

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Інформаційно-аналітичних технологій та менеджменту

(повна назва)

Кафедра Інформатики

(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА **Пояснювальна записка**

рівень вищої освіти другий (магістерський)

ДОСЛІДЖЕННЯ ТА ПОРІВНЯННЯ МЕТОДІВ РОЗПІЗНАВАННЯ

РЕКВІЗИТІВ БАНКІВСЬКИХ КАРТОК

(тема)

Виконав:

студент 2 курсу, групи ІНФМ-21-1

Бабочкін О.О.

(прізвище, ініціали)

Спеціальності 122 Комп'ютерні науки

(код і повна назва спеціальності)

Тип програми освітньо-професійна

Освітня програма Інформатика

(повна назва освітньої програми)

Керівник доц. Творошенко І.С.

(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри

(підпис)

Кобилін О.А.

(прізвище, ініціали)

2022 р.

Харківський національний університет радіоелектроніки

Факультет Інформаційно-аналітичних технологій та менеджменту
(повна назва)

Кафедра Інформатики
(повна назва)

Рівень вищої освіти другий (магістерський)

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

Тип програми освітньо-професійна

Освітня програма Інформатика
(повна назва освітньої програми)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)

« ____ » _____ 2022 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові Бабочкіну Олегу Олександровичу
(прізвище, ім'я, по батькові)

1. Тема роботи Дослідження та порівняння методів розпізнавання реквізитів банківських карток

затверджена наказом по університету від 9 листопада 2022 року № 1469Ст

2. Термін подання студентом роботи до екзаменаційної комісії 21 листопада 2022 р.

3. Вихідні дані до роботи науково-методична література, матеріали конференцій, дані інтернет-мережі, бібліотека OpenCV, мова Python, мова Swift, датасет фотографій банківських кредитних карток, теоретичні відомості про методи розпізнавання реквізитів банківських карток.

4. Перелік питань, що потрібно опрацювати в роботі _____

1. Аналіз існуючих методів розпізнавання реквізитів банківських карток.

2. Аналіз особливостей розроблення методів розпізнавання реквізитів банківських карток.

3. Реалізація та порівняння методів розпізнавання.

4. Тестування методів та аналіз отриманих результатів.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (п.5 включається до завдання за рішенням випускової кафедри) актуальність проблеми розпізнавання банківських карток, об'єкт та мета дослідження, постановка задачі, етапи розроблення, апробація результатів роботи, набір тестових зображень.

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
Консультант з дотримання діючих стандартів та норм	Доцент Творошенко І.С.		

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Отримання завдання на кваліфікаційну роботу	09.11.2022	
2	Аналіз завдання, підбір літератури	10.11.22-12.11.22	
3	Аналіз літератури з досліджуваної проблеми	12.11.22-13.11.22	
4	Аналіз сучасних систем розпізнавання реквізитів банківських карток	13.11.22	
5	Аналіз методів розпізнавання банківських карток	14.11.22	
6	Програмна реалізація	15.11.22-18.11.22	
7	Оформлення пояснювальної записки	18.11.22-19.11.22	
8	Перевірка на плагіат	24.11.2022	
9	Рецензування	25.11.2022	
10	Підготовка презентації та доповіді	27.11.2022	
11	Занесення роботи в електронний архів	29.11.2022	
12	Попередній захист кваліфікаційної роботи	30.11.2022	

Дата видачі завдання 9 листопада 2022 р.

Студент _____
(підпис)

Керівник роботи _____ доц. Творошенко І.С.
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ/ABSTRACT

Пояснювальна записка до кваліфікаційної роботи: 78 с., 2 табл., 37 рис., 1 дод, 64 джерела.

ІДЕНТИФІКАЦІЯ ОБ'ЄКТІВ, РЕКВІЗИТИ БАНКІВСЬКОЇ КАРТКИ, OCR, SWIFT, XCODE, АЛГОРИТМ ЛУНА, PYTHON, OPENCV, VISA, MASTERCARD.

Об'єктом дослідження є датасет різноракурсних зображень банківських карток.

Метою дослідження є аналіз та порівняння методів розпізнавання реквізитів банківських карток.

Використано методи, що застосовують виявлення ключевих точок банківської картки на зображенні. Проведено дослідження методів ідентифікації банківських карток на зображенні. Запроваджено метод детектування банківських карток за допомогою методи бібліотек OpenCV та мови Swift.

У результаті роботи здійснена програмна реалізація двох методів у вигляді програмних застосунків, що виконують ідентифікацію номеру банківської кредитної картки на основі вхідного зображення за допомогою методів бібліотеки OpenCV і мови Python та мови Swift.

Результати роботи апробовано у вигляді тез доповідей під час XXXVII Міжнародної науково-практичної конференції у м. Варна, Болгарія.

OBJECT IDENTIFICATION, BANK CARD DETAILS, OCR, SWIFT, XCODE, LUHN ALGORITHM, PYTHON, OPENCV, VISA, MASTERCARD.

The object of the research is a dataset of bank card images.

The purpose of the research is to analyze and compare methods of recognizing bank card details.

Methods used to detect the key points of the bank card in the image were used. A study of bank card identification methods on the image was conducted. The method of detecting bank cards using the method of OpenCV libraries and the Swift language has been introduced.

As a result of the work, the software implementation of two methods was carried out in the form of software applications that perform bank credit card number identification based on the input image using methods of the OpenCV library and the Python and Swift languages.

The results of the work were tested in the form of abstracts of reports during the XXXVII International Scientific and Practical Conference in Varna, Bulgaria.

ЗМІСТ

Перелік умовних позначень, символів, одиниць, скорочень і термінів	6
Вступ	7
1 Аналіз існуючих методів розпізнавання реквізитів банківських карток	8
1.1 Аналіз сучасних систем розпізнавання реквізитів банківських карток	8
1.2 Класифікація та аналіз існуючих методів розпізнавання реквізитів банківських карток	10
1.3 Особливості реквізитів банківських карток	19
1.4 Аналіз літературних джерел щодо апробації результатів розпізнавання реквізитів банківських карток	22
1.5 Постановка задачі дослідження.....	32
2 Особливості розроблення методів розпізнавання реквізитів банківських карток	34
2.1 Механізм оброблення зображення банківських карток	34
2.2 Принципи розпізнавання складових банківських карток	36
2.3 Методика розпізнавання реквізитів банківських карток	39
3 Реалізація та порівняння методів розпізнавання реквізитів банківських карток	44
3.1 Вибір інструментального засобу для реалізації методів розпізнавання реквізитів банківських карток	44
3.2 Етапи програмної реалізації методів розпізнавання реквізитів банківських карток	49
3.3 Тестування розроблених застосунків та аналіз результатів	54
3.4 Перспективи подальшої роботи	61
Висновки.....	63
Перелік джерел посилання.....	64
Додаток А Тестові зображення.....	71

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

США – Сполучені Штати Америки

OCR – Optical Character Recognition (оптичне розпізнавання символів)

ШІ – штучний інтелект

API – Application Programming Interface (прикладний програмний інтерфейс)

IBAN – International Bank Account Number (міжнародний номер банківського рахунку)

CVV – Card Verification Value (значення верифікації картки)

AVS – Address Verification System (адресна верифікаційна система)

ANSI – American National Standards Institute (Американський інститут національних стандартів)

ISO – International Organization for Standardization (Міжнародна організація зі стандартизації)

IIN – Issuer Identification Number (ідентифікаційний номер емітента)

BIN – Bank Identification Number (ідентифікаційний номер банку)

MII – Major Industry Identifier (ідентифікатор основної галузі)

SMS – Short Message Service (служба коротких повідомлень)

SVM – State Vector Machine (машина станів вектору)

PDF – Portable Document Format (портативний формат документа)

JSON – JavaScript Object Notation (запис об'єктів JavaScript)

ВСТУП

На вебсайтах та у мобільних засосунках користувачі часто здійснюють покупки, вводячи дані своєї кредитної картки. Ймовірно, кожен мав неприємний досвід введення 16 цифр номеру банківської картки вручну в при онлайн оплаті, адже вірогідність помилки при ручному вводі висока.

Багато програм додають автоматизацію, щоб спростити цей процес. Таким чином, щоб ввести платіжні дані, користувачі можуть сфотографувати свою кредитну картку або завантажити фотографію з фотогалереї свого пристрою.

Взагалі, зростання штучного інтелекту в останні роки було зумовлене явищем цифровізації, яке торкнулося всіх професійних середовищ. Ця цифрова трансформація була ініційована більшістю компаній, великих і малих, і одним із головних напрямків трансформації є оцифрування даних. Саме з цією метою була розроблений такий сервіс комп'ютерного зору як оптичне розпізнавання символів, широко відоме як OCR.

Виникнення OCR сягає 1950-х років, коли Девід Шепард заснував Intelligent Machines Research Corporation, першого у світі постачальника систем OCR, які керували приватні компанії для перетворення друкованих повідомлень на машинну мову для комп'ютерної обробки.

Сервіси OCR є інтелектуальними, і OCR навіть є однією з найважливіших галузей комп'ютерного зору та штучного інтелекту загалом. Завдяки OCR можна отримати текстовий файл із багатьох цифрових носіїв, наприклад, файл PDF, PNG, JPG зображення, що містить тексти та рукописні документи.

Тож можна зробити висновок, що системи із розпізнавання банківських карток мають високий потенціал досліджень та розробки. Але для кожного випадку використання оптимально порівняти різні рішення OCR, щоб вибрати найбільш ефективне рішення. Адже багато параметрів впливають на продуктивність різних рішень: якість зображення, формат зображення, рукописне/цифрове зображення, характер символів (літери, цифри, мова).

1 АНАЛІЗ ІСНУЮЧИХ МЕТОДІВ РОЗПІЗНАВАННЯ РЕКВІЗИТІВ БАНКІВСЬКИХ КАРТОК

1.1 Аналіз сучасних систем розпізнавання реквізитів банківських карток

Фінансова галузь є однією із найбільш цифровізованих сфер. За офлайн-магазинами закономірно йдуть онлайн-організації. На відміну від готівки, яка існує тут і зараз у матеріальному вигляді, банківські картки дозволяють швидко сплатити за куплений в Інтернеті товар лише увівши платіжні реквізити.

Кожний із нас щоденно здійснює різноманітні фінансові онлайн-операції, тому автоматизація процесу зчитування даних з банківських карток є актуальною задачею для будь-якого ресурсу, де користувачеві потрібно ввести власні платіжні реквізити (для оплати замовлень в інтернет-магазинах, онлайн-банкінгу, різних програмних застосунках). Щоб виконати зазначені операції необхідно вводити номер банківської картки, термін дії картки, ім'я власника картки, тип платіжної системи та CVV код (рис. 1.1) [1].

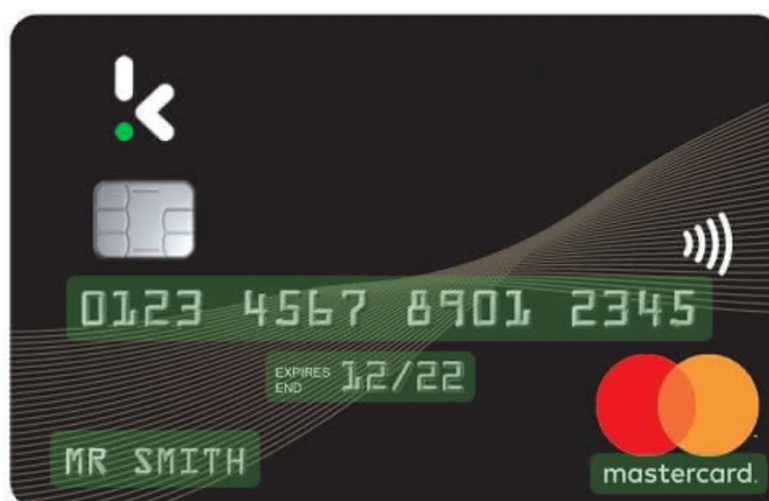


Рисунок 1.1 – Реквізити банківської картки

Єдина проблема криється у постійній необхідності пам'ятати та безпомилково вводити реквізити банківської картки.

Сучасні гаджети з їх нескінченними голосовими помічниками, модулями заповнення, підказками, розпізнаваннями, здається, вже повністю відучили людство щось вводити механічно. Це дійсно зручно, коли не треба заповнювати дані вручну зайвий раз. При здійсненні покупки доводиться вводити від 16 до 19 цифр, які складають номер банківської картки та слабо інтерпретуються з людської точки зору, що призводить до постійних помилок при наборі номеру.

З метою уникнення помилок при введенні, зменшення затрат часу для внесення даних, а також для покращення взаємодії з користувачем виникає необхідність у розробленні комп'ютерних систем оптичного розпізнавання реквізитів банківських карт.

Впровадження технологій оптичного розпізнавання реквізитів банківських карт має ряд переваг над ручним розпізнаванням. Це точність, швидкість та вартість впровадження такої технології. На жаль, повністю уникнути залучення людини у процес неможливо, але фактор її участі буде мінімальним.

Оптичне розпізнавання символів – це процес, що спрямований на оцифрування текстової інформації, яка автоматично ідентифікується із зображенням, а потім зберігається як дані.

Для розпізнавання символів використовують різні методи, серед яких варто відмітити такі як:

- матрична відповідність (є простішим та поширенішим);
- інтелектуальне розпізнавання символів.

Матрична відповідність (або порівняння з шаблоном) – це метод, суть якого полягає в порівнянні виявленого програмою оптичного символу з бібліотекою шаблонів [2]. Програмне забезпечення OCR зіставляє це зображення з відповідним символом ASCII [3].

Метод інтелектуального розпізнавання символів – це метод оптичного розпізнавання тексту, який використовує комп'ютерний інтелект для пошуку характерних ознак, таких як відкриті області, замкнуті фігури, діагональні лінії, перетин ліній тощо [4].

Розпізнавання даних із кредитної картки одночасно є актуальним і дуже цікавим з погляду алгоритмів завданням. Добре реалізована програма розпізнавання пластикових карток може позбавити людину необхідності вводити значну частину даних вручну при здійсненні інтернет-платежів і платежів у мобільних застосунках.

З точки зору розпізнавання, банківська картка – це складний документ стандартного розміру (85,6 мм × 53,98 мм), виконаний на типовому бланку, що містить певний набір полів (як обов'язкових, так і додаткових): номер картки, ім'я власника картки, дата видачі, термін дії, номер облікового запису, CVV2-код або його аналог. Частина полів знаходиться на лицьовій стороні, а деякі – на звороті. І, незважаючи на те, що для здійснення платіжної транзакції потрібно вказати тільки номер картки, практично всі платіжні системи (як автентифікація) додатково вимагають вказати ім'я власника картки, термін дії та CVV2-код [5].

Розробками алгоритмів для розпізнавання інформації з банківських карт на сьогоднішній день займаються ряд компаній, зокрема, LabsNeural, Sinosecu, Huawei, SmartEngine, Klipra та інші, а також розроблено програмні застосунки, які за власними алгоритмами «видобувають» реквізити банківських карт.

Таким чином, технологія розпізнавання банківських карток насправді виявляється передовою і вкрай необхідною за умови гідної реалізації.

Зважаючи на точність розпізнавання інформації, дослідження технологій оптичного розпізнавання реквізитів банківських карт все ще залишається актуальною задачею і вимагає подальшого вивчення.

1.2 Класифікація та аналіз існуючих методів розпізнавання реквізитів банківських карток

Все більше і більше компаній сьогодні спрощують і прискорюють бізнес-процеси за допомогою певної автоматизації.

Наприклад, у глобальному дослідженні McKinsey 2020 року [6] дві третини респондентів заявили, що їхні організації принаймні пілтують автоматизацію бізнес-процесів в одному або кількох бізнес-підрозділах або функціях, порівняно з 57% двома роками раніше.

Одним із основних методів визначення даних банківських карток є Optical Character Recognition (OCR), що розшифровується як «оптичне розпізнавання символів» [7]. Іноді його також називають технологією «оптичного зчитування символів».

Ресурс TechTerms визначає оптичне розпізнавання символів як «технологію, яка розпізнає текст у цифровому зображенні» [8]. Вона часто використовується для розпізнавання тексту в сканованих паперових документах.

Технологія оптичного розпізнавання символів знаходить і розпізнає символи, наприклад, цифри, літери чи символи у цифровому зображенні. Після цього програмне забезпечення може експортувати текст або перетворювати символи на редагований текст у файлі зображення, щоб текст можна було використовувати для обробки даних.

З розпізнаванням шаблонів програма оптичного розпізнавання символів використовує приклади різних шрифтів і форматів тексту, які вона надала, щоб порівняти та ідентифікувати символи у відсканованому документі [9].

За допомогою виявлення ознак програма оптичного розпізнавання символів має набір правил щодо конкретних особливостей кожної літери чи цифри і використовує ці правила для ідентифікації символів у відсканованому документі.

Щоб здійснити ці порівняння, програма спирається на такі характеристики [10]:

- кількість кривих;
- кількість перехресних ліній;
- кількість кутових ліній у символі.

Коли програма OCR ідентифікує символ, цей символ перетворюється на код, який комп'ютерні системи можуть використовувати для подальшої обробки даних.

Оптичне розпізнавання символів можна використовувати для автоматичної обробки та керування даними кредитної картки. Якщо особа коли-небудь здійснювала платіж за допомогою свого мобільного телефону та бачила опцію сфотографувати свою кредитну картку, а не вручну вводила номер картки, то саме ця функція працює за допомогою оптичного розпізнавання символів [11].

На основі зображення кредитної картки (наприклад, фотографування кредитної картки в мобільній програмі) програмне забезпечення OCR знаходить та ідентифікує різні символи на картці, а потім перетворює їх на дані, які можна легше обробити.

Окрім обробки даних кредитної картки, оптичне розпізнавання символів можна використовувати в багато інших способах. На рисунку 1.2 наведено кілька прикладів такого використання [12].

Незважаючи на те, що технологія OCR спочатку була розроблена для читання друкованого тексту, вона також може ідентифікувати рукописний текст. Поштова служба Сполучених Штатів Америки (США) використовує оптичне розпізнавання символів для обробки пакунків і листів на основі адреси одержувача, зазначеної на відправленні.

OCR також використовується для архівування історичних документів, таких як газети чи журнали, у форматах даних з можливістю пошуку [13].

В останні роки стало можливим внести чек, просто сфотографувавши його на мобільний телефон і надіславши через банківський застосунок. OCR – технологія, яка дозволяє це зробити.

Штучний інтелект (ШІ) і машинне навчання забезпечують подальший розвиток ринку автоматизації [14]. Ці технології дозволяють автоматизувати зростаючу кількість бізнес-процесів, від миттєвих платежів до перевірок ідентифікації.

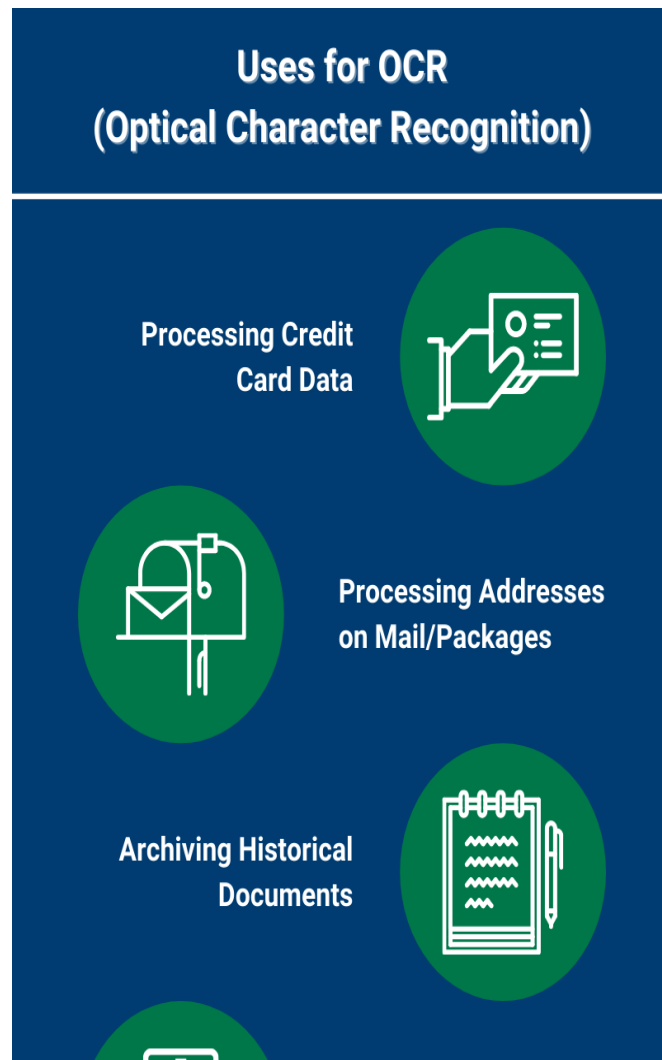


Рисунок 1.2 – Застосування методу OCR

Перш ніж звернути увагу на бізнес-процеси, які можна автоматизувати, варто знати, які банківські картки можна сканувати.

Сканування карток працює в основному з усіма видами банківських карток, які містять друковані або тиснені тексти, цифри та логотипи. Ці елементи можна зчитувати, дані можуть бути витягнуті з них автоматично. Таким чином, рішення для сканування смарт-карт є надзвичайно зручним у ручному процесі зчитування, а також для перевірки банківських карток і відповідного введення даних, яке, зазвичай, виконується клерками [15].

Найпоширенішими банківськими картками є кредитні та дебетові картки.

Кредитні картки зручні для повсякденних покупок і мають ту перевагу, що вони застосовуються для покупок, коли людина подорожує за кордон.

Картки дозволяють позичати гроші в емітента картки до певного ліміту, щоб придбати товари або зняти готівку.

З кредитних карток можна отримати такі поля даних:

- тип картки;
- мережа обробки кредитних карток;
- емітент кредитної картки;
- ім'я власника картки;
- номер картки або PAN;
- термін придатності;
- CVC (опціонально);
- підпис (бета).

Дебетові картки, як правило, легше отримати та встановити бюджет, тобто не можна витратити більше грошей, ніж є. Вони дозволяють витратити гроші, залучаючи кошти, які внесено в банк [16].

Отже, автоматичне сканування кредитних карток має такі переваги:

- зниження витрат;
- збільшення швидкості;
- зменшення кількості помилок;
- запобігання шахрайству.

Крім того, що рішення для сканування карток повністю автоматизоване без будь-яких дій вручну, воно ще має багато практичних переваг, які допоможуть бізнесу рухатися вперед.

Перевірка та обробка банківських карток вручну може бути досить трудомістким процесом. Адже це включає в себе такі процеси як ручне введення даних, перевірка картки та вимог до архівування, які супроводжуються цим.

Рішення щодо сканування карток може заощадити багато часу та, отже, трудових витрат у цьому процесі, саме завдяки автоматичному вилученню даних із банківських карток.

Інтелектуальна автоматизація, зазвичай, призводить до економії коштів від 40% до 75%, а період окупності коливається від кількох місяців до кількох років [17].

Збільшення швидкості полягає в тому, що автоматизовані процеси швидші за ручні. Такий процес, як перевірка кредитної картки займає багато часу, якщо його буде виконувати людина. Коли це робиться вручну, рецензенти мають переглянути кілька документів, переконатися, що інформація правильна та перевірити на шахрайство. Люди не такі швидкі, як комп'ютери, і автоматизація цього процесу може заощадити багато часу та грошей.

Зазвичай рівень помилок коливається від 0,55% до 3,6%, хоча було виявлено, що рівень помилок іноді досягає 26,9% [18]. Введення тексту на смартфоні є теж не комфортним з точки зору досвіду користувача, тому ці показники можуть бути ще вищими для мобільних пристроїв.

Гарним показником є те, що автоматизовані процеси та комп'ютерні програми роблять це значно краще. Для сканування кредитних карток найкраще підходить програмне забезпечення для вилучення даних з банківських карток, тим самим мінімізується ризик людської помилки.

Також важливим є запобігання шахрайству. Рішення для сканування карток включає не лише технологію оптичного розпізнавання символів, але й функцію виявлення фальсифікації зображень на основі штучного інтелекту. Воно передбачає виявлення неузгоджених піксельних структур, наприклад, дивних змін освітлення або раптових змін кольору в тексті. У результаті програмне забезпечення виявляє підроблені зображення за лічені секунди, полегшуючи фільтрацію шахрайських програм, перш ніж вони можуть спричинити проблеми.

Крім того, система може перехресно перевірити інформацію на картці особи із зовнішніми ресурсами. Наприклад, якщо в наявності є чорні списки або бази даних шахрайства з відомими шахрайськими картками, до них можна звертатися автоматично [19].

Технології намагаються отримати якомога більше даних із дебетових і кредитних карток за допомогою OCR і машинного навчання.

Типові випадки використання автоматичного сканування дебетових і кредитних карток включають у себе:

- спрощення оплати кредитною карткою;
- цифрова адаптація клієнтів за допомогою кредитних карток;
- автоматична перевірка віку за допомогою сканування кредитної картки;
- підтвердження банківського рахунку (IBAN) за допомогою сканування дебетової картки;
- автоматична анонімізація даних банківської картки [20].

Спрощення оплати кредитною карткою.

На багатьох вебсайтах і застосунках користувачі повинні вводити номери своїх кредитних карток для здійснення покупок. Це може бути відносно складним завданням, шанс зробити помилку збільшується.

Збирати інформацію про кредитні картки так само просто, як і за допомогою сканера кредитних карток на основі штучного інтелекту. Користувачі швидко переглядають платіжні форми, просто сфотографувавши або завантаживши PDF-файл своєї кредитної картки в API. За допомогою правильних моделей OCR і вилучення даних поля ідентифікуються, локалізуються, а текст у них автоматично витягується та вводиться в платіжні форми [21].

Інтерфейс програми або оформлення замовлення значно покращиться, оскільки користувачам більше не доведеться самостійно вводити інформацію. Зі сканером кредитних карток заповнювати форми стане легше, що призведе до вищих показників успішності прибутку та, зрештою, вищих доходів.

Цифрова адаптація клієнтів за допомогою кредитних карток.

Щоб забезпечити стабільну та відповідну послугу, багато платіжних і торгових онлайн-платформ вимагають дані кредитної картки особи під час реєстрації.

Основна проблема полягає в тому, щоб переконатися, що особа, яка торгує на платформі, насправді є тим, за кого себе видає, і що вона використовує дійсну авторизовану кредитну картку.

Хоча багато продавців використовують ручний процес для виконання цих перевірок, все більше користувачів бачить потребу в автоматизації та операційній ефективності.

OCR, машинне навчання та ШІ вважаються найкращими технологіями для цього [22].

Наприклад, у криптовалютній індустрії ідентифікація клієнтів перевіряється автоматично шляхом підтвердження автентичності їхніх кредитних карток і зіставлення інформації із документами, що посвідчують особу (наприклад, ім'я власника картки).

Автоматична перевірка віку за допомогою сканування кредитної картки.

У рамках процесу реєстрації клієнтів компанії можуть проводити перевірку ідентифікації, щоб переконатися, що вони залучають потрібних клієнтів для послуг із віковими обмеженнями, наприклад, вебсайти з алкогольними або тютюновими виробами та відеоігри з неприйнятним вмістом. Це можна зробити, вимагаючи від користувачів надати дані своєї кредитної картки та перевіряючи дані.

Підтвердження кредитної картки вимагає від власників кредитної картки довести, що вони володіють картою за допомогою таких методів, як Card Verification Value (CVV), Address Verification System (AVS) і 3D Secure. Це пов'язує право власності на особу, яка здійснює покупку в Інтернеті, з урахуванням того, що клієнт має бути старше 18 років, щоб подати заявку на кредитну картку [23].

Існує припущення, що переважна більшість власників кредитних карток є дорослими, це безперечно більш складна система перевірки віку, ніж просто просити користувачів заповнити свою дату народження у формі.

Підтвердження банківського рахунку (IBAN) за допомогою сканування дебетової картки.

Міжнародні номери банківських рахунків (IBAN) широко використовуються як у внутрішніх, так і в міжнародних платежах. Будучи досить довгим рядом цифр і чисел, IBAN часто є причиною помилки під час введення у форми вручну.

За допомогою програмного забезпечення компанії можуть підтвердити, що IBAN дійсно існує, та ідентифікувати індивідуальні банківські рахунки. Це допомагає зменшити кількість невдалих транзакцій через неправильно введені номери IBAN шляхом визначення структур даних і виконання перевірки формату, цифр і довжини.

Таким чином, компанії можуть перевірити IBAN і навіть надати докази володіння IBAN, зіставивши ім'я власника картки на банківській картці з документом, що посвідчує особу.

Автоматична анонімізація даних банківської картки.

Протягом багатьох років компанії могли зібрати тисячі або навіть мільйони документів, що містять конфіденційну інформацію, таку як дані банківських карток. Найчастіше люди не зовсім усвідомлюють, яка інформація там насправді та орієнтуються в цих даних не дуже ефективно.

За допомогою OCR API компанії можуть автоматично виявляти конкретні комбінації чисел у документах, наприклад, IBAN або номери кредитних карток, робити їх анонімними, видаляючи або виводячи чорними лініями ці конкретні рядки в документі або повністю видаляючи документи.

Також можливо анонімізувати дані з самого початку, автоматично виділяючи певні елементи після сканування банківської картки, наприклад, останні 3 цифри IBAN. Зазвичай, це виснажливе, дороге та обтяжливе завдання, але за допомогою програмного застосунку це стає досить просто [24].

1.3 Особливості реквізитів банківських карток

Кредитна картка – це щось більше, ніж прямокутний бланк із пластику та металу. Все відповідає суворому стандарту, навіть форма та зміст самої картки. Кредитна картка – це потужний інструмент, виданий фінансовою установою, наприклад, банком або кредитною спілкою, який дозволяє позичати кошти.

Фінансові установи використовують сторонню компанію, яка називається мережею кредитних карток, щоб полегшити зв'язок між платіжним терміналом і банком-емітентом. Таким чином, електронні перекази відбуваються швидше. Чотири основні мережі кредитних карток домінують на світових ринках: Visa, Mastercard, American Express і Discover [25].

Перша і найбільша мережа спочатку називалася «BankAmericard». Картка була заснована в 1958 році Банком Америки, щоб запропонувати кредитний продукт, орієнтований на споживачів, а не на бізнес. Але до середини 60-х років це змінилося, накопичивши великі прибутки. До 1974 року BankAmericard вийшов на міжнародні ринки, а в 1976 році став відомий як «Visa Inc». Невеликі банки відреагували на спробу Bank of America, прийнявши цю концепцію [26].

Mastercard, відомий у той час як «Interbank», з'явився в 1966 році. До 1968 року компанія стала глобальною. У 1979 році карта змінила назву і стала називатися Mastercard. Сьогодні це друга за величиною карткова мережа в світі.

Історія виникнення American Express почалася в 1850 році як невелика транспортно-експедиційна компанія. До 1950-х років вона різко зрісла і поширилася на фінансовий сектор. У 1958 році American Express випустила свою першу споживчу платіжну картку. Цікавим фактом є те, що ці примітивні картки були зроблені з паперу. Наразі American Express є третьою за величиною картковою мережею у світі [27].

Найменшою з чотирьох основних карткових мереж є Discover Card. Sears вперше випустив Discover Card у 1985 році як спосіб вийти на фінансовий ринок. Вона швидко зросла за короткий проміжок часу. Зараз нею користуються мільйони продавців у всьому світі.

Кожна з цих мереж мала одну мету: задовольнити потребу споживача в негайному доступі до коштів, не покладаючись на готівку. Сучасні кредитні картки мають заплутану та вишукану історію, більша частина якої спрямована на захист споживачів від шахрайства [28].

Передня сторона кредитної картки. На лицьовій стороні кредитної картки, зазвичай, розташовані такі об'єкти:

- брендінг банку;
- унікальний номер картки;
- ім'я власника картки;
- термін придатності;
- розумний чіп;
- логотип платіжної мережі.

Структура номера картки має певні особливості. Хоча це може здатися випадковим, кожна цифра кредитної картки стратегічно розташована та є важливою інформацією. Точні транзакції були б неможливі без певної послідовності чисел, точної форми та точного розміру, які відповідають суворим стандартам, які диктує ISO (Міжнародна організація зі стандартизації) і дотримується ANSI (Американський інститут національних стандартів) [29].

Ці стандарти дозволяють використовувати картки в усьому світі. Єдина відмінність між ними полягає в тому, що Visa, Mastercard і Discover завжди мають 16 цифр, тоді як American Express використовує 15-значний формат.

Перші 6 цифр кредитної картки представляють ідентифікаційний номер емітента (IIN), також відомий як ідентифікаційний номер банку (BIN). Ці цифри чітко ідентифікують фінансову установу, яка випустила картку. Перша цифра є ідентифікатором основної галузі (MII), її призначає Американська банківська асоціація. Кожна велика мережа кредитних карток має свій власний MII.

Картки American Express завжди починаються з цифри 3, точніше 34 або 37. Картки Visa починаються з цифри 4. Mastercards починаються з цифри 5. Картки Discover починаються з цифри 6.

Наступні 5 цифр PIN представляють конкретний банк-емітент. Ці цифри полегшують обмін інформацією для клірингу транзакції. PIN для кожної з чотирьох основних мереж:

- Visa використовує номери від 2 до 6 як BIN;
- Mastercard використовує цифри 2 і 3 або від 2 до 4, 5 або 6;
- American Express використовує цифри 3 і 4 для ідентифікації марки картки (наприклад, картка American Express Platinum, Delta Card тощо) [30].

Емітенти кредитних карток і мережі використовують математичні інструменти для боротьби з витоком даних та іншими шахрайськими діями. Алгоритм Луна або модуль 10 є одним із таких пристроїв.

Розроблений у 1960-х роках, він використовує ідентифікаційні цифри, такі як номери соціального страхування та номери кредитних карток, щоб визначити дійсність.

Кредитні картки призначені для миттєвого використання для платежів. Ось чому процес перевірки, який використовують банки, повинен негайно шифрувати та розшифровувати конфіденційні дані. Саме в цьому процесі задіяний алгоритм Луна. За його допомогою можна легко перевірити номери карток і підтвердити їх дійсність [31].

Алгоритм Луна простий у використанні. При додаванні номера чека до решти чисел на картці сума повинна дорівнювати 0. Якщо вноситься неправильне число під час онлайн-покупки, воно буде виявлено відразу, оскільки сума не буде 0.

Visa використовує цифру 13 як контрольну суму в більшості випадків, тоді як усі інші основні мережі використовують останню цифру.

Зворотня сторона кредитної картки. Після того як розглянуто інформацію, що міститься на передній стороні кредитної картки, можна перейти до зворотної сторони. Тут, зазвичай, знаходяться такі поля [32]:

- код безпеки (CVV);
- магнітна смуга;
- голограма;

- контактна інформація банку та телефон служби підтримки клієнтів;
- скринька для підписів;
- логотип карткової мережі;
- на звороті деяких кредитних карток також зазначено термін дії.

1.4 Аналіз літературних джерел щодо апробації результатів розпізнавання реквізитів банківських карток

Тематика комп'ютерного зору та автоматизації процесів є дуже популярною на сьогоднішній день. Проте, у мережі Інтернет не було знайдено великої кількості наукових джерел саме з теми автоматичного розпізнавання реквізитів банківських карток. Але, майже усюди було зазначено саме актуальність та перспективу розвитку функціоналу такого типу. Тож можна сказати, що ця тема ще активно розвивається, адже частка людей, що застосовує розпізнавання реквізитів банківських карток не є значною. Люди з кожним роком звикають до нових автоматизованих процесів та розуміють, що це зручно, безпечно та точно. Саме тому розпізнавання банківських карток поступово отримує популярність серед населення [33].

В цьому підрозділі було розглянуто деякі проаналізовані наукові статті, в яких описано методи ідентифікації реквізитів карток банків.

У статті [34] розглянуто швидкий розвиток технологій у світі та причини чому аутентифікація транзакцій не має відставати. Автор описав як за допомогою оптичного розпізнавання тексту можна автоматично заповнювати інформацію про кредитну картку, щоб полегшити використання та уникнути час на повторне введення тексту. Розглянуто визначення такого поняття як оптичне розпізнавання символів та розглянуто приклад із його застосування. Основними пунктами у статті були постановка проблеми, визначення вимог, визначення очікуваних результатів, опис літератури, процес, приклади фотографій банківських карток, результати дослідження. Основним недоліком

джерела можна вважати лише поверхневий опис процесу та відсутність конкретного приклада програми із зазначенням коду та інструментів розробки.

У статті [35] розкрито загальну тему локалізації та сегментації тексту на картинках та відео. Адже багато зображень, особливо ті, що використовуються для дизайну вебсторінок, а також відео містять видимий текст. Якби ці входження тексту можна було виявити, сегментувати та розпізнати автоматично, вони стали б цінним джерелом семантики високого рівня для індексування та пошуку. Автор описав можливості для ідентифікації текстових рядків за допомогою комплекснозначної багатосарової прямої мережі, навченої виявляти текст у фіксованому масштабі та положенні. Вихідні дані мережі в усіх масштабах і позиціях інтегровані в єдину карту помітності тексту, яка служить відправною точкою для текстових рядків-кандидатів. У випадку відео, ці рядки тексту-кандидата уточнюються шляхом використання тимчасової надлишковості тексту у відео. Рядки локалізованого тексту потім масштабуються до фіксованої висоти 100 пікселів і сегментуються на бінарне зображення з чорними символами на білому тлі. Для відео ж використовується тимчасова надлишковість для покращення ефективності сегментації. Вхідні зображення та відео можуть бути будь-якого розміру завдяки справжньому підходу з різною роздільною здатністю. Крім того, така система не лише здатна знаходити та сегментувати фрагменти тексту у великі двійкові зображення, але також здатна відстежувати кожен рядок тексту з точністю до субпікселя в усьому відео, таким чином створюючи один растровий текстовий малюнок для всіх екземплярів цього текстового рядка. У результаті сегментації тексту також можна використовувати для об'єктного кодування відео, яке підтримується MPEG-4. Дані із цього джерела були корисні у вивчені та розумінні загального процесу локалізації та подальшої обробки тексту. Недоліками статті є мала кількість таких візуальних матеріалів як скріншоти, графіки. Також відсутня демонстрація коду.

Стаття [36] містить інформацію про оцифрування даних з рахунків-фактур за допомогою OCR. Автор зазначив, що оптичне розпізнавання

символів є основним аспектом для перетворення відсканованих зображень та інших візуальних елементів у текст. Технологія комп'ютерного зору екстраполюється на систему для покращення тексту всередині оцифрованого зображення. Ці попередні налаштування зберігають інформацію про рахунок-фактуру та перетворюють її на конфігурації JSON і CSV. Ця модель може бути корисною у віщуванні на основі інженерії знань і якісного аналізу в найближчому майбутньому. Існуюча система містить вилучення даних і нічого більше. Для підвищення якості зображення в першу чергу застосовуються такі методи попередньої обробки зображення, як чорно-біле, інвертування, видалення шуму, відтінки сірого, густий шрифт і canny. З покращеним зображенням виконується більше процедур OpenCV. Для подальшої обробки використовуються три різні OCR: Keras OCR, Easy OCR і Tesseract OCR, з яких Tesseract OCR дає точний результат. Після початкових кроків небажані символи (/t, /n) очищаються, щоб отримати посилений текст як результат. Зрештою, у статті розглянуто приклад унікальної роботи, яка є високоточною у форматах JSON і CSV. Корисними були й зазначені приклади файлів JSON та CSV. Але, на жаль, стаття не містила досить скріншотів, що могло полегшити розуміння процесу.

Головною метою статті [37] є допомога у самостійному створенні зчитувача кредитних карток за допомогою бібліотеки OpenCV та машинного навчання, виконавши прийоми визначення номера кредитної картки, а також типу кредитної картки картки. Автор охопив такі теми як вступ до методу розпізнавання OCR, реалізація зчитувача кредитних карток на мові програмування Python, включаючи кожен із етапів розробки програми, приклади виводу, зразки результатів роботи розпізнавання та висновки роботи. Було зазначено, що оптичне розпізнавання символів часто використовується в машинному та глибокому навчанні. Такі програми включають ідентифікацію та зчитування кредитних карток і номеру картки. На прикладі було розглянуто роботу механізм розпізнавання застосовуючи функції з numpy та matplotlib разом із модулем OpenCV. Тип картки призначається за першою цифрою

номера картки. Щоб прочитати еталонне OCR-зображення, автор використав функцію `imread`. Еталонне зображення містить цифри 0-9 у шрифті OCR-A, які пізніше вийшло використати для виконання зіставлення. Попередня обробка зображення включає перетворення його на сіре зображення і потім визначення порогового значення інвертування зображення, щоб отримати бінарне інвертоване зображення. Відбувався процес пошука контурів, що присутні на попередньо обробленому зображенні, потім вони ця інформацію зберігалась. В результаті аналізу статті вийшло успішно ознайомитись із концепією розпізнавання банківських карток. Велика кількість коду програми із порядковим поясненням позитивно сприяли на розуміння процесу. Також, зручно було побачити візуальні матеріали, але, недоліком є мала кількість тестових прикладів карток.

У статті [38] описано теоретичну інформацію щодо розпізнавання банківських карток. Автор зазначив, що рішення для глибокого навчання, і всі види організацій, як-от технологічні гіганти, добре розвинені компанії та стартапи, тепер намагаються запровадити всі види сучасних організацій, як технологічні гіганти, так і добре розвинені компанії й стартапи, якимось включити глибоке навчання і машинне навчання у свій поточний робочий процес. Одним із таких важливих рішень, які набули досить великої популярності за останні кілька років, є саме OCR метод (рис. 1.3). Отримувати інформацію з цифрових документів легко, оскільки вони мають метадані, які можуть надати текстову інформацію. Але для відсканованих копій потрібне інше рішення, оскільки метадані тут не допомагають. Тут виникає потреба в глибокому навчанні, яке забезпечує рішення для вилучення текстової інформації із зображень. OCR став дуже популярним у наш час, цей спосіб застосовують в кількох галузях для швидшого читання текстових даних із зображень. У той час як такі рішення, як виявлення контурів, класифікація зображень, аналіз пов'язаних компонентів тощо, використовуються для документів, які мають порівнянний розмір тексту та шрифт, ідеальні умови освітлення, гарну якість зображення тощо, такі методи неефективні для нерегулярного неоднорідного тексту, який часто називають «дикий» текст.

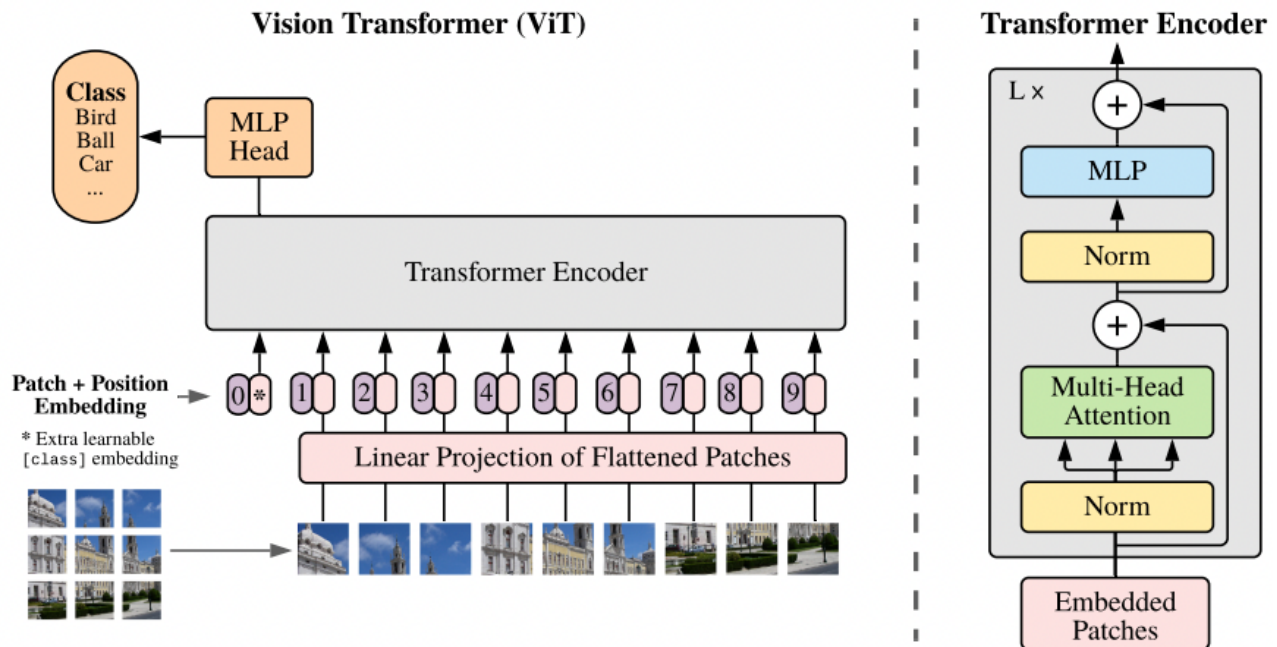


Рисунок 1.3 – Модель OCR на основі глибокого навчання

Цей текст може бути з номерного знака автомобіля, номера будинку, погано відсканованих документів (без попередньо визначених умов) тощо. Для цього використовуються рішення Deep Learning. Використання глибокого навчання для OCR є триетапним процесом, і ці кроки є попередня обробка, локалізація тексту, розпізнавання тексту. В статті описано що таке оптичне розпізнавання символів на основі глибокого навчання, різні варіанти його використання та, сценарії варіантів використання OCR. Автор зазначив актуальність тематики, адже технологія оптичного розпізнавання символів зараз переймає ручне введення даних і роботу з обробкою документів, можливо, це правильний час, щоб навчитися працювати з нею, щоб не почувалися остронь у світі глибокого навчання. Працюючи над такими варіантами використання, треба пам'ятати, що не можна отримати гарну модель за один раз, потрібно пробувати різні речі та вчитися на кожному кроці. На жаль, автор не показав ніякого практичного прикладу реалізації алгоритму OCR, а зазначив лише теоретичну базу та загальну схему його роботи. Корисним було використання зрозумілих діаграм та статистичних даних.

Автор статті [39] розглянув проблему, пов'язану з високою похибкою користувача при введенні номера банківської картки на мобільному терміналі. У своїй статті він запропонував метод розпізнавання банківської картки на основі мобільного терміналу. Метод в основному поділяється на три етапи: отримання та попередня обробка зображення банківської картки, сегментація цифрових символів номера банківської картки, розпізнавання цифрових символів. Частина попередньої обробки зображення в основному використовує різні алгоритми виявлення країв і морфологічні операції для обробки, включаючи корекцію проекції банківської картки та отримання області номера картки. Після отримання цифрової області номер картки виділяється, а номер сегментується. Після цього нейронна мережа використовується для навчання моделі реалізації остаточного розпізнавання чисел. Завдяки розробці програми на основі клієнта Android автором багато разів перевірено, що цей метод може краще ідентифікувати номер банківської картки.

Перевагою статті [39] є опис великої кількості тестових об'єктів, а саме цифр на картці у різних умовах (рис. 1.4). Приклади коду програми, на жаль, відсутні.

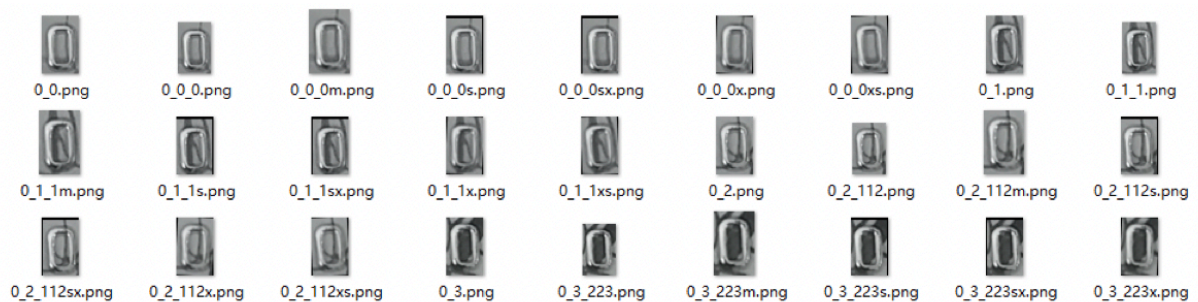


Рисунок 1.4 – Приклади тестових зображень цифри на банківській картці

Автори статті [40] дослідили та описали розроблений метод швидкого розпізнавання картки на основі MATLAB, який може ідентифікувати номер банківської картки автоматично. Завдяки обробці сірого кольору, обробці обрізання країв, обробці позиціонування номера картки, розпізнаванню номера

картки зі збігом шаблону та іншій обробці зображення банківської картки отримується точний і чіткий номер банківської картки, який має певну цінність для використання в електронній комерції та Інтернет. Він також надає деякі характеристики для обробки зображень і розпізнавання символів. Метод розпізнавання номера банківської картки на основі MATLAB може ефективно та точно ідентифікувати номер банківської картки, що зручно для щоденного використання людьми. В той же час, цей метод все ще має велику дослідницьку перспективу, а оптимізацію сегментаційного розпізнавання можна використовувати як дослідження поглибленого дослідження. Є можливість оптимізувати запропонований алгоритм, щоб покращити швидкість розпізнавання, а отже й отримати хороші експериментальні результати. Корисним є наявність математичних формул, що описують механізм обробки. Але деталей програмної реалізації не є достатньо щоб повноцінно зрозуміти алгоритм роботи даного запропонованого застунку.

У статті [41] описано метод автоматичного розпізнавання номера банківської картки. Автор зазначає, що з розвитком мобільного Інтернету мобільний платіж став одним із найпопулярніших способів оплати. Щоб підвищити ефективність роботи, зменшити витрати на оплату праці та покращити взаємодію з користувачем, інтелектуальна ідентифікація банківських карток широко використовується в мобільних платежах. Технологія оптичного розпізнавання символів (OCR) має проблеми з низьким рівнем розпізнавання при роботі з текстом банківської картки зі складним фоном. Таким чином, запропоновано метод ідентифікації номера банківської картки на основі глибинного навчання. Спочатку розширюється набір даних, потім алгоритми використовуються та оптимізуються для визначення номера картки. Деякі експериментальні результати показують, що метод має високу швидкість розпізнавання. У цій статті досліджується інтелектуальна ідентифікація номерів банківських карток. Для вирішення цієї проблеми запропоновано метод ідентифікації номера банківської картки на основі глибокого навчання. У запропонованому способі в для розпізнавання номера

банківської карти застосовано алгоритм, що створив автор на основі OCR. Запропонований метод дозволяє оптимізувати здатність виявлення та ефективність розпізнавання номера банківської картки. Результати випробувань свідчать про те, що запропонований метод може значно покращити здатність виявлення та розпізнавання номера картки для банківських карток зі складним і різноманітним фоном. Автор зазначає кілька прикладів тестування із різними фонами, проте не описує деталей реалізації, що могло б бути дуже корисним.

Зважаючи на складність ідентифікації номерів банківських карток у реальному середовищі, у дослідженні [42] пропонується метод ідентифікації номерів банківських карток на основі глибокого навчання. Метод поділяється на два етапи: визначення місцезнаходження номера банківської картки та ідентифікація номера картки після місцезнаходження. Модель даних нейронної мережі використовувалася для визначення розташування області номера картки, а алгоритм для обробки дасету. Нейронна мережа була використана для навчання моделі даних для визначення місцезнаходження номера банківської картки, після цього було завершено цифрове розпізнавання номера картки. У цьому методі використовується наскрізне позиціонування та розпізнавання номерної лінії банківської картки, яка може автоматично адаптуватися до нахилу зображення, і має хорошу адаптивність для ідентифікації номера банківської картки в природному середовищі. Це дослідження пропонує метод розпізнавання номерів банківських карток на основі глибокого навчання. Метод використовує наскрізний метод розпізнавання номерів банківських карток і не вимагає традиційних методів сегментації символів, корекції зображень та іншої обробки зображень. Обидва надруковані і тиснені банківські картки можна добре позиціонувати та розпізнавати, вони можуть автоматично адаптуватися до нахилу зображення та мають хорошу адаптивність до розпізнавання номерів банківських карток зі складним фоном у реальному середовищі. Недоліками статті є відсутність математичних формул, адже це суттєво впливає на розуміння описаного алгоритму розпізнавання.

У статті [43] автором визначено, що системи розпізнавання карток є репрезентативними програмами на основі комп'ютерного зору. Вони мають широкий спектр сценаріїв використання. Наприклад, їх можна використовувати для розпізнавання зображень, що містять візитні картки, особисті посвідчення особи, банківські картки тощо. Незважаючи на те, що системи розпізнавання вивчалися багато років, все ще важко розпізнати картки на зображеннях, зроблених камерою звичайними пристроями, наприклад, мобільними телефонами. Різноманітність точок зору та складний фон на зображеннях ускладнюють завдання розпізнавання. Існуючі системи, що використовують традиційні схеми обробки зображень, не стійкі до різноманітних умов і неефективні в роботі з природними зображеннями, наприклад, зробленими мобільними телефонами. Щоб вирішити цю проблему, автор пропонує нову структуру для розпізнавання карток із застосуванням підходу на основі нейтральної мережі. Система локалізує передній план зображення за допомогою повністю згорткової мережі. Система локалізує кути області картки та використовує перспективну трансформацію, щоб зменшити ефекти спотворення. Рядки тексту в області картки виявляються та розпізнаються за допомогою згорткової нейтральної мережі та довгострокової короткострокової пам'яті. Щоб оцінити запропоновану схему, у статті продемонстровано великий набір даних, який містить 4065 зображень у різних сценаріях зйомки. Результати експерименту свідчать про ефективність запропонованого рішення. Зокрема, воно здатне досягти точності 90,62% у наскрізному тестуванні, перевершуючи сучасні технології. В статті детально описано переваги нового методу, проте, не зазначено програмного коду та математичних формул.

Ще один спосіб розпізнавання реквізитів банківської картки за допомогою нейронної мережі зазначили автори статті [44]. Зазначено, що через багато факторів перешкод під час фотографування банківської картки, таких як невизначеність кута зйомки, складність умов освітлення та різноманітність фону банківської картки, існують великі проблеми для алгоритму цифрового розпізнавання банківської картки на основі природньої сцени зйомки. Тому

автори пропонують розпізнавання банківських карток на основі згорткової нейронної мережі. По-перше, цифрова область цільової банківської картки отримується шляхом виконання ряду алгоритмів обробки зображень, таких як корекція проекції, виявлення країв і операція морфології. По-друге, нейронна мережа згортки навчається за допомогою доповненого набору даних для отримання вказаної вище цільової цифрової області для розпізнавання ковзного вікна. Потім початкова послідовність номерів банківської картки виводиться для створення цифрового графіка. Запропоновано алгоритм оптимізації згладжування, який вводить наведений вище початковий графік номера банківської картки та оптимізує його. Потім цифрова послідовність розбивається на окремі числа і виводиться кінцевий результат. Результати експерименту показують, що алгоритм значно підвищує точність цифрового розпізнавання та сегментації банківських карток. У той же час він все ще має хорошу надійність для тих банківських карток із більш складними зображеннями. Важливо зазначити, що опис алгоритму у статті є гарно структурованим, проте відсутня достатня кількість тестувань, що ставить під питання остаточні результати випробувань.

У статті [45] описано ще один варіант розпізнавання даних кредитної картки. Автор звертає увагу на те, що зі швидким розвитком технологій та поширенням Інтернету онлайн платіж став одним із найпопулярніших способів оплати. При цьому в житті багато сценаріїв будуть передбачати прив'язку та ідентифікацію банківських карток. Багато існуючих технологій розпізнавання мають деякі недоліки, в основному це недостатню точність або швидкість. Запропонована система використовує метод наскрізної ідентифікації глибокого навчання для ідентифікації номерів банківських карток. Наскрізне означає, що входом є необроблені дані, а виходом є кінцевий результат, що дозволяє нейронній мережі автоматично зменшувати розміри та вивчати функції. В підсумку система має графічний інтерфейс, щоб забезпечити зручний інтерактивний інтерфейс користувача. Проте, в статті немає ніяких

математичних обґрунтувань та формул, що було б корисно для повноцінного розуміння процесу.

Виконавши аналіз літературних джерел щодо актуальних підходів розпізнавання реквізитів банківських карток можна зробити висновок, що ця тема зараз на часі. Проте, матеріалів українською мовою мало, що свідчить про те, що в Україні такі технології ще на початковому етапі поширення. Матеріалів англійською мовою було знайдено значно більше, вони містили досить гарні приклади систем, теоретичні відомості та результати досліджень. Цікавим фактом виявилось те, що найбільше всього статей в Інтернеті з даної тематики саме китайською мовою, що характеризує величезний розвиток та популярність систем автоматичного розпізнавання реквізитів банківських карток в Китаї.

Не всі статті знаходяться у відкритому доступі. Багато джерел доступні лише частково, а для їх повноцінного перегляду необхідно заплатити. Важливо зазначити, що більшість джерел інформації спрямовані лише на теорію або практику. Велика частка ресурсів наповнена лише теоретично, що але це підтверджує ефективність методів у вигляді практичної реалізації. І навпаки, окремі статті містять готовий запрограмований алгоритм, проте автори не зазначають теоретичні відомості щодо нього. Через це існують окремі проблеми, якщо потрібно повторити реалізацію самостійно.

У підсумку можна сказати, що є сенс шукати нові шляхи щодо вирішення проблем у наявних літературних джерелах.

1.5 Постановка задачі дослідження

Сучасні інформаційні технології мають великий попит у суспільстві. Значну кількість повсякденних задач, що виконують люди, все більше перекладають на комп'ютеризовані системи, автоматизуючи їх. Вже сьогодні вони оточують людину у багатьох сферах життя, суттєво полегшуючи

виконання багатьох процесів. Кінцевими цілями створення програмних застосунків із розпізнавання даних є спрощення людського життя, автоматизації процесів, покращення якості роботи.

Об'єктом дослідження є датасет різноракурсних зображень банківських карток.

Метою дослідження є аналіз та порівняння методів розпізнавання реквізитів банківських карток.

Виходячи із мети дослідження, необхідно вирішити такі задачі:

- проаналізувати сучасні системи розпізнавання реквізитів банківських карток;
- визначити класифікацію та провести аналіз існуючих методів розпізнавання реквізитів банківських карток;
- проаналізувати особливості реквізитів банківських карток;
- проаналізувати літературні джерела щодо апробації результатів розпізнавання реквізитів банківських карток;
- визначити механізм оброблення зображення банківських карток;
- проаналізувати принципи розпізнавання складових банківських карток;
- вивчити методику розпізнавання реквізитів банківських карток;
- вибрати інструментальний засіб для реалізації методів розпізнавання реквізитів банківських карток;
- визначити етапи програмної реалізації методів розпізнавання реквізитів банківських карток;
- провести тестування розробленого застосунку та здійснити порівняння методів розпізнавання реквізитів банківських карток;
- запропонувати перспективи подальшої роботи.

2 ОСОБЛИВОСТІ РОЗРОБЛЕННЯ МЕТОДІВ РОЗПІЗНАВАННЯ РЕКВІЗИТІВ БАНКІВСЬКИХ КАРТОК

2.1 Механізм оброблення зображення банківських карток

Заповнення полей кредитної картки є одним із найбільш виснажливих кроків під час використання онлайн-платежей. Використовуючи автоматизовану систему, користувач може просто піднести картку до камери, і форма буде заповнена автоматично. Однак, існує занепокоєння користувачів, які зберігають дані своїх карток на вебсайтах. Вебсайт може бути зламаний, тоді дані картки можуть бути отримані зловмисникам. Якщо користувачі хочуть зберегти дані своєї картки після першого «зчитування», вони можуть скористатися існуючою технологією, за якою дані картки шифруються та зберігаються [46].

Для того, щоб виконати розпізнавання даних картки необхідно локалізувати картку на зображенні, знайти інформаційні поля та виконати сегментацію на символи. Кожний із перерахованих пунктів з формальної точки зору є самостійним завданням розпізнавання. Якщо для навчання нейронних мереж існують підходи та інструменти, що зарекомендували себе, то в завданнях орієнтації та сегментації щоразу потрібен індивідуальний підхід.

Саме оптичне розпізнавання символів є чи не найголовнішим кроком у всій системі із розпізнавання реквізитів банківської картки. Блок-схема детальних етапів роботи OCR методу показана на рисунку 2.1 [47].

У процесі розпізнавання даних картки першим кроком є надання зображення або PDF-файлу дебетової чи кредитної картки до прикладного програмного інтерфейсу (Application Programming Interface або API). Це можна зробити за допомогою мобільного застосунку, просто зробивши знімок на смартфон (рис. 2.2).

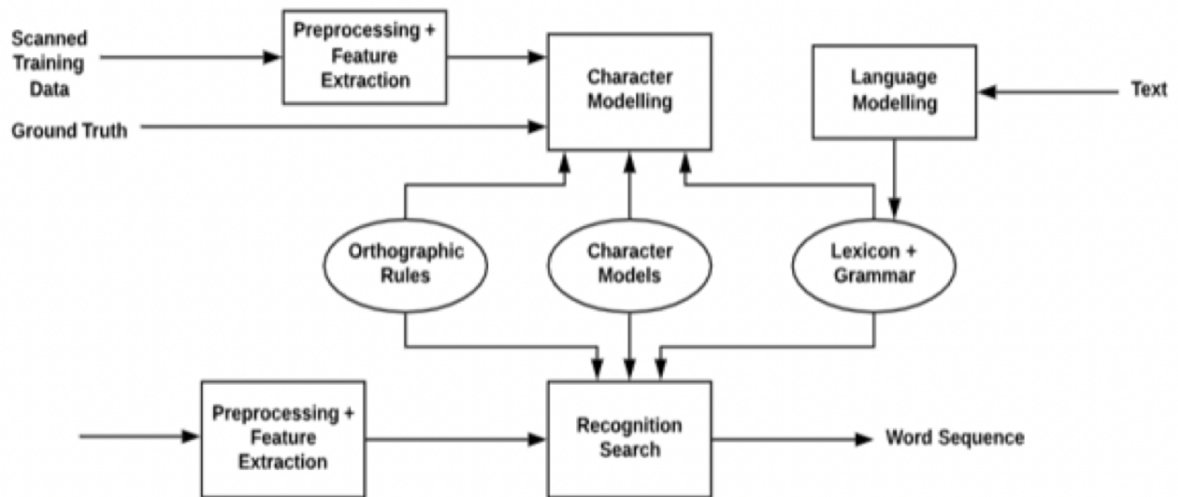


Рисунок 2.1 – Опис роботи методу розпізнавання OCR

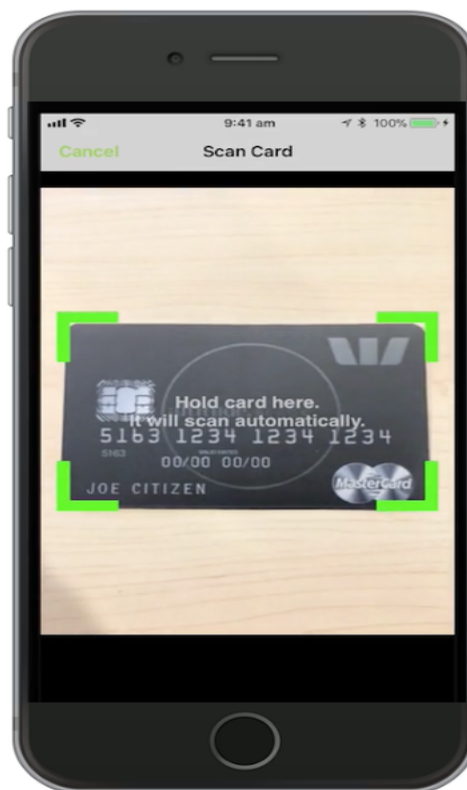


Рисунок 2.2 – Процес сканування картки смартфоном

Якщо у програмі ще немає функції камери, то можна використовувати готове рішення, наприклад, від Klipra Camera SDK може допомогти зі скануванням банківських карток [48]. Це повний набір для сканування, який має розпізнавання квадратів, кадрування, корекцію перспективи та визначення якості.

Коли зображення або PDF-файл отримано, воно оптимізується та перетворюється на необроблений текстовий файл. На цьому етапі весь текст картки витягується, але він ще не структурований (рис. 2.3) [49].

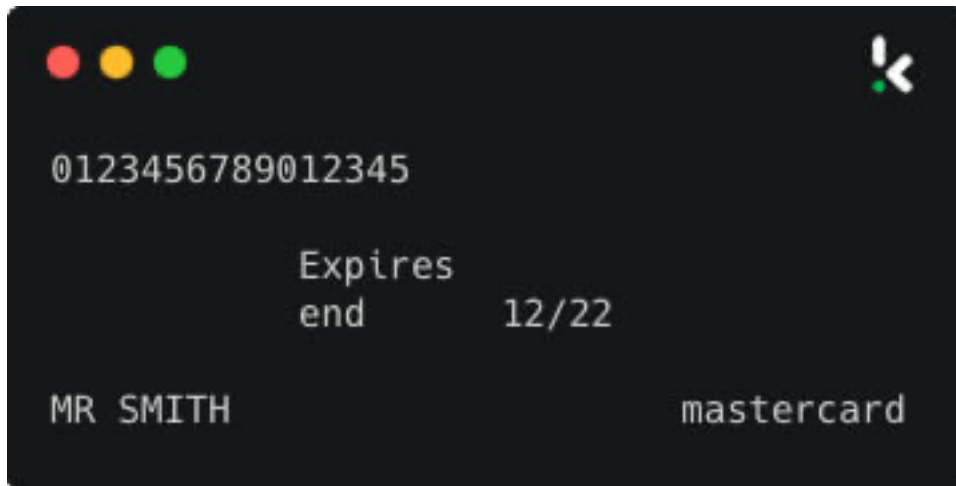


Рисунок 2.3 – Необроблений текстовий файл

Далі аналізатор бере текстовий файл і перетворює його на структурований формат JSON. Потім JSON повертається як результат API.

Видобуті дані можна використовувати відповідно до власних потреб, наприклад, передати їх у базу даних для подальшої обробки.

2.2 Принципи розпізнавання складових банківських карток

Очікуваним результатом системи з розпізнавання реквізитів банківської картки є те, що вона виявляє зображення картки та успішно розпізнає 16-значний номер картки на лицьовій стороні.

Система розпізнає тип платіжної системи картки за першими 4 цифрами номера картки. Виявлене зображення може мати мінімальну якість, щоб символи розпізнавались правильно. Після розпізнавання номера картки та терміну дії відбувається перехід до сторінки оформлення онлайн-оплати.

Коли зображення картки сканується, вона, зазвичай, зберігається як растровий файл у форматі TIF. Коли зображення відображається на екрані, його можна прочитати, але для комп'ютера це лише низка чорних і білих точок. Комп'ютер не розпізнає жодних «слів» на зображенні. Ця частина обробки виконується OCR. OCR розглядає кожен рядок зображення та намагається визначити, чи позначають чорні та білі точки певну букву чи цифру [50].

Щоб описати принцип роботи типової системи із автоматичного розпізнавання реквізитів банківських карток можна виділити такі кроки, що виконує користувач для виконання своєї онлайн-оплати (рис. 2.4) [51].



Рисунок 2.4 – Послідовність кроків користувача системи автоматичного розпізнавання реквізитів банківської картки

Для того, щоб застосувати систему із розпізнавання картки, користувач спочатку заходить на ресурс, де буде відбуватися проведення онлайн-оплати. Таким ресурсом може бути, наприклад, вебсайт або мобільний застосунок. Після того, як користувач створив замовлення (чи іншу подібну операцію, яку необхідно оплатити), він починає взаємодію саме із системою розпізнавання картки, що полегшує та прискорює процес сплати. Фізична картка сканується за допомогою вбудованої камери пристрою, а далі відправляється на процес обробки. Альтернативним варіантом також може бути завантаження вже готової фотографії картки, а не динамічний процес фотографування.

Після отримання зображення програма виконує обробку зображення. Цей процес включає в себе перевірку розміття картинки, виявлення прямокутника картки, та, найголовніше, застосування методу OCR, що виявляє реквізити:

- номер картки;
- строк дії картки;
- власника картки.

Швидкість сканування визначатиметься якістю сканерів, кольором, якістю паперу, чистотою, вагою, правильним налаштуванням системи OCR.

CVV код, що знаходиться на задній стороні картки користувача, зазвичай, вводить вручну. Адже вся необхідна інформація із задньої сторони картки це лише тризначне число, тому немає необхідності сканувати цю частину. Також, однієї із основних причин чому це, зазвичай, не відбувається це наявність динамічного CVV коду, який змінюється кожен фіксований проміжок часу (наприклад, раз на шістдесят хвилин). Таким чином, при втраті фізичної банківської картки шахраї не матимуть доступу для всієї необхідної інформації для оплати карткою. Зазвичай, користувач може дізнатись актуальний CVV код у застосунку банку або за допомогою SMS повідомлення, що приходить на мобільний телефон.

Після того, як визначено реквізити лицьової сторони картки та введено код безпеки, система із розпізнавання реквізитів банківської картки завершує виконання своєї задачі.

Далі, зазвичай, користувач підтверджує оплату за допомогою біометричних даних (сканування обличчя, відбитка пальця чи сітківки). Прикладом такої системи може бути Face ID або Touch ID, що застосовується на популярних смартфонах Apple iPhone (рис. 2.5) [52].



Рисунок 2.5 – Процес сканування обличчя за допомогою Face ID на смартфоні Apple iPhone

Щойно біометричне сканування успішно пройдено, настає етап обробки оплати. Цей процес виконує банк клієнта, платіжна система, система отримувач коштів. Якщо цей крок успішно пройдений, то можна вважати, що весь процес оплати пройдений та успішно завершений.

2.3 Методика розпізнавання реквізитів банківських карток

Виявлення картки на цифровому зображенні є першим кроком в алгоритмі розпізнавання реквізитів банківських карток. Далі відбувається пошук поля із інформацією, а потім проводиться процес сегментації отриманих даних на символи. Кожний крок можна назвати окремим завданням із розпізнавання реквізитів банківської картки.

Як перший етап необхідно знайти координати кутів картки. Оскільки геометричні характеристики картки нам відомі (картки виконані відповідно до стандарту ISO 781), то для визначення чотирикутника картки використовується один із існуючих алгоритмів [53].

При відомому чотирикутнику неважко обчислити та застосувати до зображення проєктивне перетворення, що призводить зображення карти до ортогонального вигляду з фіксованою роздільною здатністю. Таке виправлене зображення приходиться на вхід до таких етапів – орієнтації та розпізнавання конкретних інформаційних полів.

З погляду архітектури, розпізнавання трьох цільових полів складається з таких частин:

- попередня фільтрація зображення (з метою виділення реквізитів на фоні, який, зазвичай, не є однотонним);
- пошук зони (рядків) цільового інформаційного поля;
- сегментація знайденого рядка на «коробки символів».

Розпізнавання знайдених «коробок символів» відбувається за допомогою штучної нейронної мережі. Важливу роль відіграє застосування післяобробки (виявлення можливих помилок розпізнавання, застосування словників імен та прізвищ, перевірка дати тощо).

Кроки розпізнавання різних даних картки одні й ті ж самі, але складність розпізнавання відрізняється.

Найлегше розпізнати номер картки з таких причин:

- номер картки містить лише цифри;
- формат номера суворо визначено за типом платіжної картки;
- геометричне положення номера приблизно однакове;
- алгоритм Луна, що дозволяє перевірити правильність розпізнавання номера [54].

Складніше розпізнати інші поля, такі як термін дії або ім'я власника картки. Результатом роботи алгоритму по визначенню терміну дії картки є 4 десяткові цифри: по дві на місяць та рік терміну закінчення дії.

Вважається, що алгоритм видав правильну відповідь, якщо отримані 4 цифри збігаються з зображеними на карті. Символ, що їх поділяє, не враховується і може бути будь-яким. Відмова від розпізнавання сприймається як неправильна відповідь.

Процес працює так: OCR-сканер сканує фізичний об'єкт, наприклад, документ або кредитну картку, програмне забезпечення перетворює його на чорно-біле зображення. Потім програмне забезпечення аналізує зображення, кодує світліші ділянки, як фон, а темніші – як символи, що потрібно розпізнати [55].

На останньому етапі програмне забезпечення додатково обробляє темні області для розпізнавання літер і цифр. Воно ідентифікує ці символи за допомогою одного з двох різних типів алгоритмів – розпізнавання образів або виявлення ознак.

Отже, щойно документ сканується, він піддається попередній обробці. Зображення спочатку перетворюється на зображення в градаціях сірого, а потім на двійкове зображення. Цей процес перетворення називається оцифруванням.

Під час попередньої обробки зображення виконується операція згортання за допомогою фільтра Лапласа. Дисперсія цієї операції використовується для визначення розмитості зображення.

Для якомога найкращого результату потрібно отримати навчальні дані приблизно від 80 до 100 однаково розмитих і нерозмитих зображень та знайти граничне значення дисперсії за допомогою SVM (машина станів вектору) або логістичної регресії. Це значення використовується як межа, щоб визначити, чи є зображення розмитим чи ні.

На початковому етапі вручну визначається на основі приблизно 5-10 зображень, якою має бути дисперсія, щоб узгодити різкість зображення.

Зображення перетворюється на зображення у відтінках сірого за допомогою функцій із бібліотеки OpenCV2.

Ще одна технологія, яка застосовується – це використання фільтра `sanny` після розмиття за Гаусом. Зображення подаються безпосередньо для виконання оптичного розпізнавання символів (OCR-метод).

У попередньо обробленому зображенні виконується пошук, щоб знайти кожен окремий символ на зображенні. Ці розпізнані символи використовуються для порівняння зі збереженими даними.

Виявлені символи порівнюються зі значеннями інтерпретованого зображення за усіма шаблонами, попередньо завантаженими в автоматизовану систему. Після завершення порівняння розпізнані символи відображаються в текстовому форматі.

Рядок, що повертає метод OCR, додатково має оброблятися шляхом видалення всіх зайвих символів, крім:

- чисел;
- «/»;
- «\n»;
- « » (пробіл).

Щоб це виконати у програмі можна застосувати регулярний вираз, що має такий вигляд:

Лістинг 2.1 Реалізація регулярного виразу відкриття сторінки:

```
reg=r"(\d)|(/)|(\n)|(")
```

Щоб виявити текст i , що важливіше в цьому випадку, числа на зображенні, використовуються методи попередньої обробки.

По-перше, застосовується фільтр Лапласа, який використовується для визначення країв і різкості зображення. Дисперсія зображення повідомляє, чи зображення розмите, а також усуває шум із зображення.

Лапласіан $L(x,y)$ зображення зі значеннями інтенсивності пікселів $I(x,y)$ визначається як:

$$L(x,y) = \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2}, \quad (2.1)$$

де L – значення Лапласіана;

x – значення першого пікселя;

y – значення другого пікселя;

I – значення інтенсивності пікселів.

Обчислення відбувається за допомогою фільтра згортки.

Ядра, які найчастіше використовуються для цієї операції, наведені на рисунку 2.6 [56].

0	-1	0	-1	-1	-1
-1	4	-1	-1	8	-1
0	-1	0	-1	-1	-1

Рисунок 2.6 – Ядра фільтра Лапласа

3 РЕАЛІЗАЦІЯ ТА ПОРІВНЯННЯ МЕТОДІВ РОЗПІЗНАВАННЯ РЕКВІЗИТІВ БАНКІВСЬКИХ КАРТОК

3.1 Вибір інструментального засобу для реалізації методів розпізнавання реквізитів банківських карток

Для реалізації методів розпізнавання реквізитів банківських карток було обрано два способи: за допомогою бібліотеки OpenCV на мові Python використовуючи дистрибутив Anaconda та за допомогою засобів мови Swift, розробленою компанією Apple в середі розробки Xcode. Причиною такого вибору є те, що бібліотека OpenCV та мова Python є дуже популярним способом реалізації для будь-яких задач комп'ютерного зору. Бібліотека дуже широка, має безліч корисних функцій, а використовуючи Python можна швидко та зручно виконати реалізацію. В той же час, компанія Apple в останні роки дивує своїми розумними та шоковано якісними рішеннями щодо розпізнавання об'єктів. Цими застосунками користується мільйони людей по всьому світу, та рішення Apple стають все більш популярними на ринку. Ці можливості реалізовані за допомогою розробки методів єдиної мови Swift, яка використовується для створення застосунків компанії. Саме тому цікаво порівняти спроможності двох видів методів видів у межах теми розпізнавання реквізитів банківських карток.

Anaconda – це безкоштовний дистрибутив мов програмування Python та R з відкритим кодом. Дистрибутив постачається з інтерпретатором Python та різними пакетами, пов'язаними з машинним навчанням та наукою даних.

Anaconda складається із:

- системи з відкритим кодом та системи управління середовищем під назвою Conda, яка полегшує встановлення та оновлення пакетів та створення або завантаження середовищ;
- бібліотеки машинного навчання, такі як TensorFlow, scikit-learn та Theano;

- бібліотеки наукових даних, такі як pandas, NumPy та Dask;
- бібліотеки візуалізації, такі як Bokeh, Datashader, matplotlib та Holoviews;

- Jupyter Notebook, яка поєднує код, візуалізацію та текст.

Jupyter Notebook розширює консольний підхід до інтерактивних обчислень в якісно новому напрямку, надаючи вебзастосунок, що підходить для запису всього обчислювального процесу: розробки, документування та виконання коду, а також передачі результатів. Jupyter Notebook поєднує два компоненти.

Програма має простий інтерфейс (рис. 3.1) [57], що легко зрозуміти розробнику.

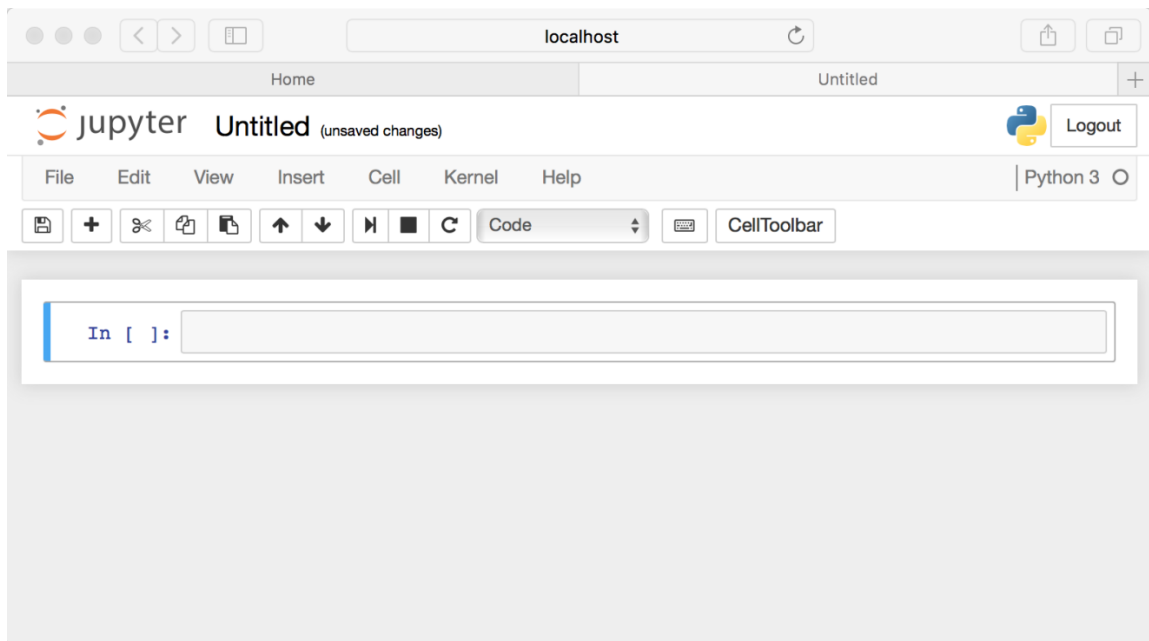


Рисунок 3.1 – Інтерфейс Jupyter Notebook

Python – це інтерпретована об’єктно-орієнтована мова програмування високого рівня з динамічною семантикою. Її високорівневі вбудовані структури даних в поєднанні з динамічною типізацією і динамічної прив’язкою роблять її дуже привабливим для швидкої розробки застосунків, а також для використання в якості мови сценаріїв або сполучного мови для з’єднання існуючих компонентів.

У сферах аналізу даних і машинного навчання Python зараз безсумнівно поза конкуренцією. Складний аналіз даних став однією з найшвидших галузей IT та одним із найкращих випадків використання Python. Переважна більшість бібліотек, що використовуються для науки про дані або машинного навчання, мають інтерфейси Python, що робить мову найпопулярнішим командним інтерфейсом високого рівня для бібліотек машинного навчання та інших числових методів. Основними перевагами мови Python вважають простоту у вивченні та використанні, широке застосування, кросплатформеність, велику кількість бібліотек, а також обширну кількість інформацію інтернет ресурсів.

OpenCV – це величезна бібліотека з відкритим кодом для комп'ютерного зору, машинного навчання та обробки зображень, і зараз вона відіграє важливу роль у роботі в режимі реального часу, що дуже важливо в сучасних системах. Використовуючи його, можна обробляти зображення та відео, щоб ідентифікувати предмети, обличчя або навіть почерк людини. При інтеграції з різними бібліотеками, такими як NumPy, Python здатний обробляти структуру масиву OpenCV для аналізу. Для ідентифікації візерунка зображення та різних його функцій ми використовуємо векторний простір і виконуємо математичні операції над цими ознаками [58].

Swift – потужна та інтуїтивно зрозуміла мова програмування для iOS, iPadOS, macOS, tvOS і watchOS. Написання коду Swift інтерактивне та просте, синтаксис стислий, але виразний, а Swift містить сучасні функції, які подобаються розробникам. Код Swift є безпечним за своєю конструкцією та створює програмне забезпечення, яке виконується миттєво.

Swift є результатом останніх досліджень мов програмування в поєднанні з досвідом створення платформ Apple. Іменовані параметри виражені в чистому синтаксисі, що робить API у Swift ще легшими для читання та обслуговування. Виведені типи роблять код «чистішим» і менш схильним до помилок, тоді як модулі усувають заголовки та надають простори імен. Щоб найкраще підтримувати міжнародні мови та емодзі, рядки є правильними для Unicode та використовують кодування на основі стандарту UTF-8 для оптимізації

продуктивності для широкого спектру випадків використання. Пам'ять управляється автоматично за допомогою жорсткого, детермінованого підрахунку посилянь, що зводить використання пам'яті до мінімуму без накладних витрат.

Swift має багато функцій, щоб зробити код розробника більш виразним:

- generics, які є потужними та простими у використанні;
- розширення протоколу, які спрощують написання загального коду;
- першокласні функції та легкий синтаксис закриття;
- швидка та коротка ітерація в діапазоні чи колекції;
- кортежі та множинні значення, що повертаються;
- структури, які підтримують методи, розширення та протоколи;
- enum може мати корисне навантаження та підтримувати відповідність шаблону;
- патерни функціонального програмування (наприклад карта та фільтр);
- вбудована обробка помилок за допомогою try, catch та throw.

Swift розробляється відкрито на Swift.org із вихідним кодом, програмою відстеження помилок та регулярними збірками розробки, доступними для всіх. Зараз мова є кросплатформенною, адже зараз Swift підтримує всі платформи Apple та Linux, а члени спільноти активно працюють над перенесенням на додаткові нові платформи. Ця широка спільнота розробників, як всередині Apple, так і сотні сторонніх учасників, працюють разом, щоб зробити Swift краще. Існує ширший спектр блогів, подкастів, конференцій і зустрічей, де розробники в спільноті діляться своїм досвідом щодо реалізації величезного потенціалу Swift.

Програмне середовище Xcode 14 містить усе необхідне для розробки, тестування та поширення програм на всіх платформах Apple (рис. 3.2) [59].

Проста та потужністю мови Swift і візуалізації SwiftUI за дозволяє створювати нові багатоплатформні застосунки, використовувати функції редактора та проводити тестування та розгортання з Xcode Cloud на TestFlight і App Store [60].

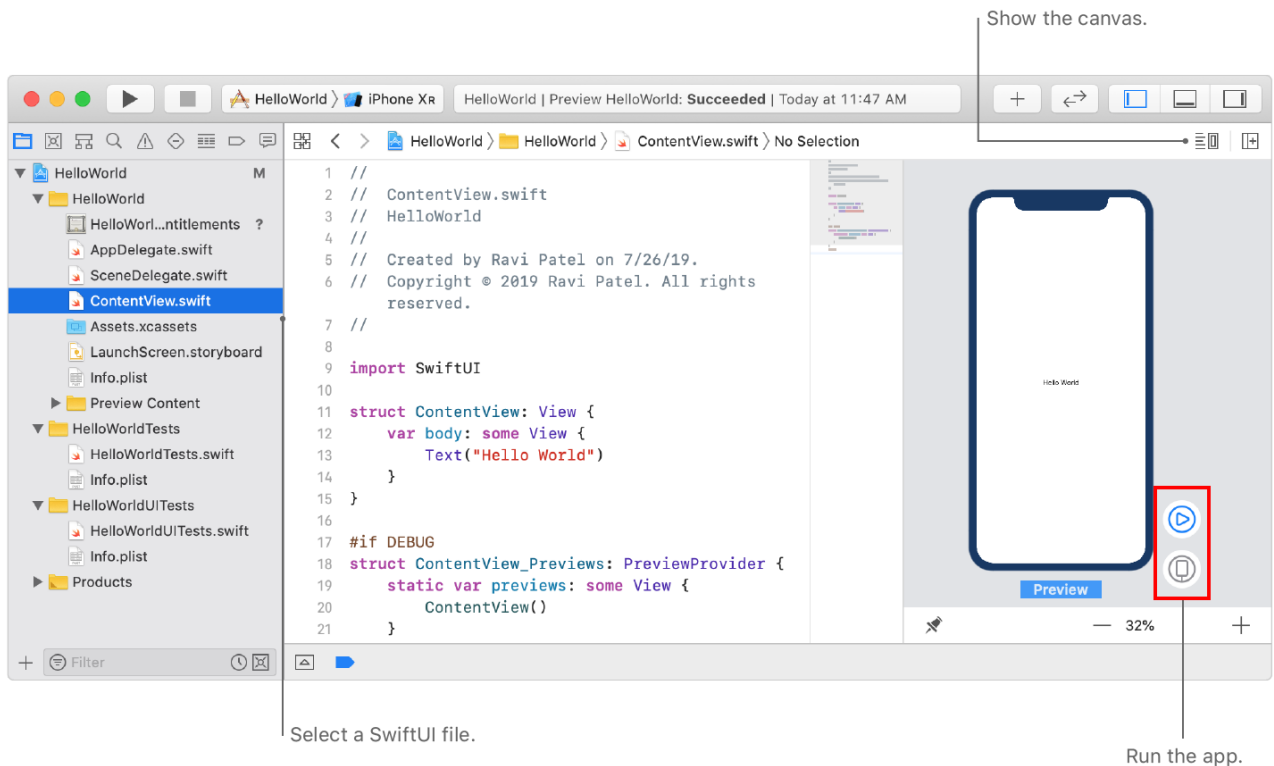


Рисунок 3.2 – Інтерфейс Xcode

Swift, SwiftUI та Xcode 14 працюють як одне ціле. Попередній перегляд SwiftUI стає миттєво інтерактивним, а варіанти інтерфейсу користувача, мають варіанти світлого і темного вигляду, що покращує сприйняття коду. Новий шаблон для інструментів полегшує налагодження та оптимізацію використання розподілених акторів та інших функцій паралелізму Swift. За допомогою плагінів збірки та командного пакета можна дуже просто і швидко налаштувати Xcode та процес збирання [61].

Xcode дає можливість створювати єдиний інтерфейс SwiftUI для використання в iOS, iPadOS, macOS і tvOS. Такий код легше підтримувати, і його можна налаштувати, щоб скористатися перевагами унікальних можливостей кожної платформи.

Завдяки переробленому досвіду роботи з піктограмами програм у каталогах активів ви можете надати один розмір піктограми для всіх платформ, а Xcode згенерує все інше.

Численні покращення Xcode дозволяють зручну роботу із кодом, адже регулярні вирази в Swift інтегровані з підсвічуванням синтаксису, операціями рефакторингу тощо [62].

Корисною функцією також є Xcode Cloud. Це – це служба безперервної інтеграції, що вбудована в Xcode і розроблена спеціально для розробників Apple. Вона надає можливість створювати свою програму всього за кілька хвилин, відстежувати статус збірки та звіти та автоматично розповсюджувати її серед тестувальників і користувачів. Організатор відгуків допомагає переглядати показники, збої та відгуки користувачів, а також містить знімки екрана.

3.2 Етапи програмної реалізації методів розпізнавання реквізитів банківських карток

Першим кроком у процесі програмної реалізації є підготовка датасету зображень карток, що буде використовуватись для спроб розпізнавання карток застосунками. Отже, буде відібрано 15 тестових зображень банківських карток у форматі PNG (рис. 3.3). Таким чином можна буде проводити етап тестування та аналізувати роботу методів.



Рисунок 3.3 – Приклад зображення із датасету карток

Далі, коли в наявності є датасет зображень потрібно саме розробити методи та порівняти їх роботу. Першою реалізовано метод із застосуванням OpenCV та Python, провівши нижче описані етапи.

Імпорт модулів. В цьому для роботи потрібно провести імпорт бібліотек withnumpy та matplotlib разом із модулем OpenCV.

Призначити тип картки. Тип картки призначається за першою цифрою номера картки. Тому створюється змінна, що буде містити в собі варіації кожної із платіжних систем. Це треба для ідентифікації типу картки на наступних кроках. Отже, відповідність буде наступна:

- цифра 3 для American Express;
- цифра 4 для Visa;
- цифра 5 для MasterCard;
- цифра 6 для Discover Card [63].

Завантаження та попередня обробка еталонного зображення. Щоб прочитати еталонне OCR-зображення, використовується функція imread. Еталонне зображення містить цифри від 0 до 9 у шрифті OCR-A, які пізніше можна використати для виконання зіставлення пізніше в алгоритмі.

Попередня обробка зображення включає перетворення його на сіре зображення, а потім визначення порогового значення та інвертування зображення, щоб отримати бінарне інвертоване зображення.

Розпізнавання контурів. На цьому кроці відбувається пошук контурів, що присутні на попередньо обробленому зображенні, а потім зберігається повернута інформацію про контури. Далі, сортуються контури зліва направо, а також ініціалізується словник, цифри, які відображають ім'я цифри в області інтересу.

Створення обмежувальних рамок навколо цифр. Тепер на цьому кроці відбувається проходження контурами зображення, що були отримані на попередньому кроці, де кожне значення містить цифру або число разом із інформацією про контур. Далі, обчислюється обмежувальна рамка для кожного контура та зберігається координати (x, y) разом із обчисленою висотою та шириною рамки.

Завантаження та попередня обробка зображення кредитної картки. На цьому кроці завантажується фотографію кредитної картки, а потім змінюється її ширина до 300 пікселів, щоб зберегти співвідношення сторін.

Після цього кроку виконується перетворення зображення в градації сірого. А далі, виконуються операції над сірим зображенням.

Наступним етапом є обчислення градієнта Шарра та збереження результату як `gradX`. Обчислюємо абсолютне значення збереженого масиву `gradX`. Потрібно намагатись масштабувати всі значення в діапазоні від 0 до 255. Тепер ця нормалізація значень відбувається шляхом обчислення мінімального та максимального значення `gradX` і формування рівняння для досягнення мінімально-максимальної нормалізації. В кінці цього кроку, знаходяться контури та зберігаються у списку, ініціалізується список для розміщення груп цифр.

Далі, групи сортуються зліва направо та створюється список для цифр кредитної картки.

Тепер, коли є інформація про те, де знаходиться кожна група з чотирьох цифр, можна проглянути чотири відсортовані групи та визначити цифри в них. Цикл також включає порогове значення, виявлення контурів і відповідність до шаблону.

Відображення остаточних результатів. Останній фрагмент програми просто проводить відображення остаточного типу картки, визначений номер картки та застосоване зображення OCR.

На цьому етап програмування алгоритму першого методу можна вважати закінченим. Можна переходити до роботи над створенням застосунку за допомогою засобів мови Swift у Xcode.

Фотографія була головним напрямком діяльності Apple з моменту появи смартфонів iPhone. Протягом багатьох років вони випустили дивовижні нові функції. Користувачі можуть робити все кращі фотографії завдяки найсучаснішим функціям інтелектуального зображення, які Apple постійно додає.

Зокрема, Apple інвестує значні кошти в сферу комп'ютерного зору, випускаючи основні оновлення щороку через свою структуру Vision, яка була випущена в 2017 році. Завдяки функціям розпізнавання обличчя, відстеження об'єктів, якості зйомки та схожості зображень Apple дає можливість розробникам мобільних пристроїв інтегрувати складні алгоритми комп'ютерного зору для створення застосунків на основі фотографій на основі штучного інтелекту. Серед нових випусків виділяються функції Vision Text Recognition and Saliency.

Отже, як і реалізація попереднього методу, цей алгоритм буде запрограмовано послідовними кроками.

Створення файлу. Необхідно спочатку запустити середу розробки Xcode і створити новий файл програми. Обов'язково треба застосувати опис політики конфіденційності камери, додавши ключ NSCameraUsageDescription у файл info.plist.

Налаштування введення з камери. У наступному фрагменті програми проводиться налаштування задньої камери з типом медіа як відео. Цей тип додається до функції AVCaptureSession. Далі, потрібно додати канал камери до перегляду ViewController.

Відображення попереднього перегляду камери та налаштування виведення. Реалізуються функції, які відображають живу трансляцію з камери та налаштовують вивід. Вихідні відеокадри зрештою будуть передані на запит Vision. Щоб отримати кадри камери, потрібно відповідати протоколу AVCaptureVideoDataOutputSampleBufferDelegate та реалізувати функцію captureOutput. Після цього можна вважати, що камера готова до використання.

Налаштування запиту на виведення зображення. Тепер час налаштувати запит на виявлення прямокутника Vision. У функції detectRectangles встановлюється VNDetectRectanglesRequest, він передається обробнику запиту зображення для початку обробки. Результат, який повертає запит Vision в обробнику завершення, має тип VNRectangleObservation, який складається з boundingBox і значення достовірності.

Використовуючи властивість обмежувальної рамки, можна зробити шар поверх камери, де буде виявлено прямокутник.

Функція `doPerspectiveCorrection` використовується для виправлення зображення у разі його спотворення. Ця функція активується, коли користувач натискає кнопку «Сканувати», щоб отримати повністю обрізану картку з каналу камери.

Створення обмежувальних рамок у камері. Координати обмежувальної рамки `Vision` належать нормалізованій системі координат, початок якої – лівий нижній кут екрана.

Отже, потрібно перетворити обмежувальну рамку `Vision CGRect` у систему координат зображення. Замість того, щоб виконувати `CGAffineTransform` для перетворення обмежувальної рамки в координатний простір зображення, можна використати такі вбудовані методи, доступні у структурі `Vision` у функції `VNNormalizedRectForImageRect` з атрибутами `imageRect`, `imageWidth`, `imageHeight`. Коли `maskLayer` встановлено на виявленому прямокутнику в каналі камери, отримується зображення картки із кутами (рис. 3.4). Наступний крок включає вилучення зображення в межах обмежувальної рамки.



Рисунок 3.4 – Зображення картки із її визначеною формою

Вилучення зображення з обмежувальної рамки. Функція `doPerspectiveCorrection` бере основне зображення з буфера, перетворює його кути та застосовує до них фільтр корекції перспективи, щоб отримати зображення. Функція `UIImageWriteToSavedPhotosAlbum` використовується для збереження зображення в бібліотеці фотографій на пристрої користувача. Застосування фільтра корекції перспективи до основного зображення фіксує орієнтацію зазначеного зображення.

Оптимізація розпізнавання відображеного тексту. Далі потрібно розглянути, як витягти лише потрібний текст зі сканованого зображення.

У iOS 13 Vision було оновлення із включенням текстових ідентифікатора у `VNRecognizeTextRequest`, який раніше лише повідомляв присутній текст чи ні. Щоб розібрати значення зазначеного тексту довелося використовувати моделі Core ML. Зокрема, було застосовано регулярний вираз, який не тільки неефективний, але й не працює універсально.

Використання шаблону регулярного виразу потребує підтримки багатьох крайніх випадків для фільтрації різних типів кредитних карток (наприклад, не всі картки мають 16 цифр, American Express має 15).

Щоб створити рухомий прямокутник, відстежується дотик користувача до сканованого зображення та перемалюємо вибрану область.

Коли користувач вибирає прямокутну область і натискає кнопку «Витягти», вирізається зображення з прямокутника та передається до запиту Vision. На цьому кроці застосунок можна вважати готовим до застосування та тестування.

3.3 Тестування розроблених застосунків та аналіз результатів

Здійснивши програмну реалізацію, можна перейти до етапу тестування та аналізу результатів програмного застосунку. З цією метою було створено датасет із 15 зображень. Для прикладу у роботі наведено декілька основних цікавих результатів тестувань.

Першими переглянемо результати методу із використанням OpenCV та Python.

Після завантаження зображення застосунок вміє визначити картку на зображенні, а саме її прямокутну форму. На рисунку 3.5 можна побачити, як програма визначила 4 групи цифр номеру картки. Платіжна система розпізнана вірно. Цей етап пройшов вдало.

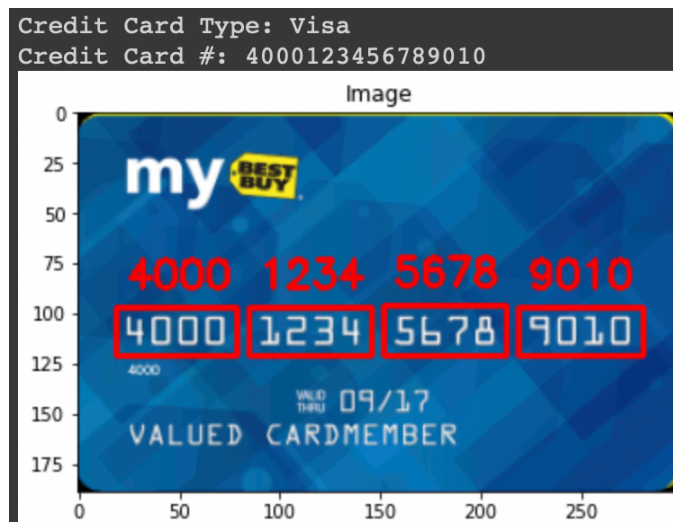


Рисунок 3.5 – Визначення регіону номеру картки

У наступному випадку протестовано зображення трохи погіршеної якості, де не всі цифри чіткі (рис. 3.6).

Застосунок вірно визначає 4 регіони номеру картки, проте знаходить зайву цифру. Усі цифри розпізнані правильно, окрім однієї. Платіжна система розпізнана вірно.

Наступна перевірка зображення із дуже поганою якістю (рис. 3.7).

На жаль, ні одна цифра номеру не визначена правильно, платіжна система також не розпізнана вірно.

Перевірка наступного зображення із датасету дає успішні результати, які показано на рисунку 3.8. Це зображення із дуже високим рівнем затемнення. Неочікувано, проте номер розпізнаний правильно, кожна із цифр вірна, помилки та невизначеності відсутні. Платіжна система розпізнана вірно. Застосунок OpenCV та Python успішно демонструє свою роботу.



Рисунок 3.6 – Визначення номеру картки

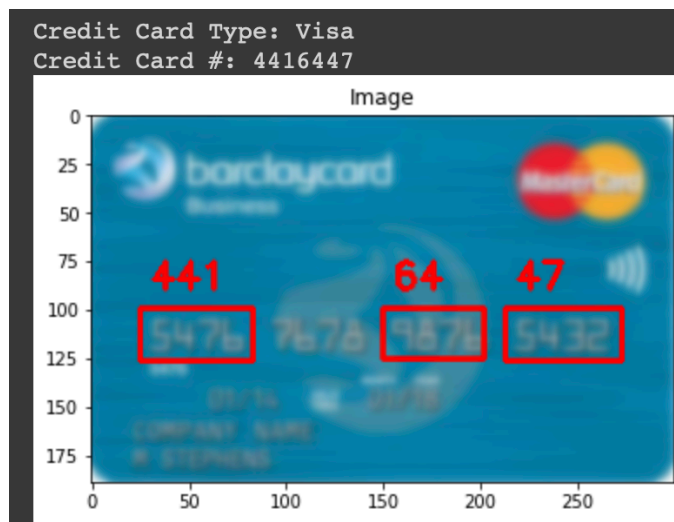


Рисунок 3.7 – Зображення картки дуже поганої якості

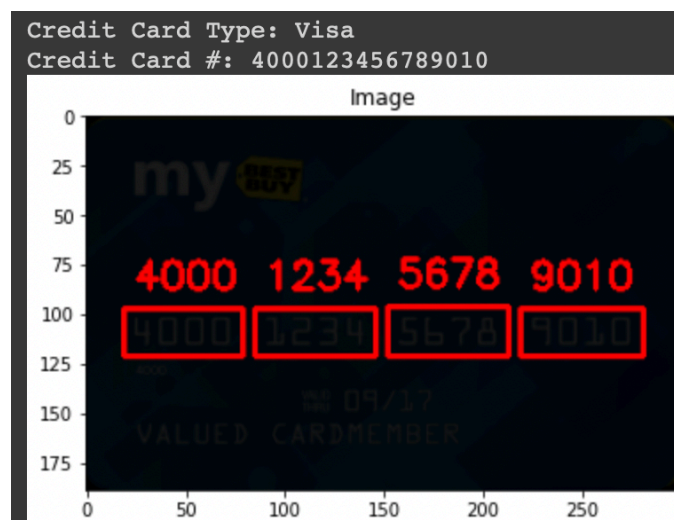


Рисунок 3.8 – Зображення картки із її визначеною формою

Тепер переходимо до аналізу результатів роботи застосунку створеного за допомогою Swift.

На рисунку 3.9 можна побачити перший результат сканування, де метод успішно визначив картку, використовуючи камеру та чітко окреслив її прямокутну форму, що є задовільним результатом.



Рисунок 3.9 – Результат визначення форми картки застосунком на Swift

Далі демонструється робота розглянутої раніше функції `doPerspectiveCorrection`, використовується для виправлення зображення у разі його спотворення (рис. 3.10). Спочатку застосунок визначає, що картка лежить, але не рівно та окреслює приблизні її рамки.

На другому етапі функція автоматично визначає, що картка лежить не рівно та виправляє це, успішно виконуючи свою задачу. Картка приближується, вирівнюється, а зайві частини зображення прибираються (рис. 3.11).

Після цього на третьому етапі алгоритм автоматично визначає ім'я та прізвище на картці, виділяючи його. З цим завданням він справляється успішно та без помилок (рис. 3.12). Для визначення номеру картки користувач натискає «Extract digits».



Рисунок 3.10 – Перший етап виявлення контуру картки на спортвореному зображенні

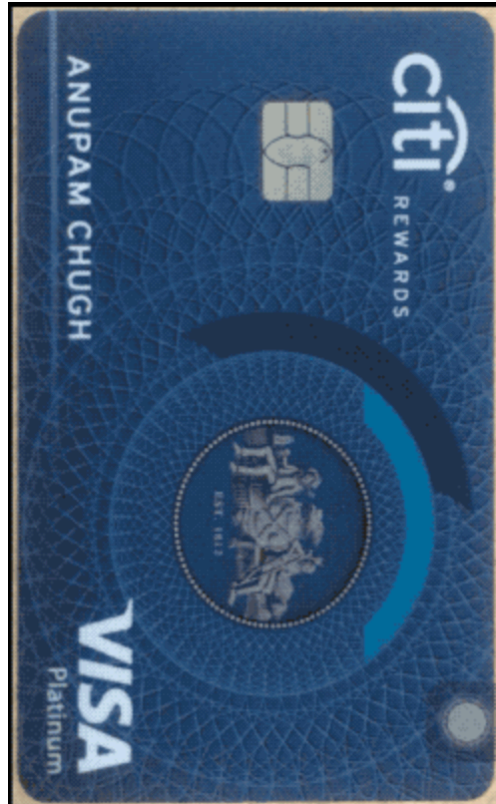


Рисунок 3.11 – Другий етап виявлення контуру картки на спотвореному зображенні



Рисунок 3.12 – Ві картки із її визначеною формою

Детальне порівняння результатів усіх тестувань двох методів наведено у двох таблицях: таблиця 3.1 із методом OpenCV та Python та таблиця 3.2 методу Swift.

Таблиця 3.1 – Результати тестових порівнянь методу OpenCV з Python

№ порівняння	№ тестового зображення	Опис тестового зображення	Відсоток розпізнаних даних	Платіжна система розпізнана правильно
1	1	Ідеальні умови	100	Так
2	2	Ідеальні умови	100	Так
3	3	Ідеальні умови	100	Так
4	4	Ідеальні умови	75	Ні
5	5	Ідеальні умови	100	Так
6	6	Під кутом 10°	100	Так
7	7	Під кутом 90°	0	Ні
8	8	Під кутом 180°	0	Ні

Продовження таблиці 3.1

№ порівняння	№ тестового зображення	Опис тестового зображення	Відсоток розпізнаних даних	Платіжна система розпізнана правильно
9	9	Вид збоку	50	Так
10	10	Вид збоку	25	Ні
11	11	Погана якість	93,75	Так
12	12	Погана якість	100	Так
13	13	Погана якість	0	Ні
14	14	Затемнення	0	Ні
15	15	Затемнення	100	Так

Таблиця 3.2 – Результати тестових порівнянь методу Swift

№ порівняння	№ тестового зображення	Опис тестового зображення	Відсоток розпізнаних даних	Платіжна система розпізнана правильно
1	1	Ідеальні умови	100	Так
2	2	Ідеальні умови	100	Так
3	3	Ідеальні умови	100	Так
4	4	Ідеальні умови	75	Так
5	5	Ідеальні умови	75	Так
6	6	Під кутом 10°	100	Так
7	7	Під кутом 90°	0	Ні
8	8	Під кутом 180°	0	Ні
9	9	Вид збоку	100	Так
10	10	Вид збоку	100	Так
11	11	Погана якість	50	Так
12	12	Погана якість	25	Ні

Продовження таблиці 3.2

№ порівняння	№ тестового зображення	Опис тестового зображення	Відсоток розпізнаних даних	Платіжна система розпізнана правильно
13	13	Погана якість	0	Ні
14	14	Затемнення	50	Так
15	15	Затемнення	0	Ні

Реалізувавши два методи зчитування реквізитів банківських карток можна зробити висновок, що вони загалом працюють непогано та справляються зі своїм завданням.

Використовувався запит на виявлення прямокутника, для OpenCV та Python із датасету зображень, для методу Swift з подачі камери в реальному часі. Таким чином, програми виявляли кредитну картку та вирізали її, враховуючи її орієнтацію та викривлення.

Рухаючись далі, застосовано розпізнавання тексту на зображенні, яке було щойно відскановано, щоб витягнути цифри кредитної картки. Дозволяючи користувачеві вибрати область для аналізу. Таким чином, це не лише дає йому можливість контролювати свою інформацію, але й позбавляємося непотрібної інформації.

Встановлення обмежувальної рамки для розпізнаних текстів і виконання тестів на їх попадання може бути ще одним способом витягти потрібну інформацію – із картки.

3.4 Перспективи подальшої роботи

Реалізовані два методи можна покращити, додавши новий функціонал, щоб кожен із них вмів, наприклад, визначати дату закінчення строку дії, а також розпізнавання іншої сторони картки.

Ще, можливо, окремі деталі брендингу картки, які можуть бути корисні для певних застосунків.

Важливо також додатково пропрацювати точність виявлення реквізитів при різних умовах (якщо картку видно не повністю, погане освітлення, якість фото тощо).

Взявши до уваги ці деталі, програми стануть суттєво кориснішими, та їх можна буде застосовувати в комерційних цілях.

ВИСНОВКИ

У рамках кваліфікаційної роботи було проведено детальний аналіз сучасних системи розпізнавання реквізитів банківських карток з метою огляду стану індустрії та перспектив розвитку.

Визначено класифікацію та проведено аналіз можливих існуючих методів розпізнавання реквізитів банківських карток. Виконано дослідження особливостей реквізитів банківських карток, таких як номер картки, срок дії картки, ім'я та прізвище власника та інші візуальні деталі.

Проаналізовано літературні джерела щодо апробації результатів розпізнавання реквізитів банківських карток із метою формування бази знань про індустрію розпізнавання банківських карток. Визначено можливий механізм оброблення зображення банківських карток, а також досліджено важливі принципи розпізнавання складових банківських карток.

Успішно засвоєно визначені за планом методики розпізнавання реквізитів банківських карток. Вибрано інструментальні засоби для реалізації методів розпізнавання реквізитів банківських карток. Проведено аналіз та визначено етапи програмної реалізації методів розпізнавання реквізитів банківських карток із детальним описом процесу.

Після реалізації обраних методів проведено тестування розробленого застосунку, використовуючи датасет із підготовленими зображеннями банківських карток, та, на основі цього, здійснено відповідний аналіз результатів. Виходячи із отриманих результативних даних, запропоновано можливі перспективи подальшої роботи.

Мету роботи досягнуто.

Результати роботи апробовано у вигляді тез доповідей під час XXXVII Міжнародної науково-практичної конференції у м. Варна, Болгарія [64].

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Raka, P., Agrwal, S., Kolhe, K., Karad, A., Pujeri, R. V., Thengade, A., & Pujeri, U. (2019). OCR to read credit/debit card details to autofill forms on payment portals. *Int. J. Res. Eng. Sci. Manag*, 2(4), 478-481.
2. Hamad, K., & Mehmet, K. A. Y. A. (2016). A detailed analysis of optical character recognition technology. *International Journal of Applied Mathematics Electronics and Computers*, (Special Issue-1), 244-249.
3. Save, P., Tiwarekar, P., Jain, K. N., & Mahyavanshi, N. (2017). A novel idea for credit card fraud detection using decision tree. *International Journal of Computer Applications*, 161(13).
4. Tvoroshenko I.S., and Gorokhovatsky V.O. (2019) Intelligent classification of biophysical system states using fuzzy interval logic, *Telecommunications and Radio Engineering*, 78(14), pp. 1303-1315.
5. Tvoroshenko Irina, Ahmad M. Ayaz, Mustafa Syed Khalid, Lyashenko Vyacheslav, and Alharbi Adel R. (2020) Modification of Models Intensive Development Ontologies by Fuzzy Logic, *International Journal of Emerging Trends in Engineering Research*, 8(3), pp. 939-944.
6. Кучеренко, Е. И., Филатов, В. А., Творошенко, И. С., & Байдан, Р. Н. (2005). Интеллектуальные технологии в задачах принятия решений технологических комплексов на основе нечеткой интервальной логики. *Восточно-Европейский журнал передовых технологий*, (2), 92-96.
7. Lyashenko V., Mustafa S.K., Tvoroshenko I., and Ahmad M.A. (2020) Methods of Using Fuzzy Interval Logic During Processing of Space States of Complex Biophysical Objects, *International Journal of Emerging Trends in Engineering Research*, 8(2), pp. 372-377.
8. Daradkeh Y.I., and Tvoroshenko I. (2020) Technologies for Making Reliable Decisions on a Variety of Effective Factors using Fuzzy Logic, *International Journal of Advanced Computer Science and Applications*, 11(5), pp. 43-50.

9. Tvoroshenko I.S., and Gorokhovatsky V.O. (2020) Effective tuning of membership function parameters in fuzzy systems based on multi-valued interval logic, *Telecommunications and Radio Engineering*, 79(2), pp. 149-163.
10. Molloy, I., Li, J., & Li, N. (2007, February). Dynamic virtual credit card numbers. In *International Conference on Financial Cryptography and Data Security* (pp. 208-223). Springer, Berlin, Heidelberg.
11. Творошенко, І. С., Шевченко, А. Р. (2018) Удосконалення просторової мережі навчальних закладів міста Сєвєродонецька на основі геоінформаційного аналізу. *Системи обробки інформації*, (1), 46-52.
12. Кобилін, О. А., & Творошенко, І. С. (2021). Методи цифрової обробки зображень.
13. Tvoroshenko I.S., and Kramarenko O.O. (2019) Software determination of the optimal route by geoinformation technologies, *Radio Electronics Computer Science Control*, 3, pp. 131-142.
14. Asaad Ma. Babker, Abd Elgadir A. Altoum, Irina Tvoroshenko, and Vyacheslav Lyashenko (2019) Information Technologies of the Processing of the Spaces of the States of a Complex Biophysical Object in the Intellectual Medical System HEALTH, *International Journal of Advanced Trends in Computer Science and Engineering*, 8(6), pp. 3221-3227.
15. Villar, A. S., & Khan, N. (2021). Robotic process automation in banking industry: a case study on Deutsche Bank. *Journal of Banking and Financial Technology*, 5(1), 71-86.
16. Hussein, K. W., Sani, N. F. M., Mahmud, R., & Abdullah, M. T. (2013). Enhance Luhn algorithm for validation of credit cards numbers. *Int. J. Comput. Sci. Mob. Comput*, 2(7), 262-272.
17. Wang, N., Zhu, X., & Zhang, J. (2015, April). Research of ID card recognition algorithm based on neural network pattern recognition. In *2015 International Conference on Mechatronics, Electronic, Industrial and Control Engineering (MEIC-15)* (pp. 964-967). Atlantis Press.

18. Костюк, І. С. (2019). *Алгоритм і програма розпізнавання реквізитів банківських карток* (Bachelor's thesis, КПІ ім. Ігоря Сікорського).
19. Творошенко, І. С., & Табашник, В. А. (2018). Розробка просторової моделі геоінформаційної підтримки людей з обмеженими можливостями, що пересуваються на інвалідних колясках, у місті Харків. *Збірник наукових праць Харківського національного університету Повітряних Сил*, (1), 122-128.
20. Gorokhovatskyi, V. A., Rusakova, N., & Tvoroshenko, I. S. (2020). The application of image analysis methods and predicate logic in applied problems of magnetic monitoring. *Telecommunications and Radio Engineering*, 79(20).
21. Tvoroshenko, I., & Tkachenko, D. (2020). Mechanisms of image classification based on descriptors of local features.
22. Tvoroshenko, I., & Babochkin, O. (2021). Object identification method based on image keypoint descriptors.
23. Кучеренко, Є. І., Творошенко, І. С., Анопрієнко, Т. В. (2016) Моделювання та оцінювання станів складних об'єктів із застосуванням формальної логіки. *Системи обробки інформації*, (2), 76-82.
24. Творошенко, І. С. (2010). Анализ процессов принятия решений в интеллектуальных системах. *Системи обробки інформації*, (2), 248-253.
25. Творошенко, І. С. (2018). Особливості застосування сучасних принципів штучного інтелекту до розробки ефективних механізмів моделювання складних систем. *Science and Technology of the Present Time: Priority Development Directions of Ukraine and Poland*, 118-121.
26. Xin, Y., Lin, Y., Shi, P., Han, S., & Tian, B. (2019, November). An Automatic Recognition Method for Bank Card Number. In *Journal of Physics: Conference Series* (Vol. 1345, No. 2, p. 022049). IOP Publishing.
27. Baggio, D. L. (2012). *Mastering OpenCV with practical computer vision projects*. Packt Publishing Ltd.
28. Lin, H., Zhan, Y., Liu, S., Ke, X., & Chen, Y. (2022). A deep learning based bank card detection and recognition method in complex scenes. *Applied Intelligence*, 1-19.

29. Wong, S. Y. T. (2017). iOS application to recognise CityU timetable's content.
30. Raka, P., Agrwal, S., Kolhe, K., Karad, A., Pujeri, R. V., Thengade, A., & Pujeri, U.(2019). OCR to read credit/debit card details to autofill forms on payment portals. *Int. J. Res. Eng. Sci. Manag*, 2(4), 478-481.
31. Dong, C., Zhu, X., Deng, Y., Loy, C. C., & Qiao, Y. (2015). Boosting optical character recognition: A super-resolution approach. *arXiv preprint arXiv:1506.02211*.
32. Pimentel, J., & Bernardino, A. A Comparison of Methods for Detection and Recognition of Playing Cards.
33. Varghese, T., Nambiar, V., Dandekar, P., & Hegde, G. (2018). Authentication of Credit Card Using Facial Recognition. *IJLTEMAS*, April.
34. Keskar, V. (2020). Comparing different models for credit card fraud detection. *Journal of Critical Reviews*, 7(2), 981-986.
35. Xin, Y., Lin, Y., Shi, P., Han, S., & Tian, B. (2019, November). An Automatic Recognition Method for Bank Card Number. In *Journal of Physics: Conference Series* (Vol. 1345, No. 2, p. 022049). IOP Publishing.
36. Mori, S., Nishida, H., & Yamada, H. (1999). *Optical character recognition*. John Wiley & Sons, Inc..
37. Gayathri, S., & Mohana, R. S. (2019, December). Optical Character Recognition in Banking Sectors Using Convolutional Neural Network. In *2019 Third International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC)*(pp. 753-756). IEEE.
38. Nasir, T., Malik, M. K., & Shahzad, K. (2021). MMU-OCR-21: Towards End-to-End Urdu Text Recognition Using Deep Learning. *IEEE Access*, 9, 124945-124962.
39. Datta, K. (2011). Credit Card Processing Using Cell Phone Images. *Jan*, 1, 1-7.
40. Wang, W. (2022). Designing User Interfaces with SwiftUI. In *Beginning iPhone Development with SwiftUI* (pp. 17-34). Apress, Berkeley, CA.

41. Wilde, M., Hategan, M., Wozniak, J. M., Clifford, B., Katz, D. S., & Foster, I. (2011). Swift: A language for distributed parallel scripting. *Parallel Computing*, 37(9), 633-652.
42. Tvoroshenko I.S., and Gorokhovatsky V.O. (2019) Modification of the branch and bound method to determine the extremes of membership functions in fuzzy intelligent systems, *Telecommunications and Radio Engineering*, 78(20), pp. 1857-1868.
43. Кучеренко, Е. И., Творошенко, И. С. (2010) Прикладные аспекты моделирования нечетких процессов в сложных системах. *Збірник наукових праць Харківського університету Повітряних сил*, (1), 127-131.
44. Tvoroshenko, I., & Dziubenko, M. (2020). Modern methods of analysis of the movement scheme using video detection of vehicles.
45. Tvoroshenko, I. (2019). Development of models of spatial analysis of status of interactive processes of complex systems.
46. Kelly, M., & Nozzi, J. (2013). *Mastering Xcode: Develop and Design*. Peachpit Press.
47. Tvoroshenko I., and Zarivchatskyi R. (2020) Analysis of existing methods for searching object in the video stream, Abstracts of VI International Scientific and Practical Conference «About the problems of science and practice, tasks and ways to solve them» (October 26-30, 2020). Milan, Italy, pp. 500-505.
48. Tvoroshenko, I., & Kukharchuk, V. (2021). Current state of development of applications for recognition of faces in the image and frames of video captures.
49. Daradkeh Y.I., Gorokhovatskyi V., Tvoroshenko I., and Al-Dhaifallah M. (2022) Classification of Images Based on a System of Hierarchical Features, *Computers, Materials & Continua*, 72(1), pp. 1785-1797.
50. Tvoroshenko I., and Gorokhovatskyi V. (2022) The Application of Hybrid Intelligence Systems for Dynamic Data Analysis, *International Journal of Engineering and Information Systems*, 6(2), pp. 40-48.
51. Гороховатський В.О., Творошенко І.С. (2022) Аналіз багатовимірних даних за описом у формі множини компонент: монографія. Харків: ХНУРЕ. 124 с.

52. Ahmad M.A., Gorokhovatskyi V., Tvoroshenko I., Vlasenko N., and Mustafa S.K. (2021) The Research of Image Classification Methods Based on the Introducing Cluster Representation Parameters for the Structural Description, *International Journal of Engineering Trends and Technology*, 69(10), pp. 186-192.

53. Гороховатський В., Творошенко І., Сидоренко Д. (2021) Класифікація зображень із використанням кластерного подання. Міжнародний науковий симпозиум «Інтелектуальні рішення-С». Обчислювальний інтелект (результати, проблеми, перспективи). Теорія прийняття рішень: праці міжн. наук. симпозиуму (Вересень 29, 2021). Київ – Ужгород, С. 44-45.

54. Гороховатський В.О., Творошенко І.С., Чмутів Ю.В. (2022) Застосування систем ортогональних функцій для формування простору ознак у методах класифікації зображень. *Сучасні інформаційні системи*, 6 (3), С. 5–12.

55. Daradkeh Y.I., Gorokhovatskyi V., Tvoroshenko I., and Zeghid M. (2022) Cluster representation of the structural description of images for effective classification, *Computers, Materials & Continua*, 73 (3), pp. 6069–6084.

56. Lienhart, R., & Wernicke, A. (2002). Localizing and segmenting text in images and videos. *IEEE Transactions on circuits and systems for video technology*, 12(4), 256-268.

57. Kamisetty, V. N. S. R., Chidvilas, B. S., Revathy, S., Jeyanthi, P., Anu, V. M., & Gladence, L. M. (2022, March). Digitization of Data from Invoice using OCR. In *2022 6th International Conference on Computing Methodologies and Communication (ICCMC)* (pp. 1-10). IEEE.

58. Zhao, Y., Deng, X., & Lai, H. (2020). A deep learning-based method to detect components from scanned structural drawings for reconstructing 3D models. *Applied Sciences*, 10(6), 2066.

59. Cai, A., Qiu, B., Tan, X., & Li, S. (2020). Bank Card Recognition Method Based on Mobile Terminal.

60. Ou, Z., Xiong, B., Xiao, F., & Song, M. (2020). ERCS: An efficient and robust card recognition system for camera-based image. *China Communications*, 17(12), 247-264.

61. Liu, Y. (2020). Research on Identification of Bank Card Number Based on Deep Learning.
62. Shang-lin, L. I., Lu-da, W. A. N. G., & Dong, L. I. U. (2020). Digital recognition method of bank card based on CNN. *Journal of Graphics*, 41(1), 81.
63. Ge, J., Fang, Z., & Tao, Q. (2020, June). Bank Card Number Recognition based on Deep Learning. In *2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)* (Vol. 1, pp. 863-867). IEEE.
64. Бабочкін О. (2022) Аналіз сучасних систем розпізнавання реквізитів банківських карток, *Abstracts of XXXVII International Scientific and Practical Conference «Modern ways of solving the latest problems in science» (September 20 – 23, 2022)*. Varna, Bulgaria, pp. 430-434.