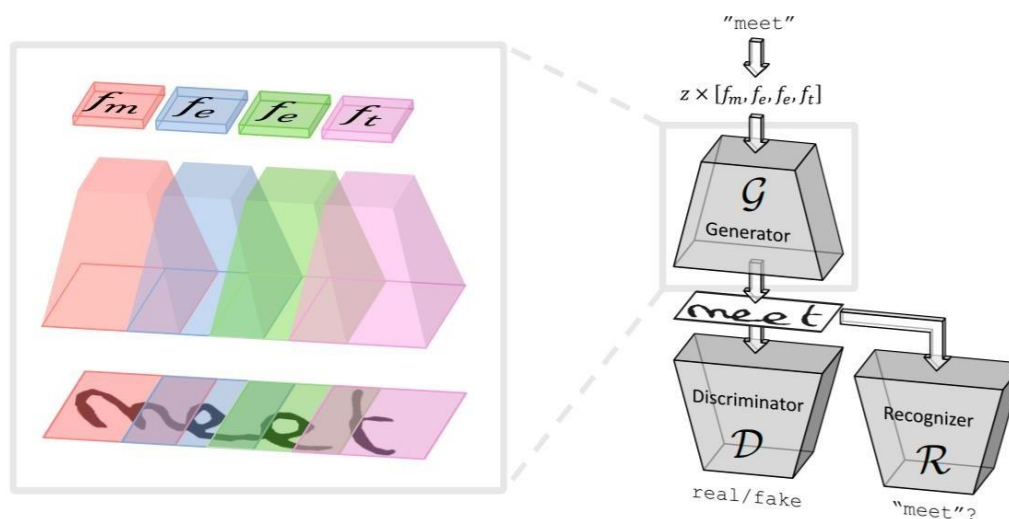


Рисунок 2.6 – Огляд архітектури для ситуація обробки слова «meet»

Згортковий дискримінатор, заснований на архітектурі BigGAN, використовується для класифікації того, чи виглядає стиль створення зображень фальшивим чи справжнім. Дискримінатор не покладається на анотації рівня символів *i*, отже, не базується на умовному GAN класу. Перевага цього полягає в тому, що немає потреби в позначених даних, *i*, отже, дані з невидимого корпусу, який не є частиною навчальних даних, можна використовувати для навчання дискримінатора. Разом із дискримінатором розпізнавач тексту *R* навчається класифікувати, чи має створений текст сенс у реальному світі, чи він є брехнею. Розпізнавач заснований на архітектурі CRNN з видаленою повторюваною головою, щоб зробити розпізнавальник трохи слабкішим і не розпізнавати текст, навіть якщо він нечіткий. Текст, згенерований на виході *R*, порівнюється з

вхідним текстом, наданим генератору, і відповідний штраф додається до функції втрати [19].

Вихідні дані, згенеровані ScrabbleGAN, показані на рисунку 2.7.



Рисунок 2.7 – Згенеровані вхідні дані ScrabbleGAN

Набори даних.

IAM: набір даних IAM містить близько 100 тисяч зображень слів з англійської мови зі словами, написаними 657 різними авторами. Набір поїздів, тестування та перевірки містить слова, написані взаємовиключними авторами [13].

CVL: набір даних CVL складається з семи рукописних документів, написаних приблизно 310 учасниками, в результаті чого отримано близько 83 тисяч слів, розділених на набори поїздів і тестів.

RIMES: містить слова з французької мови з близько 60 тисяч зображень і написані 1300 авторами, що відповідають приблизно 5 листам, написаним кожною особою [13].

Показники.

Рівень помилок символів обчислюється як відстань Левенштейна (2.1):

$$CER = \frac{S_c + I_c + D_c}{N_c} \quad (2.1)$$

де S_c – сумою замін символів;

I_c – вставок;

D_c – видалення, які необхідні для перетворення одного рядка в інший, поділена на загальну кількість символів.

N_c – символи в базовій формі.

Коефіцієнт помилок слів (2.2):

$$WER = \frac{S_w + I_w + D_w}{N_w} \quad (2.2)$$

де S_w – сума замін слів;

I_w – вставок;

D_w – видалення, які необхідні для перетворення одного рядка в інший, поділена на загальну кількість слів;

N_w – загальна кількість слів.

2.4 Опис обраного алгоритму

Для аналізу рукописного тексту був обраний метод аналізу рукописного тексту на основі гібридних неймереж, в основі якого техніка бінаризації. Вибір класифікатора візерунків та методика, яка використовується для виділення ознак, є основними факторами, які визначають розпізнавання точність і здатність системи оптичного розпізнавання символів (OCR). Основним напрямком цієї роботи є вилучення особливості, отримані методом бінаризації для розпізнавання рукописних символів англійської мови. Визнання рукописних зображень

символів було зроблено за допомогою багат шарової штучної нейронної мережі прямого зв'язку як класифікатор. Деякі методи попередньої обробки, такі як витончення, видалення переднього та фонового шумів, обрізання та розмір нормалізація тощо також використовуються для попередньої обробки зображень персонажів перед їх класифікацією. Дуже багатообіцяючі результати досягаються, коли функції бінаризації та багат шаровий класифікатор нейронної мережі прямого зв'язку використовується для розпізнавання офлайн-скорописні рукописні символи. Не можна нехтувати значенням аркуша паперу для покращення пам'яті людей. Його використовують як для приватної (листи, записки, адреси, нагадування, списки, щоденники тощо), так і для службової кореспонденції (банківський чеки, податкові бланки, бланки прийому тощо). Папір важливий у нашому повсякденному житті, тому що він дешевий, надійний, легко доступний, гнучкий у заповненні, безпечний для майбутніх довідок і його легко зберігати. Величезна кількість важливі історичні дані також записуються на паперах. Отже, існує великий попит на оцифрування всіх цих паперів документи, щоб люди в усьому світі мали доступ до цих важливих джерел знань. Для цього призначення, зображення рукописного тексту попередньо обробляється і сегментується на окремі символи і є розпізнається класифікатором нейронної мережі. Процес читання рукописного тексту зі статичних поверхонь називається офлайн-скорописним почерком. Імітація поведінки людського мозку в машині (для завдання читання рукописного тексту або друкований текст) відкрили інноваційні перспективи вдосконалення людино-машинного інтерфейсу. За останні чотири десятиліття в класифікації скорописних і необмежених рукописних символів була основною проблемою в цій галузі дослідження [4].

Бінаризація зображення є одним із найважливіших етапів попередньої обробки, що призводить до значного зменшення в обсязі

інформації, що піддається подальшому аналізу, і дозволяє збільшити її швидкість.

Така операція зазвичай використовується в багатьох системах, які використовують переважно розпізнавання форми методи і не вимагають аналізу кольору або текстури. Деякі гарні приклади можуть бути роботизованими розширені додатки, включно з стежками ліній і візуальною навігацією в коридорах і лабіринтах системи допомоги водієві (ADAS) і автономні транспортні засоби з відстеженням смуги руху, а також широко використовуються методи оптичного розпізнавання символів (OCR). Аналіз двійкового зображення також може бути успішно застосований у вбудованих системах з обмеженим обсягом пам'яті та низькою обчислювальною потужністю.

Тим не менш, відповідні результати аналізу двійкових зображень, зокрема розпізнавання тексту, залежать від правильної попередньої бінаризації. У деяких додатках, де рівномірне освітлення сцена може бути забезпечена, наприклад, популярні планшетні сканери або якась неруйнівна автоматизована книга сканери, навіть з додатковими інфрачервоними камерами, що дозволяють програмно вирівнювати відскановану книгу сторінки, може бути достатньо найпростішого глобального порогового значення. Однак у багатьох інших ситуаціях освітлення може бути нерівномірним, особливо на природних зображеннях, знятих камерами, і тому слід застосовувати більш складні адаптивні методи.

Одна з найскладніших проблем пов'язана з впливом порогів зображення на подальше аналіз – це бінаризація зображення документа, тому, як правило, нові алгоритми підтверджено використанням спеціально підготовлених зображень документів, що містять різні спотворення. Для цього тому, як правило, використовуються добре відомі набори даних конкурсів бінаризації зображень документів (DIBCO). Ці бази даних готуються до щорічних конкурсів бінаризації зображень документів, що

проводяться протягом двох провідних конференції в цій галузі – Міжнародна конференція з аналізу та визнання документів (ICDAR) та Міжнародна конференція з кордонів у розпізнаванні рукописного тексту (ICFHR), де використовуються набори даних N-DIVCO, які містять лише рукописні зображення документів без машини друковані зразки. Усі набори даних DIVCO містять не тільки спотворені зображення документів, але й «землю правда» бінарні зображення, і тому результати бінаризації можна порівняти з ними на пікселі рівень аналізу кількості правильно та неправильно класифікованих пікселів.

Незважаючи на те, що бінаризація зображень не нова тема, деякі вдосконалення алгоритмів є все ще пропонується, зокрема для бінаризації зображення історичних документів, а також нерівномірно освітленого природні образи. Пропозиція такого вдосконалення на основі ентропійного фільтра зображення, можлива застосовують у багатьох загальновідомих методах бінаризації, представлений у цій роботі.

Такий підхід веде до максимізації міжкласової дисперсії і, отже, до добра досягається розділення двох класів пікселів, представлених нарешті як чорний і білий. Внаслідок операцій на гістограмах цей метод є швидким, хоча правильно працює лише для рівномірно освітлені зображення з бімодальними гістограмами.

Оскільки зображення, що містять тіні, які є результатом нерівномірного освітлення, повинні не можна двійковувати, використовуючи єдиний глобальний поріг, деякі адаптивні алгоритми, які потребують аналізу були запропоновані також околиці кожного пікселя. Найпопулярніший підхід, розроблений Уейн Ніблак припустив визначення локального порогу як середню локальну інтенсивність зменшено на локальне стандартне відхилення, масштабоване за постійним параметром k . Подальша модифікація цей підхід із використанням додаткової нормалізації локального стандартного відхилення шляхом його поділу на його максимальне значення на зображенні відоме як метод

Саувола. Далі була його багатомасштабна версія розроблена Лаззарою і Жеро.

Простий вибір локального порогу як середньої мінімальної та максимальної інтенсивності в межах локального вікна (так зване значення midgray) було запропоновано Джоном Бернсеном, тоді як Бредлі і Рот розробили метод із використанням інтегрального зображення для розрахунку локального середнього інтенсивність сусідства. Реалізація цього методу також в модифікованих варіантах використовуючи локальну медіану та середнє зважене по Гауссу, доступний як функція адаптації порога MATLAB [8].

Розробка будь-яких нових алгоритмів обробки зображень зазвичай вимагає їх надійної перевірки на основі порівняння отриманих результатів з іншими методами. Stathis та ін. запропоновано метод оцінки алгоритмів бінаризації, заснований на порівнянні окремих пікселів, використовуючи коефіцієнт помилки пікселів (PERR), пікове відношення сигнал/шум (PSNR) і подібні показники, тоді як деякі інші підходи були представлені в оглядовій роботі Сезгіна та Санкура. Набагато більше популярним підходом є використання типових показників класифікації, заснованих на точності, запам'ятовуванні, чутливості, специфічність або F-Measure, а також застосування метрики штрафу за неправильну класифікацію (MPM) або дистанційне взаємне спотворення (DRD). Було представлено ще один метод оцінки бінаризації Лінс та ін., яка використовує набір даних синтетичних зображень для порівняння різних порогових значень алгоритми. Тим не менш, враховуючи кінцеві результати розпізнавання зображення документа як розпізнаних текстових рядків, більш корисним підходом було б застосування метрик, розрахованих для символи замість окремих пікселів. Окрім F-Measure, деякі показники, призначені для текстових рядків, наприклад відстань Левенштейна, що визначається як кількість символічних операцій, необхідних для перетворення однієї рядок в інший, також можна застосувати.

Аналізуючи нерівномірно освітлені зображення документів, можна отримати важливу інформацію з використанням локальної ентропії зображення, яку можна обчислити за допомогою MATLAB `entropyfilt` функція. Використовуючи параметри за замовчуванням, локальну міру випадковості рівнів сірої околиці, визначені маскою 9×9 пікселів, були досягнуті та збережені як результат для центральний піксель. Такий підхід може бути корисним для виявлення подробиць зображень, цілей перемикачів в адаптивній медіанній фільтрації, а також для попередньої обробки зображення з наступним порівнянням властивостей регіонів зображення. Отже, локальний ентропійний фільтр розглядався у запропонованому методі як один із етапи попередньої обробки для адаптивного зображення бінаризація нерівномірно освітлених зображень документів піддається подальшому оптичному розпізнаванню тексту.

Варто зазначити, що більшість механізмів OCR використовували деякі «вбудовані» процедури порогового значення і тому їхні результати залежать також від якості вхідних даних. Наприклад, широко використовував безкоштовне програмне забезпечення Tesseract OCR, розроблене Google, використовувало глобальне порогове значення Otsu, тоді як комерційне програмне забезпечення ABBYY FineReader використовувало адаптивний метод Бредлі. Тому застосування деяких інших методів бінаризації зображень може покращити або зменшити розпізнавання точність, оскільки «внутрішнє» порогове значення OCR не змінює вхідне двійкове зображення. отже, попереднє порогове значення зображення може розглядатися як заміна методів за замовчуванням, що використовуються в механізмі OCR [9].

Спрощена блок-схема методу показана на рисунку 2.8.

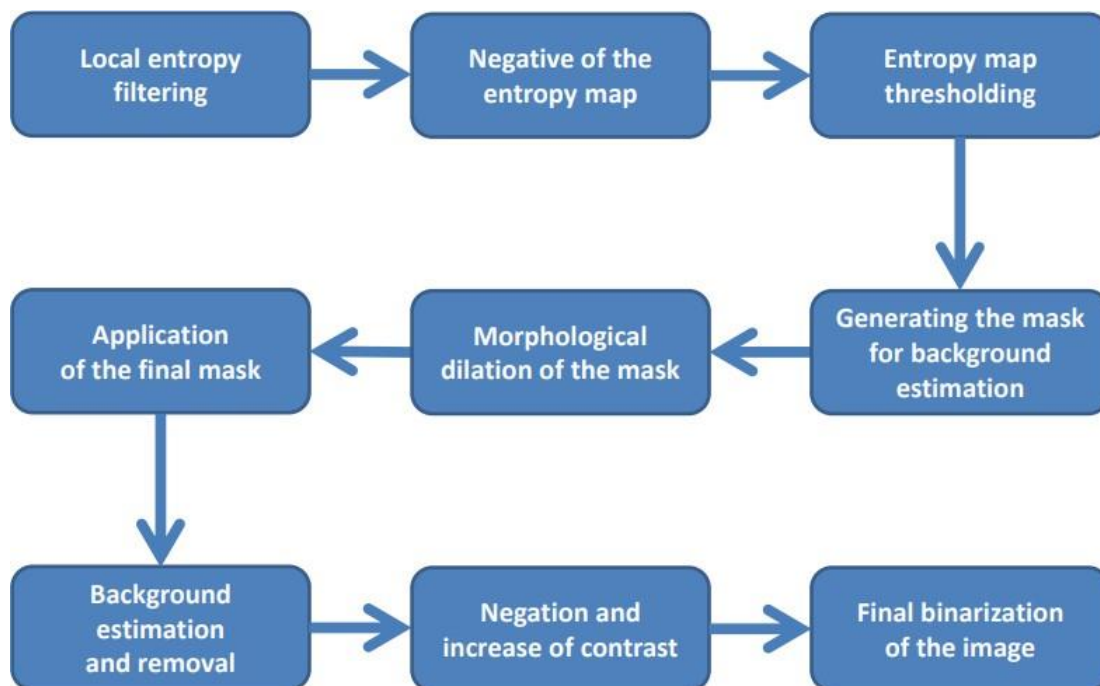


Рисунок 2.8 – Спрощена блок-схема запропонованого способу

Вибравши найбільш підходящі параметри запропонованого методу попередньої обробки, отримані значення F-міри та відстані Левенштейна для всього набору даних і кожного з них порівнювалися типи підсвічування, а також окремі шрифти та модифікації стилю деякі інші методи, що застосовуються без запропонованої попередньої обробки. Порівняння впливу запропонованого методу попередньої обробки за значеннями F-міри представлено в таблиці 1, тоді як відповідні відстані Левенштейна наведені в таблиці 2. Аналізуючи результати, значне зменшення відстані Левенштейна, а також збільшення значень F-міри можна спостерігати для всіх методи, що доводять корисність запропонованого підходу. Найкращих результатів досягла Niblack, Пороги Саувола і Вольфа, а також простий метод Меантреша, що було значно покращено використанням попередньої обробки на основі ентропійної фільтрації.

3 РЕАЛІЗАЦІЯ МЕТОДУ

3.1 Вирішення задачі

Рукописні зображення персонажів фіксуються за допомогою цифрової камери. Зображення персонажів можуть також сканувати за допомогою сканера. Цей процес відомий як отримання зображення. Все рукописне зображення символів перетворюються в уніфікований формат зображення, наприклад .bmp або .jpg, щоб створити всі зображення готовий до наступного етапу обробки. Може використовуватися чисто білий фон або якийсь кольоровий (шумний) фон щоб написати/роздрукувати ці рукописні зображення символів. Ці зразки можуть бути написані різними ручками різнокольорові чорнила. Зразки зображень персонажів, надані 10 різними людьми (віком 15-50 років) зібрано, де кожен учасник пише 5 зразків повного англійського алфавіту (a-z). Таким чином 1300 для запропонованого експерименту збираються зразки зображень символів ($10 \times 5 \times 26 = 1300$).

Попередня обробка.

Попередня обробка виконується для видалення мінливості, яка присутня в рукописних символах офлайн.

На цьому етапі попередньої обробки вхідне зображення рукописного символу у форматі .bmp з локального база даних, як показано на малюнку 1(a), перетворюється у формат відтінків сірого за допомогою функції «rgb2gray» MATLAB і отримане рукописне зображення символу показано на рисунку 3.1 (b).

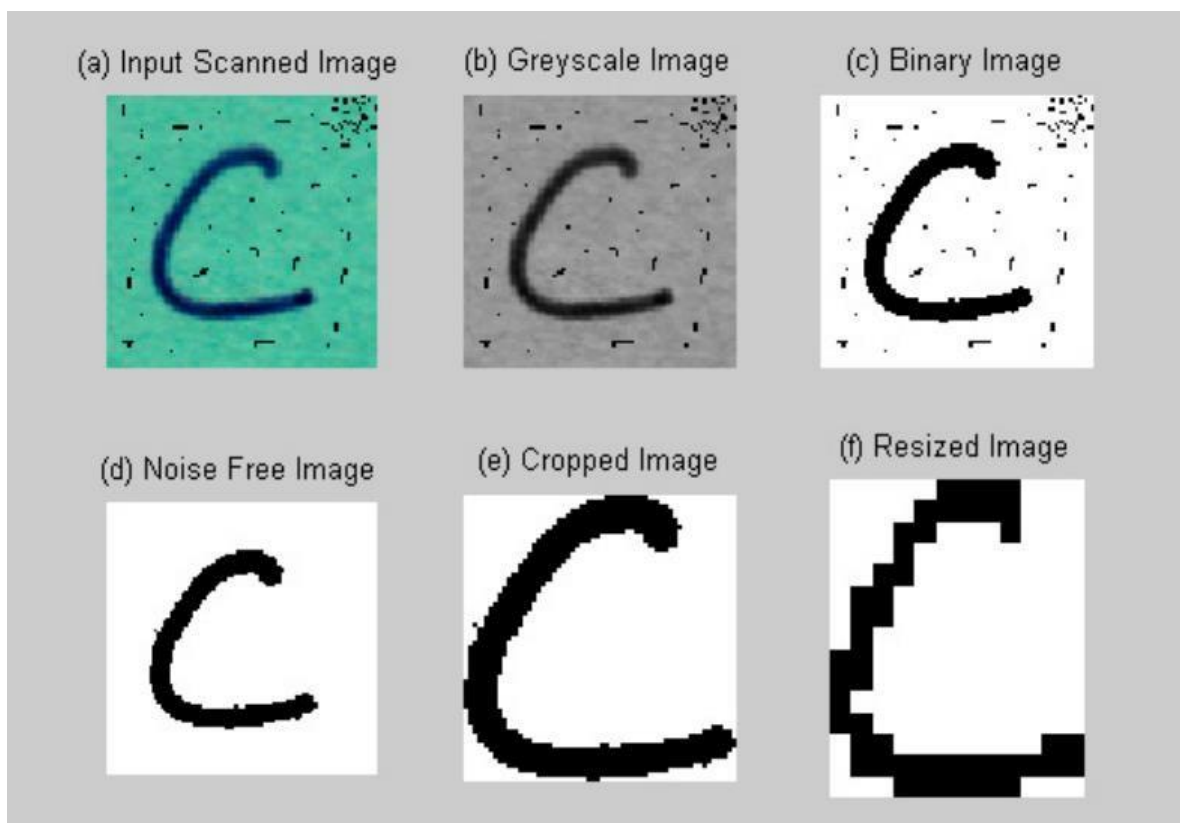


Рисунок 3.1 – (а) введення відсканованого рукописного зображення символу; (b) рукописне зображення символу в сірому форматі; (c) зображення символу в двійковому форматі формат; (d) зображення персонажа після видалення шуму переднього плану; (e) Обрізане зображення символу; (f) Змінений розмір зображення рукописного символу

Особливості вилучення та підготовки зразків для навчання.

Двійкове зображення символу «с» показано на малюнку 2(a). Його розмір змінено до матриці 15×12 , як показано на малюнку 2(b). «0» вказує на наявність білого пікселя, а «1» означає наявність чорного пікселя, як показано на двійкове матричне представлення символу «с» на рис. 2(c). Ця двійкова матриця розміром 15×12 потім змінюється перший спосіб рядка до двійкової матриці розміром 180×1 за допомогою функції «перетворення» MATLAB і показано на рисунок 3.2 (d). Цей вектор стовпця розміром 180×1 є вектором ознак символу «с».

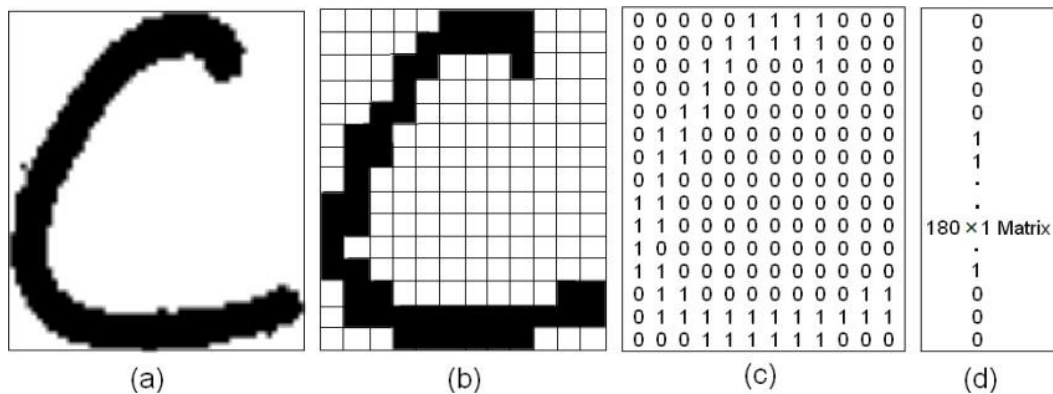


Рисунок 3.2 – (а) двійкове зображення символу «с»; (b) змінений розмір двійкового зображення символу «с»; (c) бінарне матричне представлення & (d) Вектор ознак символу «с»

Аналогічно вектори ознак усіх 26 символів (a-z) створюються у вигляді двійкової матриці стовпців розміром 180×1 кожна. Усі ці 26 векторів ознак об'єднані в двійкову матрицю розміром 180×26 , як показано на рисунку 3.3. Ця матриця називається зразком.



Рисунок 3.3 – Матричне представлення вхідної вибірки розміром 180×26

У цій матриці перший стовпець представляє вектор ознак символу «а», другий стовпець представляє ознаку вектор символу «b», третій стовпець представляє вектор ознак символу «с» тощо. Для того, щоб створити зразки, 1300 зображень персонажів зібрані з 10 авторів (вік 15-50 років), де кожен автор надав 5 зразків повного англійського алфавіту (a-z) ($10 \times 5 \times 26 = 1300$). Таким чином, кожен зразок складається з 26 англійських алфавітів. Усі ці зразки використовуються для навчання класифікатора нейронної мережі.

Розмір вхідного шару залежить від розміру зразка, представленого на вході, і розміру вихідний рівень визначається відповідно до кількості вихідних класів, в яких знаходиться кожен з вхідних шаблонів підлягати класифікації. У запропонованому експерименті вектор ознак кожного з зображень із 26 символів має розмір 180×1 . Отже, 180 нейронів використовуються у вхідному шарі, а 26 нейронів використовуються у вихідному шарі класифікатора нейронної мережі. Для досягнення оптимальних результатів 80 нейронів утримуються в прихованому шарі методом проб і помилок метод. Функція активації «tansig» використовується як для нейронів прихованого, так і для вихідного шару. Нейронна мережа тренувальний процес показаний на рис. 4 і використана функція адаптивного навчання «traingdx». Середній квадрат Помилка (MSE) була обрана як функція вартості в процесі навчання, показаному на рисунку 3.4.

У випадку нейронних мереж із зворотним поширенням, загальноприйнята функція вартості для вимірювання продуктивність узагальнення MSE. Менше значення функції вартості вказує на те, що нейронна мережа є здатний правильно відобразити вхід і вихід. Прийнятний поріг для MSE (вартість значення функції) було обрано як 0,001, і навчання нейронної мережі завершиться, коли помилка стає меншою або дорівнює цьому пороговому значенню. Значення продуктивності вказує на ступінь навчання мережі. Низьке значення

продуктивності (0,000865) вказує на те, що мережа пройшла навчання правильно.

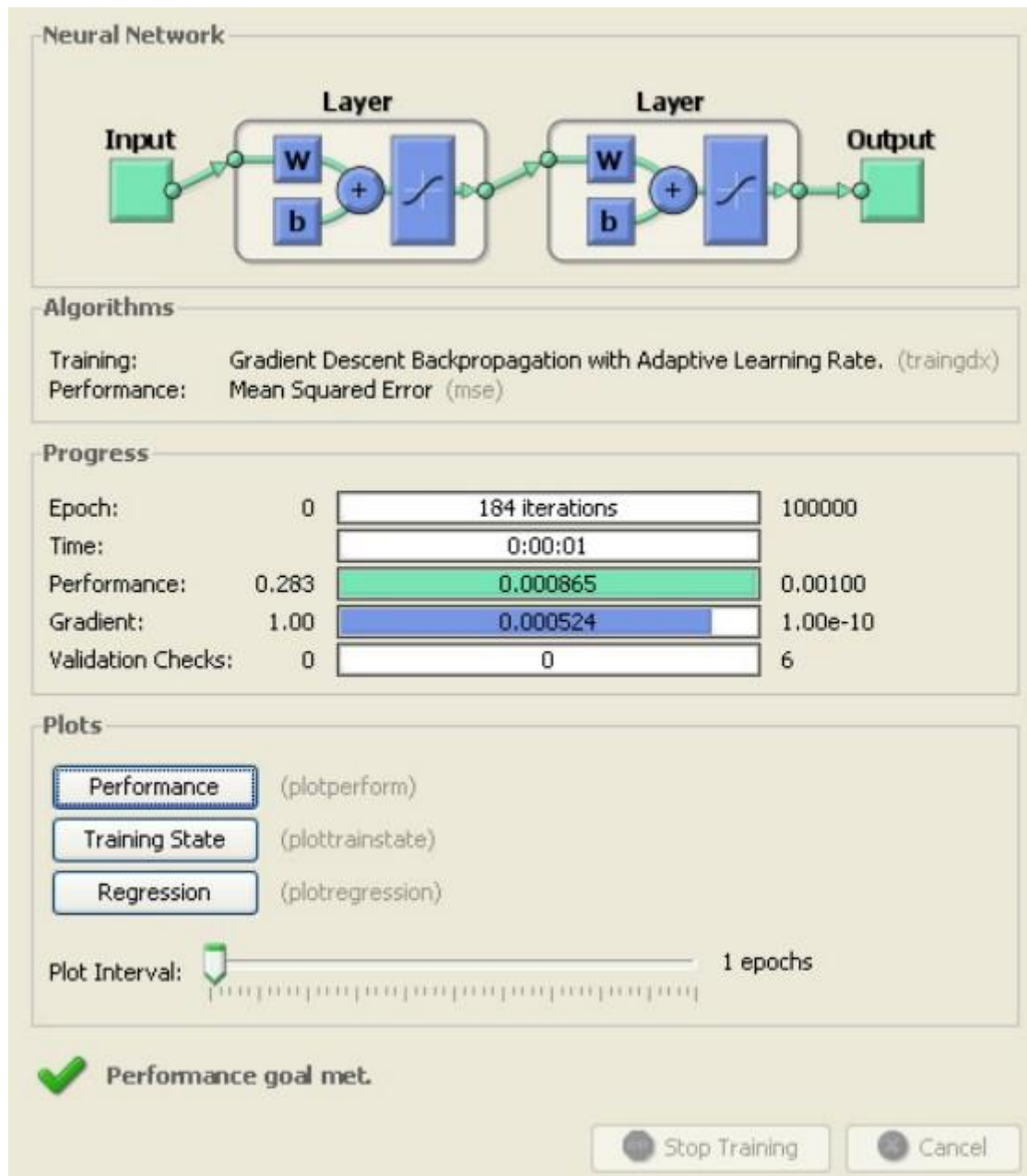


Рисунок 3.4 – Навчальний процес мережі

У випадку реальних додатків продуктивність класифікатора нейронної мережі також залежить від кількості навчальних ітерацій, необхідних для навчання мережі. Занадто менша кількість епох навчання призводить до погано підготовленої мережі через недостатнє підібрання

мережі. З іншого боку, виникає занадто багато епох навчання у поганому узагальненні через переобладнання мережі. Ітерації навчання мережі необхідно вибрати в таким чином, щоб мережа могла правильно сходитися з найменшою помилкою узагальнення. Максимально дозволений епох для навчального процесу встановлено на 100 000, як показано на рис. 4. Якщо мережа не може сходитися в межах максимально дозволеної кількості епох навчання припиняється.

Результат.

Рівно 50 зразків зображення кожного персонажа готуються до навчального процесу та представлені класифікатором нейронних мереж. Кожен шаблон символу, представлений на вхідному шарі, поставить «1» лише одному нейрону вихідного шару, до якого існує найвища довіра. На всіх нейронах, що залишилися, ставиться «0». Для кожного шаблону символу на вході, на виході є матриця стовпців 26×1 , в якій «1» присутній на одному лише місці, а решта 25 записів – «0», наприклад, символ "a" призводить до (1, 0, 0 ..., 0) і символ "b" призводить до (0, 1, 0 ..., 0) тощо. Таким чином, усі окремі символи на вході представлені вектори-стовпці розміром 26×1 кожен на виході. Оскільки вибірка містить 26 символів, вихід вибірки на вході представлена матриця 26×26 . У запропонованому експерименті з розпізнавання рукописних символів нейронна мережа була навчена кожним з 26 символів 50 разів, тобто 1300 ($50 \times 26 = 1300$) символів зразків зображення з бази даних було беруть участь у процесі навчання. Невизначеність розпізнавання між різними символами представлена на рис. 5. Символ «a» розпізнається точно 43 рази з 50. З 7 помилок, символ «a» класифікується як «e», 2 рази і «o», 5 разів відповідно. Середня загальна точність розпізнавання 85,62% – це спокійно для цього експерименту з розпізнавання рукописних символів, як показано у вигляді матриці представляє плутанину між різними англійськими алфавітами, як показано на рисунку 3.5.

Alphabets	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	Success (%)
a	43	0	0	0	0	0	0	1	0	0	0	0	2	8	5	0	0	0	0	0	0	3	0	0	0	0	86
b	0	49	0	2	0	0	0	0	0	0	3	0	0	0	5	0	0	0	0	0	0	0	0	0	0	98	
c	0	0	44	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	88	
d	0	1	0	38	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	76	
e	2	0	4	0	44	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	88	
f	0	0	0	0	0	46	0	0	0	0	0	3	0	0	0	0	0	0	0	1	0	0	0	0	1	92	
g	0	0	0	0	0	0	39	0	0	10	0	0	0	0	1	4	0	0	0	0	0	0	0	13	0	78	
h	0	0	0	0	0	0	0	49	0	0	5	0	4	0	0	0	0	0	0	0	0	0	0	0	0	98	
i	0	0	0	0	0	0	0	0	37	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	74	
j	0	0	0	0	0	0	1	0	5	29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	58	
k	0	0	0	0	0	0	0	0	0	0	42	2	0	0	0	0	0	0	0	1	0	0	0	1	0	84	
l	0	0	0	1	0	1	0	0	3	0	0	39	0	0	0	0	0	0	0	0	0	0	0	0	0	78	
m	0	0	0	0	0	0	0	0	0	0	0	41	2	0	0	0	1	1	0	0	0	0	0	0	0	82	
n	0	0	0	0	0	0	0	0	0	0	0	0	0	38	0	0	0	1	0	0	0	0	0	0	0	76	
o	5	0	1	0	1	0	0	0	0	0	0	0	0	0	43	0	0	0	1	0	0	0	0	0	0	86	
p	0	0	0	5	0	0	0	0	0	0	0	0	0	0	0	44	0	0	0	10	0	0	0	0	0	88	
q	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	46	0	0	0	0	0	0	0	1	0	92	
r	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	49	0	0	0	0	0	0	0	98	
s	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	46	0	0	0	0	0	0	92	
t	0	0	0	4	0	3	0	0	5	0	6	0	0	0	0	0	0	0	38	0	0	0	0	0	2	76	
u	0	0	1	0	1	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	48	5	0	0	0	96	
v	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	40	0	0	0	0	80	
w	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	50	0	0	0	0	100	
x	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	1	0	0	0	0	0	50	1	100	
y	0	0	0	0	0	0	8	0	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	35	70	
z	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	46	92	
Overall Character Recognition Accuracy =																									85.62 %		

Рисунок 3.5 – Матриця плутанини, що представляє продуктивність класифікатора нейронної мережі

ВИСНОВКИ

Було проведено аналіз технологій для зчитування рукописних символів. Виявлено недоліки та переваги. Гібридні нейронні мережі підходять для задач розпізнавання тексту, але дуже незручні для з'ясування питання, як вони здійснюють таке розпізнавання. Вони можуть автоматично отримати знання, а процес їх навчання часто відбувається досить повільно, та аналіз навчальної мережі дуже складний (навчера мережа зазвичай — чорний ящик для користувача). При цьому яку-небудь апріорну інформацію (знання експерта) для прискорення процесу її навчання в нейронну систему ввести неможливо.

Запропонована нейронна мережа на основі гібридних нейронних мереж ідентифікує рукописний символ. Шляхом використання алгоритму Левенберга для навчання, класифікація символів виконується з точністю 85% на основі розміру навчальних даних, кількості прихованих нейронів і кількості прихованих шарів.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. “Individuality of handwriting: A validation study,” in Proceedings of the Sixth International Conference on Document Analysis and Recognition, ICDAR '01, (Washington, DC, USA), pp. 106–, IEEE Computer Society, 2001.
2. R. A. Huber and A. Headrick, “Handwriting identification: facts and fundamentals,” 1999.
3. S.N. Srihari, Feature extraction for address block location, From Pixels to Features, J.C. Simon, editor, North Holland, Elsevier, Amsterdam, 1989, 261-274.
4. Srihari SN. Recognition of handwritten and machine-printed text for postal address interpretation. Pattern Recognition Letters 1993;14: 291303.
5. D. G. Lowe, “Distinctive image features from scale-invariant keypoints,” Int. J. Comput. Vision, vol. 60, pp. 91–110, Nov. 2004.
6. J. Rodriguez-Serrano and F. Perronnin., “Local gradient histogram features for word spotting in unconstrained handwritten documents.,” ICFHR, Montreal, 2008.
7. S. Dey, A. Dutta, J. I. Toledo, S. K. Ghosh, J. Lladós, and U. Pal, “Signet: Convolutional siamese network for writer independent offline signature verification,” CoRR, vol. abs/1707.02131, 2017.
8. A. Lebedev, V. Khryashchev, A. Priorov, and O. Stepanova, “Face verification based on convolutional neural network and deep learning,” in 2017 IEEE East-West Design Test Symposium (EWDTS), pp. 1–5, Sept 2017.
9. Винокурова, Е.А. Интеллектуальный анализ и визуализация нечетких данных на основе метода главных компонент / Е.А. Винокурова, Н.В. Касаткина, В.А. Филатов // Вестник Херсонского национального технического университета, 2010. – № 2 (38) – С. 154-158.

10. Filatov V. Fuzzy models presentation and realization by means of relational systems // *Econtechmod : an international quarterly journal on economics in technology, new technologies and modelling processes.* – Lublin ; Rzeszow, 2014. – Vol.(3), № 3. – P. 99-102.
11. J. Bromley, I. Guyon, Y. LeCun, E. Sackinger, and R. Shah, “Signature verification using a “siamese” time delay neural network,” in *Proceedings of the 6th International Conference on Neural Information Processing Systems, NIPS’93, (San Francisco, CA, USA), pp. 737-744, Morgan Kaufmann Publishers Inc., 1993.*
12. D. E. Rumelhart, G. E. Hinton, and R. J. Williams, “Parallel distributed processing: Explorations in the microstructure of cognition, vol. 1,” ch. *Learning Internal Representations by Error Propagation*, pp. 318-362, Cambridge, MA, USA: MIT Press, 1986.
13. Filatov V., Voloshchuk O., Spivak N. Implementation and support fuzzy systems by means the relational data model, «European Cooperation», Vol 4, No 11 (2016). – P. 49-61.
14. K. Yan, Y. Wang, D. Liang, T. Huang, and Y. Tian, “Cnn vs. sift for image retrieval: Alternative or complementary?,” in *Proceedings of the 2016 ACM on Multimedia Conference, MM ’16, (New York, NY, USA), pp. 407–411, ACM, 2016.*
15. Doskalenko S. N. On the Approach to Searching for Functional Dependences of Data in Relational Systems / S. N. Doskalenko, V. A. Filatov. // *Innovative technologies and scientific solutions for industries.* Kharkiv. 2018. No. 3 (1). P. 54–58. doi:10.30837/2522-9818.2018.3.054
16. V. Chandrasekhar, J. Lin, O. Morere, H. Goh, and A. Veillard, “A practical guide to cnns and fisher vectors for image instance retrieval,” *CoRR*, vol. abs/1508.02496, 2015.
17. Filatov V., Semenets V. Methods for Synthesis of Relational Data Model in Information Systems Reengineering Problems // *International Scientific-Practical Conference «Problems of Infocommunications. Science and Technology» (PIC S&T-2018) Kharkiv, 2018. doi: 10.1109/INFOCOMMST.2018.8632144*

18. G. Zhang, Z. Zeng, S. Zhang, Y. Zhang, and W. Wu, "Sift matching with cnn evidences for particular object retrieval," vol. 238, 02 2017.
19. L. Zheng, Y. Yang, and Q. Tian, "SIFT meets CNN: A decade survey of instance retrieval," CoRR, vol. abs/1608.01807, 2016.
20. C. Huang and S. N. Srihari, "Mapping Transcripts to Handwritten Text," in Tenth International Workshop on Frontiers in Handwriting Recognition (G. Lorette, ed.), (La Baule (France)), Universite de Rennes 1, Suvisoft, Oct. 2006. <http://www.suvisoft.com>.
21. Інтелектуальні технології в медичній діагностиці, лікуванні та реабілітації: монографія / [С. В. Павлов, О.Г. Аврунін, В.О Філатов, Є.В. Бодянський та ін.]; за редакцією С. Павлова, О. Авруніна. – Вінниця: ПП «ТД «Едельвейс і К», 2019. – 260 с. ISBN 978-617-7237-59-3
22. K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," CoRR, vol. abs/1409.1556, 2014.
23. A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in Advances in Neural Information Processing Systems 25 (F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, eds.), pp. 1097–1105, Curran Associates, Inc., 2012.
24. K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," CoRR, vol. abs/1512.03385, 2015.
25. C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," CoRR, vol. abs/1512.00567, 2015.
26. Filatov, V., Semenets, V. and Zolotukhin, O. (2020) "Data mining in relational systems", Innovative Technologies and Scientific Solutions for Industries, (3 (13), pp. 65-76. doi: 10.30837/ITSSI.2020.13.065
27. M. Muja and D. G. Lowe, "Fast approximate nearest neighbors with automatic algorithm configuration," in In VISAPP International Conference on Computer Vision Theory and Applications, pp. 331–340, 2009.

28. Filatov, V. Model of semantic integration of information systems properties in relay database reengineering problems / O. Avrunin, O. Vlasov, V. Filatov. // Innovative Technologies and Scientific Solutions for Industries, 2020 - (4 (14)), 5-12. doi:10.30837/itssi.2020.14.005
29. N. Kanopoulos, N. Vasanthavada, and R. L. Baker, "Design of an image edge detection filter using the sobel operator," IEEE Journal of SolidState Circuits, vol. 23, pp. 358–367, Apr 1988.
30. J. T. Favata and G. Srikantan, "A multiple feature/resolution approach to handprinted digit and character recognition," vol. 7, pp. 304-311, 12 1998.