

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ  
ІМЕНІ ТАРАСА ШЕВЧЕНКА  
Факультет інформаційних технологій  
Кафедра інтелектуальних технологій**

**КВАЛІФІКАЦІЙНА РОБОТА  
на здобуття освітнього ступеня «магістр»  
НА ТЕМУ:**

Система пошуку та ідентифікації документів на фотографії для мобільних пристроїв на базі операційної системи iOS

Галузь знань: 12 «Інформаційні технології»

Спеціальність: 122 «Комп'ютерні науки»

Освітньо-наукова програма «Технології штучного інтелекту»

Виконав:

студент 2 курсу магістратури  
групи ТШП-21

Бондар Максим Олександрович  
(ПІБ)

Науковий керівник:

Красовська Ганна Валеріївна  
(ПІБ)

кандидат технічних наук, доцент  
(науковий ступінь, вчене звання)

Кваліфікаційна робота допущена до захисту  
рішенням кафедри *інтелектуальних технологій*

Протокол № \_\_\_\_ від « \_\_\_\_ » травня 2020 р.

В.о. зав. кафедри \_\_\_\_\_ доц. Красовська Г.В.  
підпис

**Київ 2020**

## РЕФЕРАТ

Кваліфікаційна робота складається зі вступу, 3 розділів, висновків, списку використаної літератури із 54 джерел та 1 додатку. Загальний обсяг роботи 95 сторінок. Робота містить 18 таблиць та 56 рисунків.

Актуальність теми. В останні роки якість мобільної фотографії значно зросла, що дозволяє використовувати смартфони для фіксації важливих документів із конференцій чи засідань. Останнім часом все більшої популярності набувають застосунки, які дозволяють сканувати документи за допомогою камери. Проте основною проблемою, притаманною даним застосункам, є те, що результати сканування є ідеальними лише коли мобільні пристрої знаходяться максимально паралельно до сканованого документу, при гарному освітленні, коли на площині знаходиться лише один документ та коли мобільний пристрій не рухомий під час фотографії, інакше результати сканування можуть бути непридатними для роботи. Також значна частина програмних продуктів на ринку має недоліки у вигляді поганої якості результатів сканування документів та відсутності можливості сканування кількох документів одночасно. Саме тому необхідно створити технологію, яка дозволила би максимально усунути виявлені недоліки та розширити функціонал за рахунок визначення класів документів.

Мета роботи: дослідження та розробка застосунку для мобільних пристроїв на базі операційної системи iOS, який працює відповідно до розробленої технології для пошуку та ідентифікації документів на фотографії за допомогою нейронних мереж.

Об'єктом дослідження – процес сканування та класифікації документів на фотографії.

Предметом дослідження – методи пошуку та ідентифікації документів на фотографії із використанням нейронних мереж.

Результати роботи та наукова новизна. В результаті виконання роботи було створено застосунок для мобільних пристроїв на базі операційної

системи iOS, який за основу використовує розроблену технологію для сканування та ідентифікації документів. Наукова новизна одержаних результатів роботи:

- удосконалено технологію пошуку документів на фотографії в напрямках збільшення швидкості сканування документів, сканування декількох документів одночасно, пошуку тексту на документах та визначення розмитості зображення;
- вперше було запропоновано технологію класифікації документів використовуючи згорткові нейронні мережі, що дозволяє групувати відскановані документи за типами.

Апробація роботи. Основні наукові результати та практичні розробки магістерської роботи доповідались і обговорювались на конференціях:

- 1) міжнародний науковий симпозіум «Інтелектуальні рішення» (м. Ужгород, 15–20 квітня 2019 р.);
- 2) міжнародна науково-практична конференція «Інформаційні технології та взаємодії» (м. Київ, 20 грудня 2019 р.).

Ключові слова. Сканування, класифікація документів, згорткова нейронна мережа, попередня обробка зображення, аналіз якості зображення.

## STRUCTURAL ABSTRACT

Qualification work consists of an introduction, 3 chapters, conclusions, a list of references from 54 sources and 1 annex. The total volume of the work is 95 pages. The work contains 18 tables and 56 figures.

Relevance. During the last years, the quality of mobile photography has increased significantly, which allows the use smartphones to capture important documents from conferences or meetings. Recently, applications that allow you to scan documents with a camera are becoming more and more popular. However, the main problem with these applications is that scan results are ideal only when mobile devices are located as parallel as possible to the scanned document, in good light, when there is only one document on the plane, and when the mobile device is stabilized during the photo, otherwise, the scan results may be unusable. Also, a significant part of software products on the market has disadvantages in the form of poor quality of document scanning results and the lack of ability to scan multiple documents simultaneously. That is why it is necessary to create a technology that would eliminate the identified shortcomings and expand the functionality by defining classes of documents.

The goal: research and development of an application for mobile devices based on the iOS operating system, which works in accordance with the developed technology for searching and identifying documents in the photo using neural networks.

The object of research – the process of scanning and classifying documents in photographs.

The subject of research – methods of search and identification of documents in photographs using neural networks.

Results of work and scientific novelty. As a result of the work, the application for mobile devices based on the iOS operating system was created, which uses the developed technology for scanning and identification of documents. Scientific novelty of the obtained results of work:

- improved technology for searching documents in photos in the areas of increasing the speed of scanning documents, scanning multiple documents simultaneously, searching for text on documents and determining image blur;
- for the first time, the technology of document classification using convolutional neural networks was proposed, which allows to group scanned documents by types.

Approbation of work. The main scientific results and practical developments of the master's thesis were reported and discussed at conferences:

- 1) International Scientific Symposium "Intelligent Solutions" (Uzhhorod, April 15-20, 2019);
- 2) International scientific-practical conference "Information Technologies and Interactions" (Kyiv, December 20, 2019).

Keywords. Scanning, document classification, convolutional neural network, image pre-processing, image quality analysis.

## Зміст

|                                                                                                  |    |
|--------------------------------------------------------------------------------------------------|----|
| ВСТУП                                                                                            | 8  |
| РОЗДІЛ 1. СИСТЕМНИЙ АНАЛІЗ ТЕХНОЛОГІЇ ІДЕНТИФІКАЦІЇ ТА КЛАСИФІКАЦІЇ ДОКУМЕНТІВ НА ФОТОГРАФІЇ     | 12 |
| 1.1 Дослідження ринку програмних застосунків                                                     | 12 |
| 1.1.1 Застосунок Adobe Scan                                                                      | 14 |
| 1.1.2 Застосунок HP Smart                                                                        | 15 |
| 1.1.3 Застосунок FineScanner                                                                     | 17 |
| 1.1.4 Застосунок iScanner                                                                        | 18 |
| 1.1.5 Застосунок Notes                                                                           | 20 |
| 1.1.6 Узагальнення результатів порівняльного аналізу застосунків                                 | 21 |
| 1.2 Узагальнена технологія пошуку документів на фотографії                                       | 23 |
| 1.3 Постановка задач розробки системи для пошуку та ідентифікації документів на фотографії       | 25 |
| РОЗДІЛ 2. ТЕХНОЛОГІЯ ПОШУКУ ТА ІДЕНТИФІКАЦІЇ ДОКУМЕНТІВ НА ФОТОГРАФІЇ                            | 27 |
| 2.1 Технологія ідентифікації та класифікації документів на фотографії                            | 27 |
| 2.2 Аналіз технологій пошуку прямокутних областей на фотографії                                  | 32 |
| 2.3 Аналіз методів покращення та визначення якості зображення для попередньої обробки зображення | 35 |
| 2.3.1. Аналіз методів приведення відсканованого документу у відтінки сірого                      | 35 |
| 2.3.2. Аналіз методів приведення відсканованого документу у чорно-біле                           | 37 |
| 2.3.3. Аналіз методів зміни яскравості зображення                                                | 38 |
| 2.3.4. Аналіз методів зміни контрастності зображення                                             | 39 |

|                                                                                                                |    |
|----------------------------------------------------------------------------------------------------------------|----|
| 2.3.5. Аналіз методів аналізу якості зображення                                                                | 40 |
| 2.4 Аналіз методів класифікації зображень для етапу сканування документів та класифікації за типом             | 41 |
| 2.5 Вибір засобів розробки                                                                                     | 45 |
| 2.5.1 Порівняльний аналіз засобів реалізації методу Віоли-Джонса для пошуку областей із ймовірними документами | 45 |
| 2.5.2 Аналіз інструментів побудови та навчання згорткової нейронної мережі                                     | 47 |
| 2.6 Висновки до другого розділу                                                                                | 49 |
| РОЗДІЛ 3. ОПИС РЕАЛІЗАЦІЇ ЗАСТОСУНКУ                                                                           | 50 |
| 3.1 Визначення архітектури згорткової нейронної мережі                                                         | 50 |
| 3.2 Навчання згорткової нейронної мережі                                                                       | 52 |
| 3.3 Архітектура застосунку для пошуку та ідентифікації документів на фотографії                                | 60 |
| 3.4. Опис програмного інтерфейсу користувача застосунку для пошуку та ідентифікації документів на фотографії   | 66 |
| 3.5 Аналіз результатів роботи розробленого застосунку                                                          | 73 |
| 3.6 Висновки до третього розділу                                                                               | 85 |
| ВИСНОВКИ                                                                                                       | 86 |
| СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ                                                                                 | 88 |
| ДОДАТОК А                                                                                                      | 94 |

## ВСТУП

Актуальність теми. В даний момент часу швидкими темпами розвивається світ переносних та портативних пристроїв, до яких відносяться, в першу чергу, смартфони. Смартфони дозволяють замінити велику кількість більш громіздких пристроїв, таких як комп'ютери, фотоапарати, сканери тощо. З одного боку, вони й близько не можуть конкурувати з якістю чи швидкістю їх роботи, але смартфони можуть вміщувати в тій чи іншій мірі деяку частину можливостей вище наведених пристроїв.

В останні роки, якість мобільної фотографії зросла в десятки, а то й сотні разів, що дозволяє використовувати смартфони для фіксації важливих моментів життя: від походу із сім'єю у парк до важливих документів із конференцій чи засідань, щоб у майбутньому переглянути їх детальніше і провести більш прискіпливий аналіз. Саме сканування документів за допомогою смартфона дозволяє зменшити витрати людини на громіздкі та здебільшого незручні пристрої для сканування документів, але й також, значною мірою, дозволяє економити найдорожчий людський ресурс – час.

Враховуючи те, що на смартфонах роздільна здатність камери дозволяє передавати на фотографії дрібні деталі, наприклад, текст та позначки на малюнку чи схемі, останнім часом все більшої популярності набувають застосунки, які дозволяють сканувати документи за допомогою камери. Здебільшого дані застосунки мають однакові функціональні можливості: сканування документів, вибілювання, групування в один документ сканованих сторінок і т. д.

Проте основна проблема притаманна даним застосункам, що результати сканування є ідеальними, лише коли мобільні пристрої знаходяться максимально паралельно до сканованого документу, при гарному освітленні, коли один документ знаходиться на площині, та коли мобільний пристрій

не рухомий під час фотографії, інакше результати сканування можуть бути непридатними для роботи.

З цього випливає, що сканування документів є досить актуальною темою в даний період часу так як значній частині програмних продуктів на ринку притаманні недоліки у вигляді сканування документів поганої якості та відсутність можливості сканування кількох документів одночасно.

Дана кваліфікаційна робота спрямована на пошук технології для сканування та ідентифікації типів документів для покращення результатів сканування документів із фотографії. В результаті технологія повинна ідентифікувати типи документів для спрощення роботи із вже відсканованими документами, покращити якість відсканованих документів, збільшити кількість коректних сканувань та зменшити кількість хибних сканувань.

Випускна кваліфікаційна робота виконувалася в рамках наукових досліджень кафедри інтелектуальних технологій.

Об'єктом дослідження є процес сканування та класифікації документів на фотографії.

Предметом дослідження є методи пошуку та ідентифікації документів на фотографії із використанням нейронних мереж.

Мета дослідження. Мета дослідження полягає у розробці застосунку для мобільних пристроїв на базі операційної системи iOS, який працює відповідно до розробленої технології для пошуку та ідентифікації документів на фотографії за допомогою нейронних мереж.

Для досягнення поставленої мети сформульовані наступні задачі дослідження:

1. провести аналіз доступних на ринку програмних продуктів для сканування документів за допомогою камери смартфона;
2. сформулювати та описати технологію для пошуку та ідентифікації документів на фотографії із використанням нейронних мереж;
3. визначити архітектуру застосунку для пошуку документів на фотографії та інструменти для його розробки;

4. розробити застосунок для пошуку та ідентифікації документів на фотографії із використанням нейронних мереж та провести детальний аналіз результатів роботи.

Методи дослідження: аналіз ринку програмних продуктів, синтез та збір даних для навчання нейронної мережі, експеримент з пошуку оптимальних параметрів для навчання нейронної мережі, метод Віулі-Джонса, згорткова нейронна мережа для ідентифікації класів документів, методи приведення зображення до відтінків сірого, метод перетворення зображення у чорно-біле, методи регулювання яскравості та контрасту зображення, дисперсія Лапласа.

Наукова новизна одержаних результатів:

- удосконалено технологію пошуку документів на фотографії в напрямках збільшення швидкості сканування документів, сканування декількох документів одночасно, пошуку тексту на документах та визначення розмитості зображення;
- вперше було запропоновано технологію класифікації документів використовуючи згорткові нейронні мережі, що дозволяє групувати відскановані документи за типами.

Практичне значення одержаних результатів. В результаті дослідження розроблено застосунок для пошуку та ідентифікації документів на фотографії, який дозволяє сканувати одночасно декілька документів на фотографії, визначає приналежність відсканованої області до документів, визначає якість документа та тип документа.

Апробація результатів дослідження. Основні наукові результати та практичні розробки магістерської роботи доповідались і обговорювались на конференціях:

- 1) міжнародний науковий симпозіум «Інтелектуальні рішення» (м. Ужгород, 15–20 квітня 2019 р.) [1];
- 2) міжнародна науково-практична конференція «Інформаційні технології та взаємодії» (м. Київ, 20 грудня 2019 р.) [2].

За матеріалами магістерської роботи є 2 публікації у збірниках матеріалів науково-практичних конференцій.

Структура роботи. Відповідно до мети та завдань дослідження робота складається із вступу, 3 розділів, 15 підрозділів, висновків, списку використаної літератури із 53 найменувань, 1 додатку. Загальний обсяг роботи 91 сторінка.

В першому розділі проаналізовано доступні програмні продукти на ринку для пошуку та сканування документів на фотографії, визначено їх основні переваги та недоліки, сформовано загальну технологію сканування документів. В результаті зазначеного аналізу було поставлено задачі до вирішення.

В другому розділі детально описано розроблену технологію пошуку та ідентифікації документів на фотографії. Розглянуто способи покращення якості зображення та ідентифікації його розмитості. Визначено тип нейронної мережі для ідентифікації типів документів та технологію для її навчання.

В третьому розділі детально описано побудовану згорткову нейронну мережу та розроблений програмний продукт для мобільних пристроїв на базі iOS. Наведено архітектуру застосунку у вигляді дерева із програмними модулями та схему інтерфейсу користувача [3] у вигляді графу. Також проведено детальний аналіз даних отриманих під час навчання моделей згорткової нейронної мережі для вибору найкращої моделі для ідентифікації документів.

В результаті виконання роботи було розроблено систему для пошуку та ідентифікації документів на фотографії для мобільних пристроїв на базі операційної системи iOS. Розроблена система дозволила зменшити кількість хибних сканувань та збільшила кількість сканувань документів високої та середньої якості. За рахунок визначення класів документів вдалося розширити базові фільтри для сортування та групування документів.

## РОЗДІЛ 1. СИСТЕМНИЙ АНАЛІЗ ТЕХНОЛОГІЇ ІДЕНТИФІКАЦІЇ ТА КЛАСИФІКАЦІЇ ДОКУМЕНТІВ НА ФОТОГРАФІЇ

### 1.1 Дослідження ринку програмних застосунків

Досліджуючи дану предметну область було визначено декілька доступних програмних продуктів, які можливо знайти в магазині застосунків App Store так як фінальна система буде представлення у мобільному варіанті:

- 1) Adobe Scan від компанії Adobe Systems, Inc. (42 місце в категорії «Бізнес» в магазині застосунків) [4];
- 2) HP Smart від компанії HP Inc. (47 місце в категорії «Продуктивність» в магазині застосунків) [5, 6];
- 3) FineScanner від компанії ABBYY USA Software House Inc (29 місце в категорії «Бізнес» в магазині застосунків) [7];
- 4) iScanner від компанії BP Mobile LLC (33 місце в категорії «Бізнес» в магазині застосунків) [8];
- 5) вбудований сканер в застосунок Notes від компанії Apple (не бере участь у рейтингу застосунків) [9, 10].

Перш ніж аналізувати дані програмні продукти необхідно виділити функції, на які в першу чергу необхідно звернути увагу при скануванні документів (таблиця 1.1).

Таблиця 1.1 – Список функцій, які аналізувалися та їх детальний опис.

| № функції | Назва функції                     | Опис функції                                                                               |
|-----------|-----------------------------------|--------------------------------------------------------------------------------------------|
| 1         | Завантаження фотографії з галереї | Наявність можливості відправити фотографію із документом на сканування з галереї смартфона |
| 2         | Фотографії з камери               | Наявність можливості отримати фотографію із документом за допомогою камери смартфона       |

Таблиця 1.1 – Продовження

|   |                                            |                                                                                                                           |
|---|--------------------------------------------|---------------------------------------------------------------------------------------------------------------------------|
| 3 | Сканування одного документа камери/галереї | Наявність можливості відсканувати один документ із фотографії                                                             |
| 4 | Сканування декількох документів одночасно  | Наявність можливості відсканувати декілька документів із фотографії                                                       |
| 5 | Покращення якості відсканованого документа | Наявність можливості покращити якість відсканованого документа (вибілювання: переведення у відтінки сірого, згладжування) |
| 6 | Аналіз якості відсканованого документа     | Наявність можливості автоматичного відсіювання документів з поганою якістю (розмиті або не чіткі фотографії)              |
| 7 | Визначення типу документів                 | Наявність можливості автоматичного визначення типу документа                                                              |

В результаті аналізу зазначених у таблиці 1.1 функціональних можливостей можна буде виділити переваги та недоліки даних програмних продуктів та визначити можливі шляхи подолання недоліків.

### 1.1.1 Застосунок Adobe Scan

Застосунок Adobe Scan було розроблено компанією Adobe Systems, Inc (рисунок 1.1) .

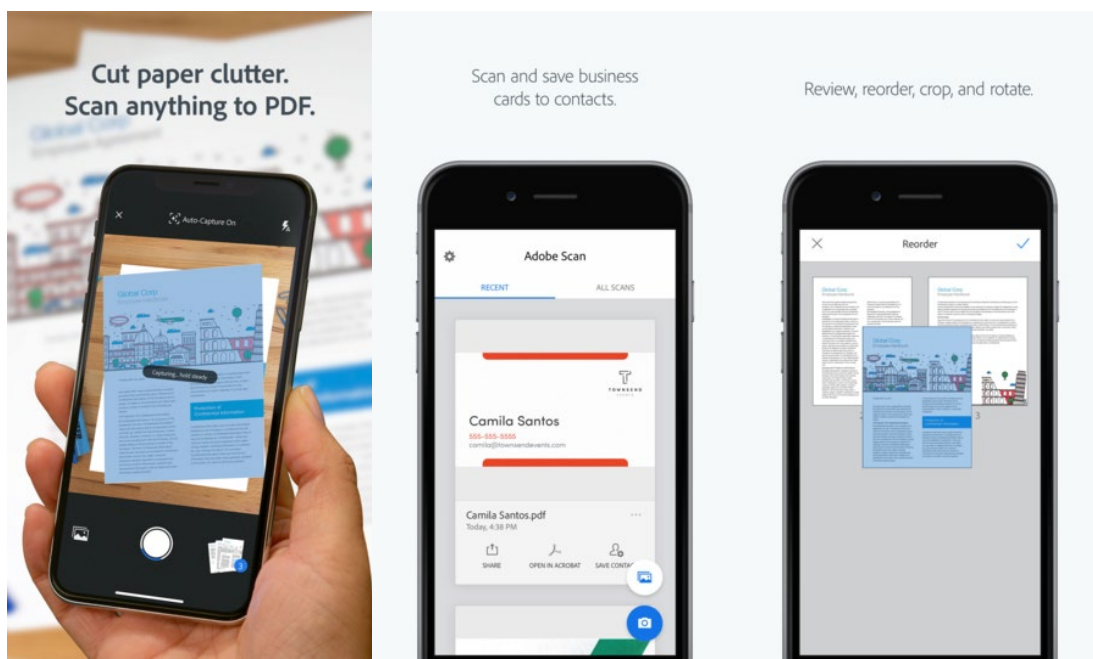


Рисунок 1.1 – Приклад інтерфейсу Adobe Scan із магазину застосунків AppStore

Даний програмний продукт має простий інтерфейс користувача, велику кількість інструментів для обробки відсканованих документів та обмежений функціонал із вже відсканованими документами.

Відповідно із наведеної інформації з сайту розробника даний програмний продукт дозволяє сканувати будь-які документи: візитівки, замітки, квитанції і т. д., дозволяє автоматично знаходити фотографії з документами, розпізнавання контактів та автоматичне їх додавання до списку контактів, інтеграція з іншими програмними продуктами для роботи з документами та багато інших функції [4].

Результати аналізу доступних функцій наведено у таблиці 1.2.

Таблиця 1.2 — Аналіз доступних функцій в програмному продукті Adobe Scan

| № функції | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|-----------|---|---|---|---|---|---|---|
| Наявність | + | + | + | - | + | - | - |

Проаналізувавши доступну інформацію на сайті розробника та результати аналізу в таблиці 1.2 можна зробити висновок, що даний програмний продукт має велику кількість функціоналу із редагування відсканованих документів та роботи із іншими програмними продуктами для роботи із документами. Але серед основних недоліків даного застосунку можна виділити відсутність можливості сканування декількох документів одночасно, можливості аналізу якості документів та визначення типів документів.

### 1.1.2 Застосунок HP Smart

Застосунок HP Smart було розроблено компанією HP Inc (рисунок 1.2).

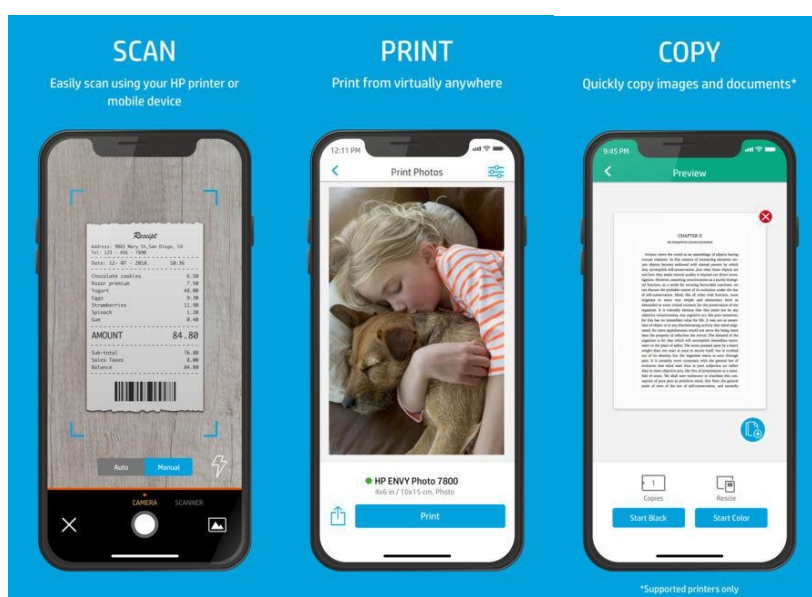


Рисунок 1.2 – Приклад інтерфейсу HP Smart із магазину застосунків AppStore

Виходячи із інформації наведеної розробником, даний програмний продукт в основному призначений для конфігурації принтерів та сканерів від компанії HP та лише як додаткову функцію має функцію сканування документів та відправлення їх одразу на друк чи збереження для подальшої роботи [5]. Серед основних функцій даного програмного продукту можна виділити широкий функціонал редагування відсканованих документів, збереження відсканованих документів у декількох форматах: PDF, JPEG та PNG.

Результати аналізу доступних функцій наведено у таблиці 1.3.

Таблиця 1.3 — Аналіз доступних функцій в програмному продукті HP Smart

| № функції | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|-----------|---|---|---|---|---|---|---|
| Наявність | + | + | + | - | + | - | - |

Проаналізувавши доступну інформацію на сайті розробника та результати аналізу в таблиці 1.3 можна зробити висновок, що даний програмний продукт має велику кількість функціоналу із редагування відсканованих документів та роботи із налаштованими домашніми пристроями – принтерами та сканерами. Але серед основних недоліків даного застосунку можна виділити відсутність можливості сканування декількох документів одночасно, можливості аналізу якості документів та визначення типів документів.

### 1.1.3 Застосунок FineScanner

Застосунок FineScanner розроблений компанією ABBYY USA Software House Inc (рисунок 1.3). Так як даний застосунок є платним, то його переваги та недоліки було отримано із використанням доступної інформації з інтернет ресурсів [11 – 13].

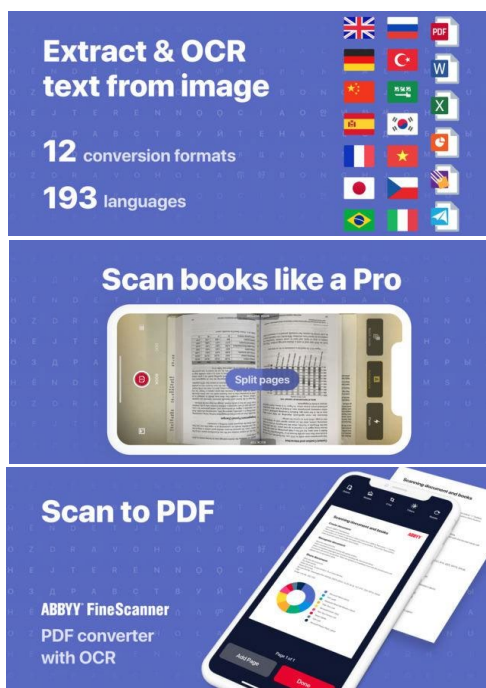


Рисунок 1.3 – Приклад інтерфейсу та функціональних можливостей FineScanner із магазину застосунків AppStore

Виходячи із інформації наведеної розробником, даний програмний продукт має значну кількість функцій для роботи із різними типами документів та дозволяє визначати типи об'єктів на документі: таблиці, помітки, текст, малюнки [7]. Серед основних функції даного застосунку можна виділити можливість сканування книжок, збереження відсканованих документів у декількох форматах та можливість розпізнавання тексту на 193 мовах.

Результати аналізу доступних функцій наведено у таблиці 1.4.

Таблиця 1.4 — Аналіз доступних функцій в програмному продукті FineScanner

| № функції | 1 | 2 | 3 | 4   | 5 | 6 | 7 |
|-----------|---|---|---|-----|---|---|---|
| Наявність | + | + | + | +/- | + | - | - |

Проаналізувавши доступну інформацію на сайті розробника та результати аналізу в таблиці 1.4 можна зробити висновок, що даний програмний продукт має велику кількість функціоналу із розпізнавання об'єктів на документах та має можливість із розпізнавання великої кількості мов на документах. Але серед основних недоліків даного застосунку можна виділити часткову можливість із сканування декількох документів, бо наявна можливість сканування книжок, можливості аналізу якості документів та визначення типів документів.

#### 1.1.4 Застосунок iScanner

Застосунок iScanner є розробкою компанії BP Mobile LLC (рисунок 1.4).

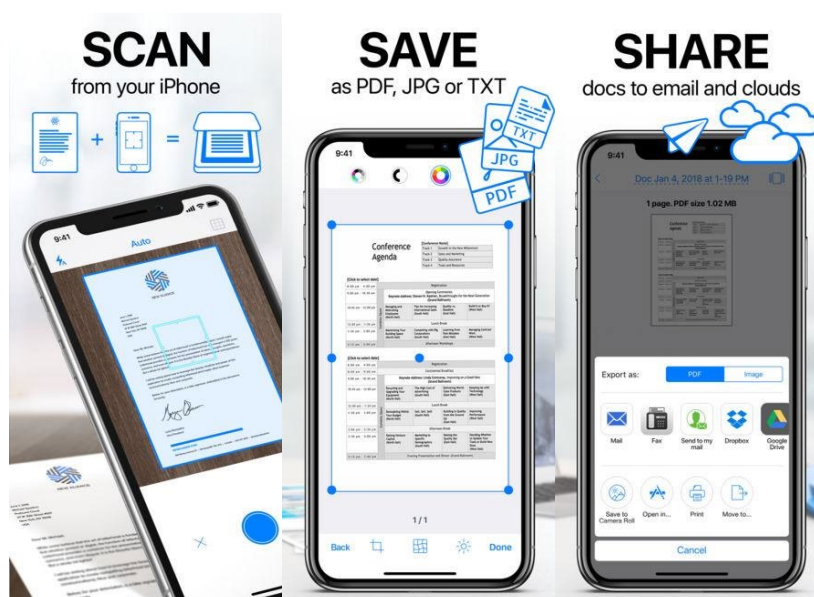


Рисунок 1.4 – Приклад інтерфейсу iScanner із магазину застосунків AppStore

Виходячи з інформації наведеної розробником, даний програмний продукт має незначну кількість функцій, але зроблено акцент на безпеці відсканованих документів за рахунок встановлення паролів на документи та можливості різного покращення якості документів: вибілювання, збільшення чіткості та інших функцій [8].

Результати аналізу доступних функцій наведено у таблиці 1.5.

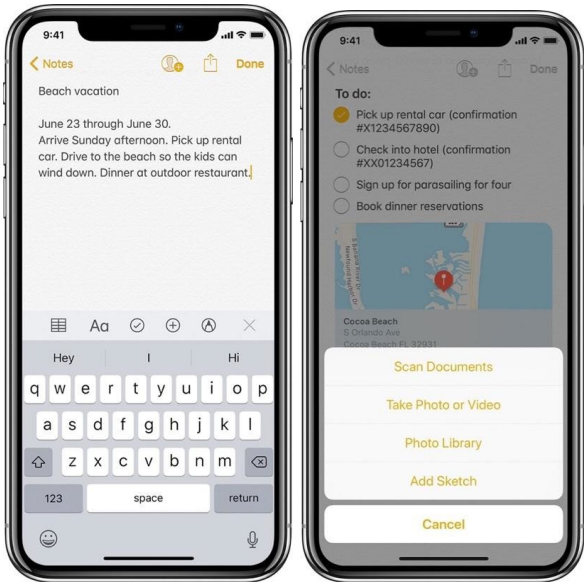
Таблиця 1.5 — Аналіз доступних функцій в програмному продукті iScanner

| № функції | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|-----------|---|---|---|---|---|---|---|
| Наявність | + | + | + | - | + | - | - |

Проаналізувавши доступну інформацію на сайті розробника та результати аналізу в таблиці 1.5 можна зробити висновок, що даний програмний продукт має велику кількість функцій із ручного та автоматичного покращення якості документів. Але серед основних недоліків даного застосунку можна виділити відсутність можливості сканування одночасно декількох документів, можливості аналізу якості документів та визначення типів документів.

1.1.5 Застосунок Notes

Застосунок Notes розроблено компанією Apple Inc (рисунок 1.5) [10].



Рисунки 1.5 – Приклади інтерфейсу застосунку Notes із офіційного сайту Apple

Виходячи із інформації наведеної розробником, даний програмний продукт в основному призначений для роботи із замітками, але як додаткову функцію має можливість вставки документів як фотографій у замітки [9, 10]. Дана програма має можливість вільного сканування документів – комфортне виділення області сканування, та можливість різноманітного редагування відсканованих документів.

Результати аналізу доступних функцій наведено у таблиці 1.6.

Таблиця 1.6 — Аналіз доступних функцій в програмному продукті iScanner

|           |   |   |   |   |   |   |   |
|-----------|---|---|---|---|---|---|---|
| № функції | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| Наявність | - | + | + | - | + | - | - |

Проаналізувавши доступну інформацію на сайті розробника та результати аналізу в таблиці 1.6 можна зробити висновок, що даний програмний продукт має досить простий функціонал, який є досить скудний у порівнянні із попередніми програмними продуктами. Але серед основних недоліків даного застосунку можна виділити відсутність можливості сканування документів із галереї мобільного пристрою, сканування одночасно декількох документів, можливості аналізу якості документів та визначення типів документів.

#### 1.1.6 Узагальнення результатів порівняльного аналізу застосунків

В результаті проведеного дослідження ринку можна зробити висновок, що більшість застосунків тій чи іншій мірі справляються із поставленими цілями – сканування документів, але крім відсутності деяких функцій, які досліджувалися їм притаманні ще й інші суттєві недоліки: хибні сканування предметів, що не мають відношення до документів (рисунок 1.6) та неспроможність відсканувати декілька документів на фото (рисунок 1.7) (хіба що FineScanner дозволяє сканувати книжки).



Рисунок 1.6 – Приклад хибного сканування клавіатури як документа

- Список літератури.
1. Бевз Г.П. Алгебра: Проб. Пар. Для 7-9 кл. серед. Шк.. 3-тє вид.—К.: Освіта, 2001.—303с.
  2. Галицкой, М.Л., Гольдман, А.М., Завич, Л.И. Сборник задач по алгебре для 8-9 классов; М.: Просвещение, 1992. - 272 с.
  3. Александров, А.Д., Вернер, А.Л., Рыжик, В.И. Геометрия для 8-9 классов: Учебное пособие для учащихся школ и классов с углубленным изучением математики; М.: Просвещение, 1991. - 415 с.
  4. Ю.И.; Миндюк, Н.Г.; Монахов, В.М. и др. Алгебра 8 класс; М.: Просвещение, 2000. - 250 с.
  5. С.А. Сборник задач для подготовки и проведения письменного экзамена по алгебре за курс основной школы. 9 класс; М.: АСТ, 2005. - 256 с.
  6. Земляков, А.Н. Геометрия в 9 класі: Просвещение, 2006. - 208 с.
  7. А.В. Погорелов. Учебник по геометрии для 7 - 11 классов средней школы; М.: Просвещение, 2001. - 383. с
  8. Атанасян Л.С., Бутузов В.Ф. и др. Геометрия. 7-9 классы; М.: Просвещение.

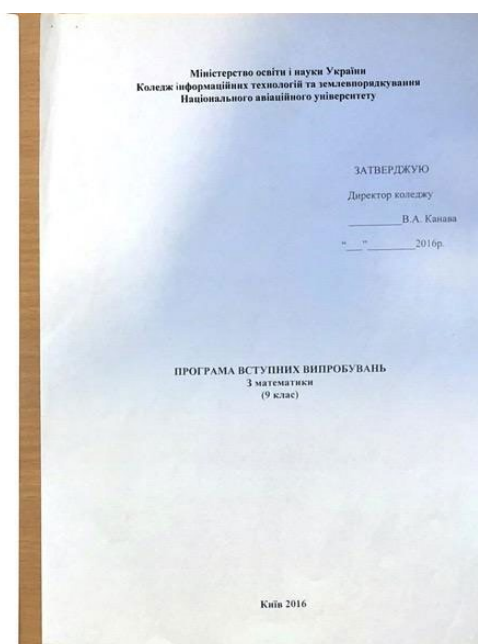


Рисунок 1.7 – Приклад сканування двох документів як одного

Для більшої наочності застосунки можна порівняти за даними наведеними нижче в таблиці 1.7.

Таблиця 1.7 – Доступність необхідних функцій в програмних продуктах з магазину мобільних застосунків AppStore.

| № функції | Adobe Scan | HP Smart | FineScanner | iScanner | Notes |
|-----------|------------|----------|-------------|----------|-------|
| 1         | +          | +        | +           | +        | -     |
| 2         | +          | +        | +           | +        | +     |
| 3         | +          | +        | +           | +        | +     |
| 4         | -          | -        | -           | -        | -     |
| 5         | +          | +        | +           | +        | +     |
| 6         | -          | -        | -           | -        | -     |
| 7         | -          | -        | -           | -        | -     |

Проведений аналіз ринку програмних продуктів для ідентифікації та класифікації документів на фотографії підтвердив необхідність створення застосунку для пошуку документів на фотографії, який дозволив би подолати виявлені недоліки під час аналізу.

## 1.2 Узагальнена технологія пошуку документів на фотографії

Кожний проаналізований застосунок звісно має свою архітектуру та особливості, але в основному вони мають одну узагальнену технологію, притаманну для них (рисунок 1.8)

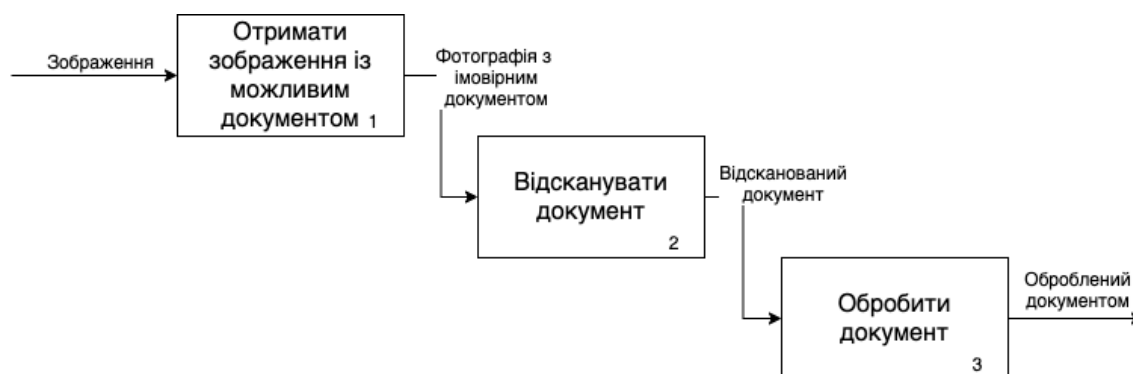


Рисунок 1.8 – Узагальнена технологія сканування документів

Відповідно з рисунку 1.8 видно, що узагальнена технологія включає в себе всього три етапи. Розглянемо більш детально кожен із них.

Етап отримання зображення для сканування зазвичай може бути виконано декількома способами:

- отримання зображення з камери;
- отримання зображення з галереї пристрою;
- отримання зображення з хмарних сховищ.

Перший та другий спосіб отримання зображення для сканування присутній у всіх проаналізованих програмних продуктах. А от отримання зображення з хмарних сховищ є не досить поширеною функцією, але значно спрощує процес роботи з документами.

Етап сканування документів також представлений в вигляді декількох способів:

- сканування в реальному часі при роботі з камерою;
- сканування документа як окремий процес.

Останній спосіб сканування документа присутній у всіх проаналізованих програмних продуктах, а от сканування в реальному часі є особливістю лише декількох застосунків. Проте саме завдяки скануванню в реальному часі, досить часто існує ймовірність сканування документів у поганий якості: недостатньо освітлених чи розмитих документів.

Етап обробки зображення у проаналізованих програмних продуктах зазвичай представлений у вигляді наступних функцій:

- корекція кольорів;
- регулювання яскравості;
- регулювання контрастності;
- корекція меж відсканованого документа;
- додавання поміток на документі.

Регулювання яскравості, контрастності та корекція кольорів є найбільш поширеними функціями серед усіх проаналізованих застосунків.

В результаті використання даної технології більшість програмних продуктів здійснює сканування документів з фотографії. Проте саме така простота технології призводить до того, що користувачі отримують відскановані документи поганої якості, які не можуть бути використані в подальшому.

### 1.3 Постановка задач розробки системи для пошуку та ідентифікації документів на фотографії

В результаті проведеного аналітичного огляду технології ідентифікації та класифікації документів на фотографії було аргументовано необхідність побудови системи для сканування та класифікації документів (рисунок 1.9).

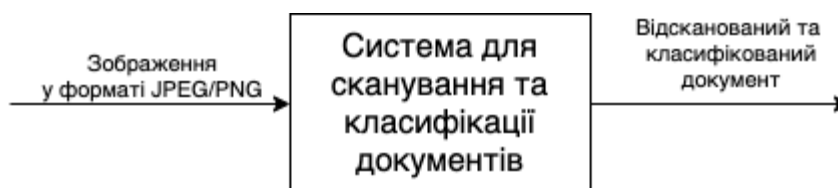


Рисунок 1.9 – Система для сканування та класифікації документів

Необхідно, щоб система задовольняла наступні функціональні вимоги:

- сканування документів з вже зроблених фотографій;
- сканування документів зроблених з камери;
- сканування кількох документів одночасно;
- корекція яскравості та контрастності;
- визначення якості документів;
- визначення класів відсканованих документів;
- перегляд відсканованих документів;
- збереження відсканованих документів;
- сортування документів за різними параметрами: датою, класом документу, якістю.

Також необхідно, щоб система задовольняла наступні технологічні вимоги:

- застосунок має працювати на мобільних пристроях на базі операційної системи iOS не нижче версії 12.2;
- застосунок повинен працювати лише у вертикальному положенні без зміни орієнтації інтерфейсу при повороті екрану;

- обсяг використовуваної операційної пам'яті не має перевищувати 150 Мб на піку роботи;
- час пошуку документів на фотографії не має перевищувати 0.4 секунди;
- похибка визначення типів документів має бути менше 15%.

В результаті виконання поставлених задач буде розроблена система для пошуку та ідентифікації документів на фотографії для мобільних пристроїв на базі операційної системи iOS.

## РОЗДІЛ 2. ТЕХНОЛОГІЯ ПОШУКУ ТА ІДЕНТИФІКАЦІЇ ДОКУМЕНТІВ НА ФОТОГРАФІЇ

### 2.1 Технологія ідентифікації та класифікації документів на фотографії

В результаті аналізу ринку програмних продуктів було сформовано власну технологію ідентифікації та класифікації документів на фотографії. Дана технологія включає в себе наступні етапи:

- 1) отримання зображення із можливим документом;
- 2) підготовка зображення до аналізу та класифікації;
- 3) сканування документа(-ів) та класифікація за типом.

Враховуючи те, що дана технологія буде впроваджуватись для мобільних пристроїв на операційній системі iOS, то увесь подальший опис відбувається з урахуванням особливостей розробки саме для даної операційної системи. Загальний опис даної технології представлено на рисунку 2.1.

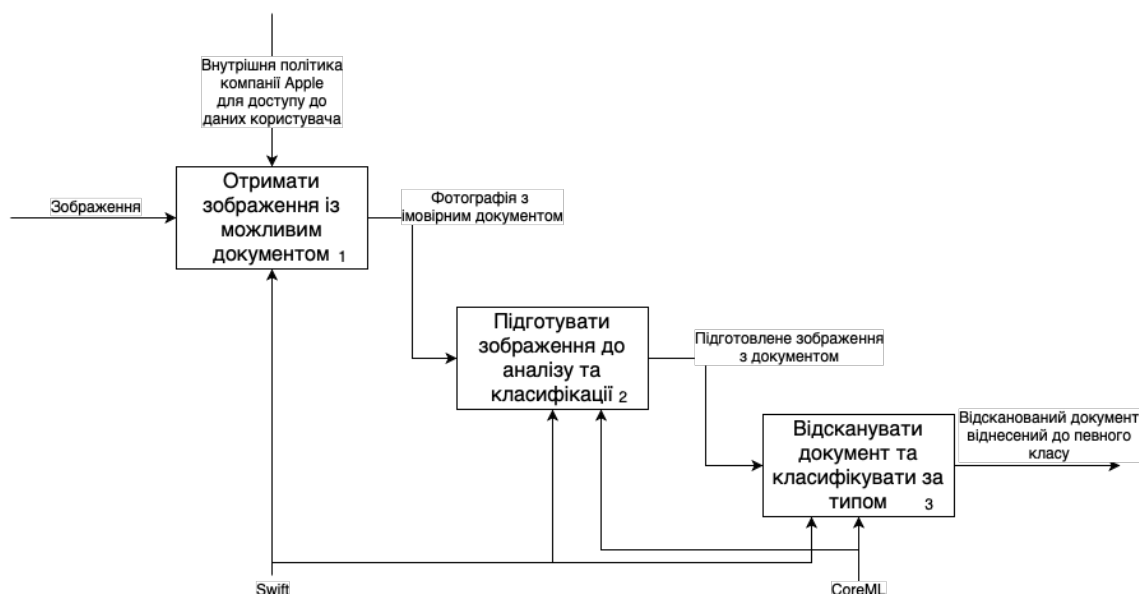


Рисунок 2.1 – Технологія пошуку та ідентифікації документів на фотографії

Розпочнемо із процесу отримання зображення із можливим документом. Процес отримання зображення із можливим документом, якщо не брати до уваги потокове відео, можливо розглянути як два окремі етапи (рисунок 2.2).



Рисунок 2.2 – Етапи отримання зображення із можливим документом

Отримання зображення із можливим документом може здійснюватися у двох напрямках: з камери або з галереї смартфона. Враховуючи рекомендації Apple [19] з метою забезпечення швидкодії обробки та використання меншої кількості пам'яті мобільний застосунок для операційної системи iOS має приймати на вхід зображення формату PNG або JPEG.

Наступним етапом є підготовка зображення до аналізу та класифікації (рисунок 2.3).

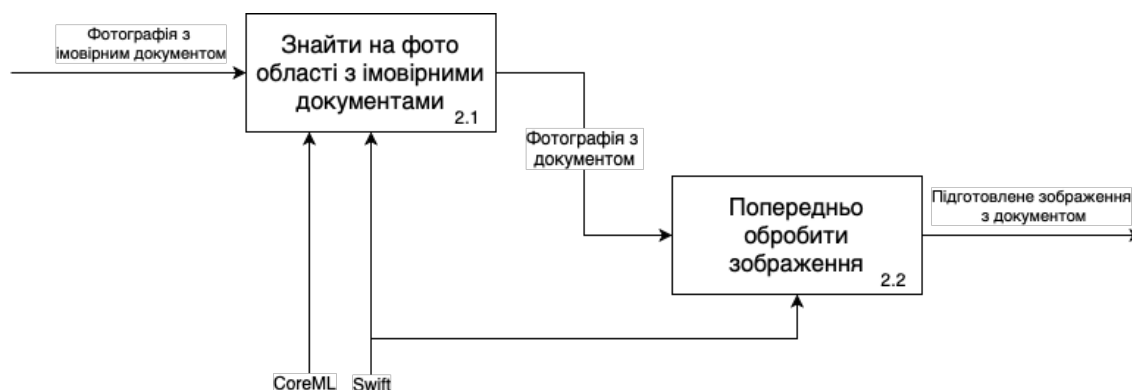


Рисунок 2.3 – Схематичний опис етапу підготовки зображення до аналізу та класифікації

Етап пошуку областей із імовірним документом призначений для того, щоб зменшити площу пошуку документів і зменшити кількість неправильно ідентифікованих документів, наприклад, ідентифікацію двох документів, які знаходяться поруч як одного документа.

Далі здійснюється покращення документа в декілька етапів. Відповідно один за одним виконуються наступні покращення: фільтрація документів, які є занадто розмиті та корекція кольорів. Наступним кроком на даному етапі є визначення кількості документів на фотографії для одночасного сканування декількох документів. Усі ці етапи теоретично дозволять усунути недоліки, що були виявлені у наявних програмних продуктах. Процес підготовки зображення до аналізу та класифікації наведено на рисунку 2.4.

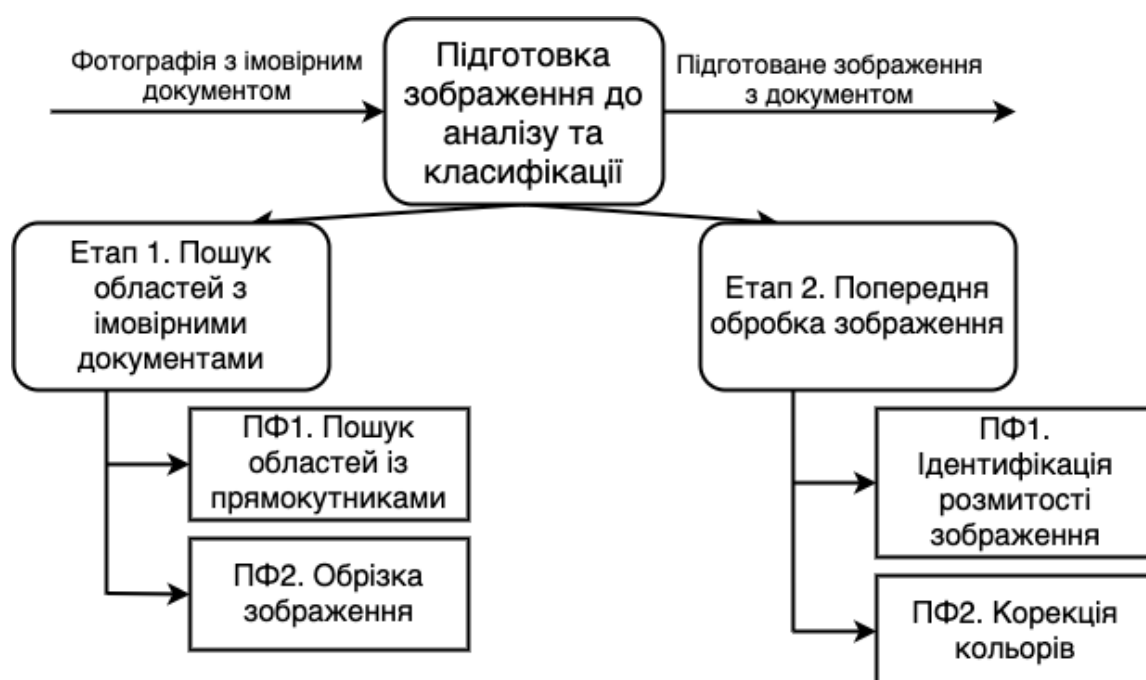


Рисунок 2.4 – Підготовка зображення до аналізу та класифікації

Після виділення областей з документами на фотографії підготовлене зображення передається на етап сканування та ідентифікації (класифікації) документа (рисунок 2.5). Ідентифікація документів за типами є важливою рисою запропонованої технології, що дозволяє здійснювати групування вже

відсканованих документів за типами, спрощує навігацію, сортування та надає можливість розширення варіації фільтрів. На етапі 3.1 до обробки приймається підготовлене зображення з інформацією про усі можливі прямокутники на фотографії, чіткість зображення та наявні області із текстом.

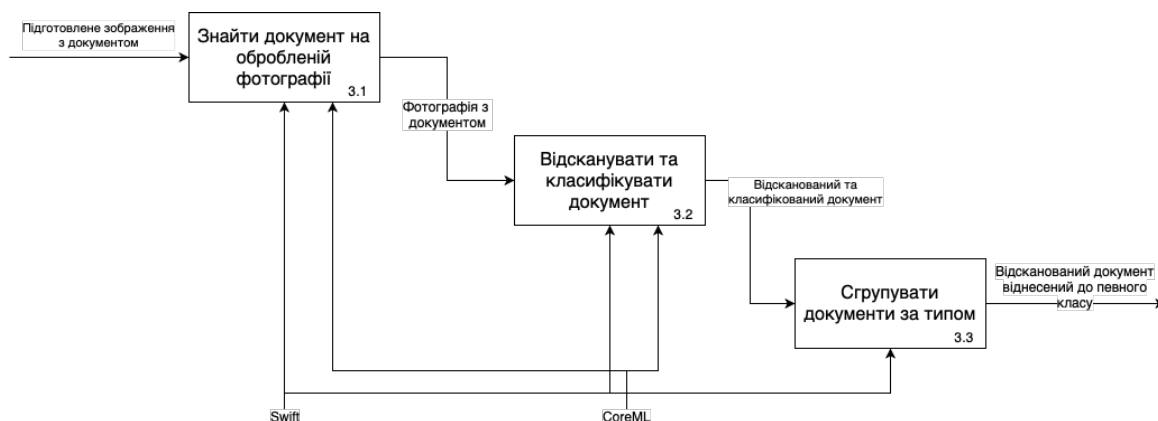


Рисунок 2.5 – Опис етапу сканування документа та класифікації за  
ТИПОМ

Виходячи із схеми вище процес сканування документа включає в себе наступні 3 етапи:

- 1) пошук документа на обробленій фотографії;
- 2) сканування та класифікація документа;
- 3) групування документів за типом.

Кожен із етапів включає декілька підфункцій зображених на рисунку 2.6.

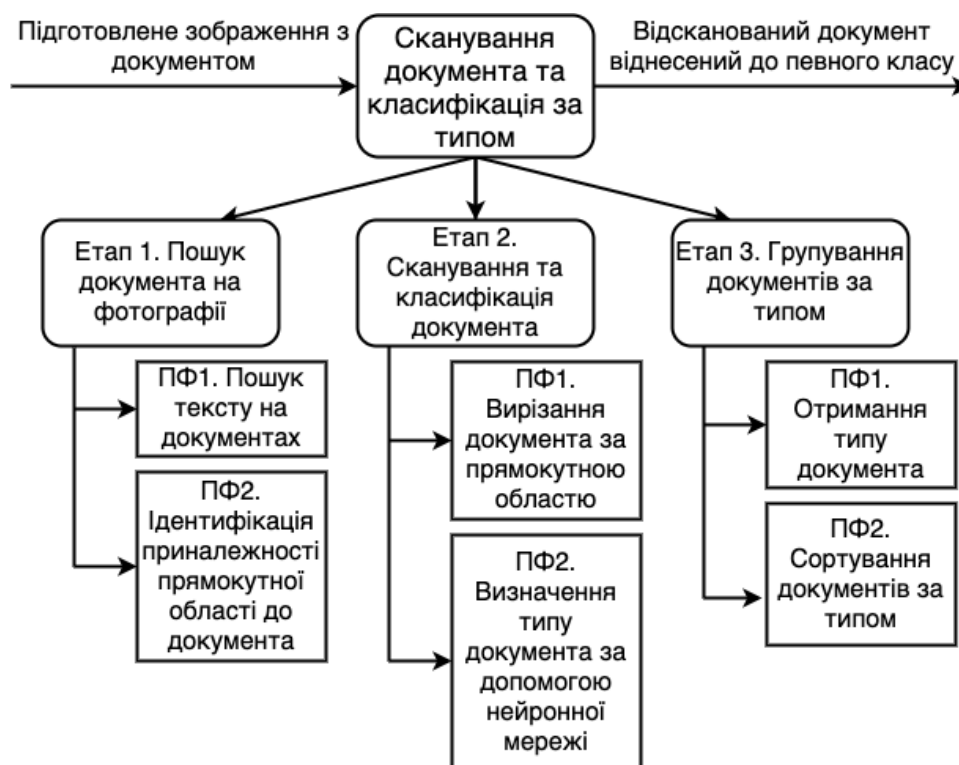


Рисунок 2.6 – Процес сканування документа та його класифікації

Відповідно для першого етапу головними підфункціями є пошук тексту на документах та ідентифікація приналежності прямокутної області до документа, які фактично взаємопов'язані. Пошук тексту на прямокутних областях дозволить визначити їх приналежність до документів та зменшить кількість хибних сканувань.

Головною підфункцією другого етапу є визначення типу документа. Визначення типу документу дозволить виконати 3 етап та розширює стандартний функціонал програмних продуктів.

Відповідно на третьому етапі беруться доступні відскановані документи та сортуються у відповідні групи за документом.

В результаті сканування та ідентифікації типів документів на фотографії буде отримано документи віднесені до певних класів із можливістю сканування декількох лише якісних документів. Теоретично використання даної технології дозволить подолати усі раніше виявлені недоліки та дозволить розширити базовий функціонал за допомогою визначення типів документів.

## 2.2 Аналіз технологій пошуку прямокутних областей на фотографії

Для ідентифікації документів на фотографії перш за все необхідно знайти межі документів. Враховуючи те, що документ це переважно прямокутна область, то необхідно проаналізувати методи для пошуку прямокутних областей.

Дана задача може бути вирішена значною кількістю алгоритмів для виділення меж документів:

- оператор Собеля;
- оператор Робертса;
- виділення меж Кенні;
- оператор Превітта;
- оператор Шара;
- диференціальне виділення меж;
- метод Віюлі-Джонса.

Щоб порівняти ефективність використання кожного методу нижче наведено приклади їх роботи.

На рисунку 2.7 наведено оригінальне зображення, яке обробляється за допомогою різних методів та операторів виділення границь.

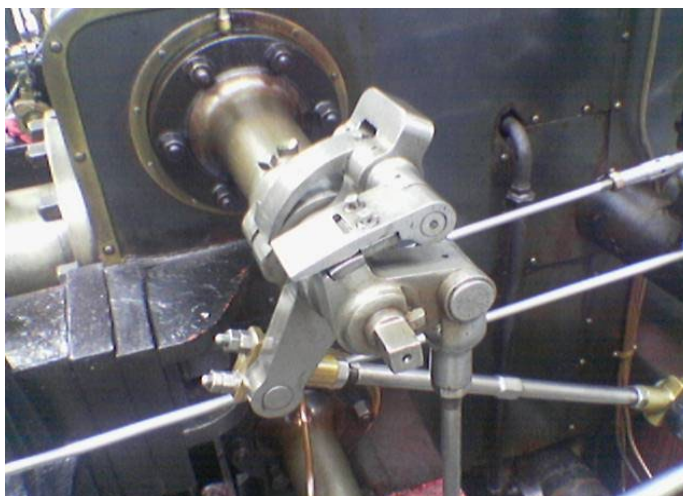


Рисунок 2.7 – Оригінальне зображення [14]

На рисунках 2.8 – 2.11 наведені результати використання операторів Робертса, Превітта, Собеля та Кенні відповідно.



Рисунок 2.8 – Результат використання оператора Робертса [14]

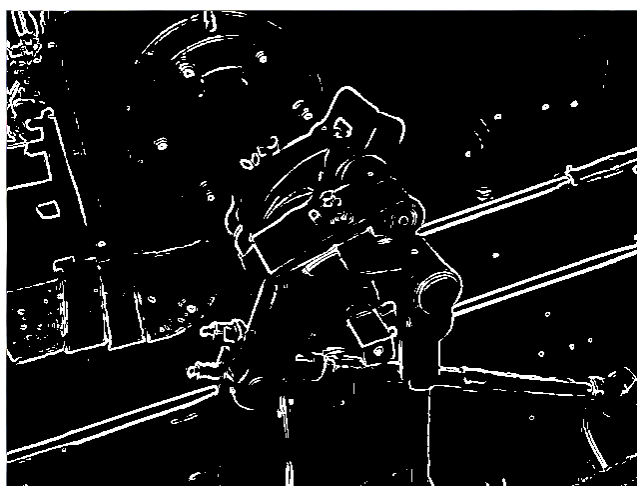


Рисунок 2.9 – Результат використання оператора Превітта [14]

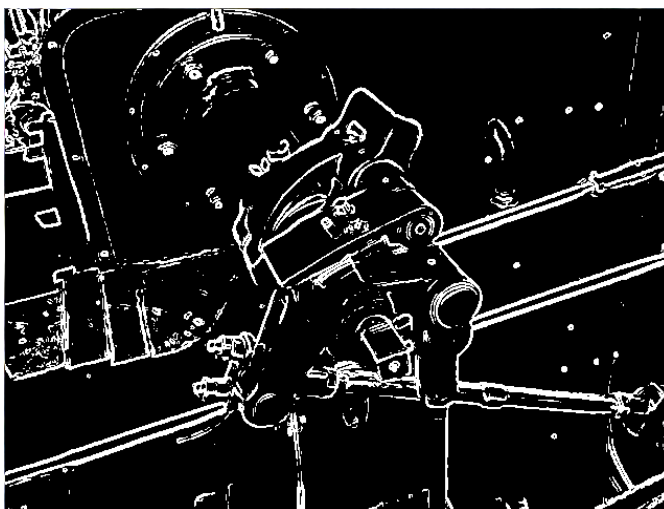


Рисунок 2.10 – Результат використання оператора Собеля [15]

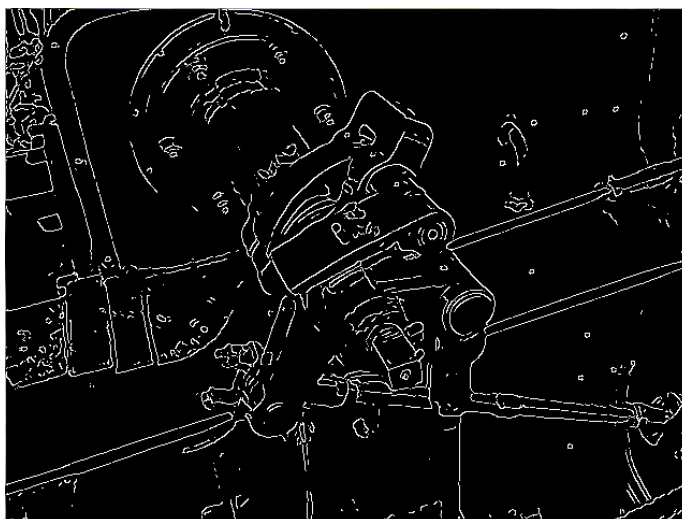


Рисунок 2.11 – Результат використання оператора Кенні [15]

Проте всі методи, крім останнього, працюють досить добре в умовах доведених до ідеальних, коли об'єкти пошуку мають чітку форму без розмитих меж та досить добре їх можна вирізнити з поміж фону. Також слід зазначити, що виділені методи працюють досить повільно, що суттєво сповільнює процес сканування документів.

Враховуючи той факт, що умови пошуку прямокутних областей можуть бути досить різними, то для вирішення даної задачі найкраще підійде метод Віюлі-Джонса [16]. Цей алгоритм добре працює на об'єктах реального світу і має досить високу швидкість детектування. Суть алгоритму полягає у тому, що замість звичайного зображення використовується інтегральне зображення – зображення, кожен піксель якого представляє собою суму всіх пікселів які лежать лівіше і вище даного пікселя, включаючи сам піксель [17]. В якості ознак класифікації використовуються примітиви Хаара [18], які зазвичай є прямокутниками, розташовані вертикально, горизонтально або діагонально. Вони містять світлу і темну частини. Значення складається з різниці суми світлої та відповідно суми темної частини примітиву Хаара. Для обчислення різниці використовується інтегральне зображення, яке дозволяє обчислити різницю таких сум за сталий проміжок часу.

## 2.3 Аналіз методів покращення та визначення якості зображення для попередньої обробки зображення

Попередня обробка зображення є головним етапом для покращення якості сканованих документів. На даному етапі має проводитись попередня обробка зображення, для того аби забезпечити кращу якість друку сканованих документів.

Під час аналізу доступних програмних продуктів, було визначено, що усі наявні продукти на ринку мають хоча б один механізм покращення якості зображення чи обробки його. Здебільшого це механізми:

- перетворення відсканованого документу до зображення у відтінках сірого;
- перетворення відсканованого документу у чорно-біле зображення;
- регулювання яскравості та контрастності;
- визначення якості зображення.

Тому як окремий етап сканування документів виділено етап покращення якості документів та як новий функціонал вирішено визначати чіткість зображення та використовувати його як кількісний показник якості фотографії.

### 2.3.1. Аналіз методів приведення відсканованого документу у відтінки сірого

Розглянемо деякі прості алгоритми перетворення зображення у відтінки сірого. Так як фотографії не можуть мати прозорості для кожного окремого пікселю, то відповідно зазначені алгоритми будуть враховувати перетворення зображення RGB до grayscale зображення. Існує 3 найбільш прості алгоритми перетворення зображення у відтінки сірого [23]:

- lightness метод, суть якого полягає у тому, що ми знаходимо середнє арифметичне максимального та мінімального компоненту із кольорів:

$$(\max(R, G, B) + \min(R, G, B))/2, \quad (2.1)$$

де  $R, G, B$  – це відповідно значення компонентів червоного, зеленого та синього кольорів.

- average метод, суть якого полягає у тому, що ми просто знаходимо середнє значення усіх компонентів пікселю:

$$(R + G + B)/3, \quad (2.2)$$

де  $R, G, B$  – це відповідно значення компонентів червоного, зеленого та синього кольорів.

- luminosity метод, який є більш чітким та витонченим методом, суть якого полягає у тому, що середнє значення формуються відповідно до людського сприйняття кольорів:

$$R * 0.21 + G * 0.72 + B * 0.07, \quad (2.3)$$

де  $R, G, B$  – це відповідно значення компонентів червоного, зеленого та синього кольорів.

Кожен із зазначених методів є по-своєму ефективний, але в результаті все одно буде отримане зображення у відтінках сірого. Приклад роботи, кожного із алгоритмів зображено на рисунках 2.12 та 2.13.

Original image



Рисунок 2.12 – Оригінальне зображення, яке перетворилось у відтінки сірого [23]

Lightness



Average



Luminosity



Рисунок 2.13 – Приклад роботи кожного із алгоритмів приведення зображення до відтінків сірого [23]

### 2.3.2. Аналіз методів приведення відсканованого документу у чорно-біле

Наступним можливим типом покращення якості зображення є перетворення зображення у чорно-біле. Найбільш простим варіантом перетворення зображення у чорно-біле є використання порогової функції для зображення у відтінках сірого [24]. Суть методу полягає у наступному: якщо значення відтінку сірого більше якогось порогу, то ми перетворюємо значення пікселю у чорний колір і навпаки у протилежному випадку:

$$f(x) = \begin{cases} 1, & x > \varepsilon \\ 0, & x \leq \varepsilon \end{cases} \quad (2.4)$$

де  $x$  – значення відтінку сірого, а  $\varepsilon$  – порогове значення.

В результаті виконання такої простої операції буде отримане зображення як на рисунку 2.14.

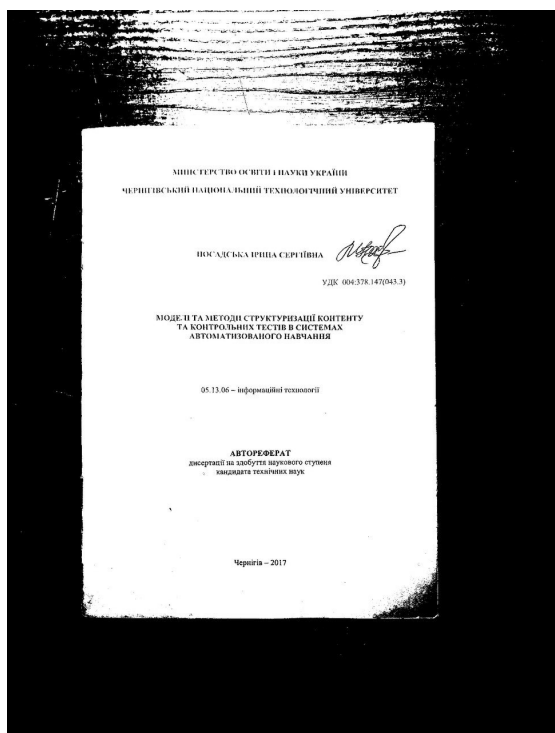


Рисунок 2.14 – Приклад зображення перетвореного у чорно-біле

Як бачимо перетворення зображення у чорно-біле є досить простим процесом, який не потребує значний математичних операцій.

### 2.3.3. Аналіз методів зміни яскравості зображення

Ще одним способом покращення якості зображення є зміна його яскравості. Щоб змінити яскравість зображення необхідно розкласти зображення на RGB компоненти та до кожної компоненти додати або відняти однакове значення яскравості [25], тобто:

$$R' = R \pm \text{brightness}, G' = G \pm \text{brightness}, B' = B \pm \text{brightness}, \quad (2.5)$$

де R, G, B – це відповідно значення компонентів червоного, зеленого та синього кольорів.

Але під час розрахунку нових значень для кожної складової кольору слід враховувати, що значення повинне знаходитись в межах від 0 до 255 або від 0 до 1 в залежності від проміжку, в якому знаходяться значення. Також слід не забувати про те, що якщо ми обмежуємо значення пікселів в межах  $[0, 255]$ , то наприклад, якщо ми додамо до червоної компоненти  $R = 100$  значення яскравості 200, то значення буде рівне 255, а при відніманні того ж значення ми вже отримаємо  $R = 55$  замість 100. Тому необхідно пам'ятати про такі випадки та зберігати початкове значення, якщо необхідно.

#### 2.3.4. Аналіз методів зміни контрастності зображення

Ще одним можливим варіантом корекції документів є зміна контрасту. Для того, щоб змінити значення контрасту документа необхідно підрахувати коефіцієнт корекції контрасту [26]:

$$F = \frac{259(C+255)}{255(259-C)}, \quad (2.6)$$

де  $C$  – це значення зміни контрасту в межах від -128 до 128.

Після підрахунку коефіцієнту корекції контрасту необхідно вирахувати нове значення для кожної компоненти кольору:

$$R' = F(R - 128) + 128, \quad (2.7)$$

де  $R'$  – це нове значення червоної компоненти.

### 2.3.5. Аналіз методів аналізу якості зображення

Наступним важливим етап обробки зображення є визначення його якості, а саме визначення чіткості зображення. Одним із доступних варіантів визначення чіткості зображення є дисперсія Лапласа [27]. Для отримання значення дисперсії Лапласа необхідно пройти по простому одно-канальному зображенню, наприклад, зображенню у відтінках сірого чи чорно-білому зображенню, за допомогою ядра згортки Лапласа (рисунок 2.15).

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

Рисунок 2.15 – ядро Лапласа

Наступним кроком необхідно порахувати дисперсію від отриманої відповіді. Якщо дисперсія опускається нижче заздалегідь визначеного порогу, то зображення вважається розмитим; інакше зображення не розмите. Значення порогу може вибиратись експериментально в залежності від потреб.

Відповідно в результаті виконання всіх раніше зазначених операцій покращення зображення буде отримано документ прийнятної якості, а за рахунок визначення чіткості зображення в процесі сканування будуть відкинуті усі документи незадовільної якості.

## 2.4 Аналіз методів класифікації зображень для етапу сканування документів та класифікації за типом

Для класифікації зображень необхідно вирішити задачу розпізнавання зображень, ідеальним інструментом для вирішення якої є нейронні мережі [28], так як немає чітких умов і формул для розпізнавання того чи іншого типу документу. Зазвичай розглядаючи нейронні мережі одразу згадують про повнозв'язні нейронні мережі із зворотнім поширенням похибки, але саме для вирішення задачі цей тип мережі є неефективним, бо процес навчання може зайняти велику кількість часу. Приведемо простий приклад розрахунків нижче.

Враховуючи, що зображення це набір пікселів різного кольору, то ідеальним буде визначення зображення як матриці розміром  $n$  на  $n$ . Так як сучасні телефони дозволяють робити фотографії розміром до 4032 пікселів на 3024 пікселів (максимальне розширення фотографії для мобільних пристроїв на базі iOS), а такий розмір матриці матиме трохи більше 12 млн вхідних значень, то досить розумним є обмеження входів хоча б до матриці 640 на 640, що в результаті рівне 409600 входів. Наступним є визначення кількості прихованих шарів та кількості нейронів на даних шарах. Зазвичай прийнято використовувати більше нейронів на прихованих шарах [29]. Навіть якщо взяти всього один прихований шар із 500 тисячами нейронів, а на виході мати хоча б 10 типів документів, то в результаті необхідно буде налаштувати трохи менше ніж 204 млрд зв'язків, що навіть з досить швидким комп'ютером потребує значну кількість часу для 1000 навчальних прикладів, хоча їх може бути набагато більше.

Тому даний тип нейронних мереж є зовсім не підходящим для вирішення даної задачі. Але для вирішення даної задачі із розпізнаванням зображень досить добре справляється людський мозок та мозок інших тварин, наприклад, мавп. Праця Г'юбела та Візела в 1950-ті та 1960-ті роки показала, що зорові кори котів та мавп містять нейрони, які окремо реагують на маленькі ділянки

зорового поля. За умови, що очі не рухаються, область зорового простору, в межах якої візуальний стимул впливає на збудження одного нейрону, відома як його рецептивне поле [30]. Саме починаючи із даних досліджень було розроблено інший тип нейронних мереж, який базується на сприйнятті різних частин зображення. Даний тип нейронних мереж має назву CNN – convolutional neural network або згорткові нейронні мережі.

Згорткова нейронна мережа - спеціальна архітектура штучних нейронних мереж, запропонована Яном Лекуном в 1988 році [31] і націлена на ефективне розпізнавання образів, входить до складу технологій глибокого навчання (англ. Deep learning). Основна ідея згорткових нейронних мереж полягає в чергуванні згорткових шарів і субдискретизуючих шарів. Структура мережі – односпрямована (без зворотних зв'язків), принципово багатшарова [32]. Основну суть навчання нейронної мережі зображено на рисунку 2.16.

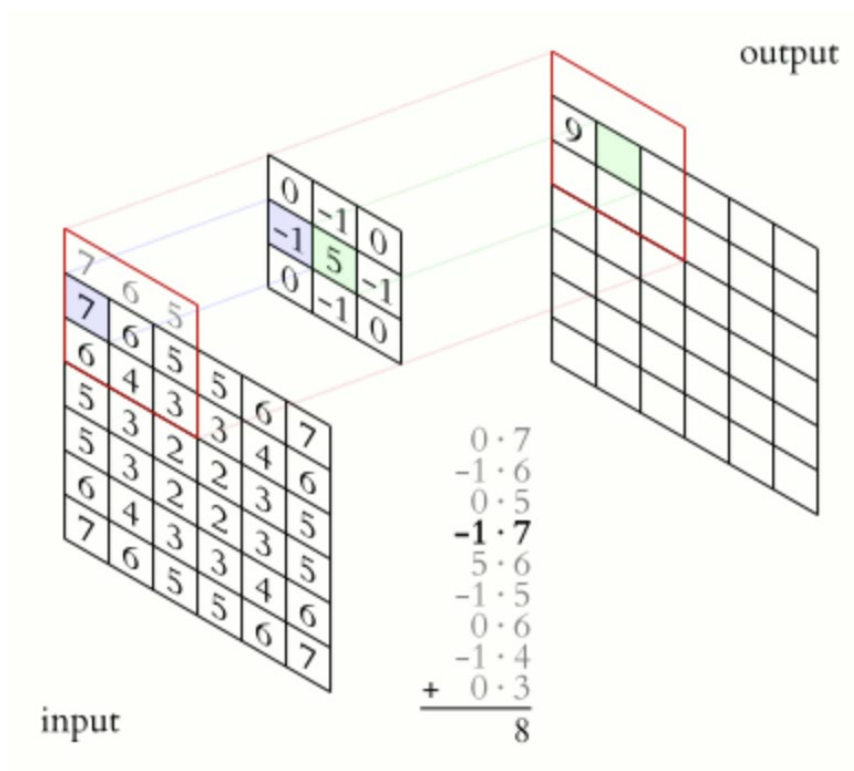


Рисунок 2.16 – Суть навчання згорткової нейронної мережі із використання згорткового ядра [33]

В результаті використання операції згортки разом із субдискретизацією досягається зменшення розрахункових операцій без втрати якості розпізнавання об'єктів. Типова структура згорткової нейронної мережі зображена на рисунку 2.17.

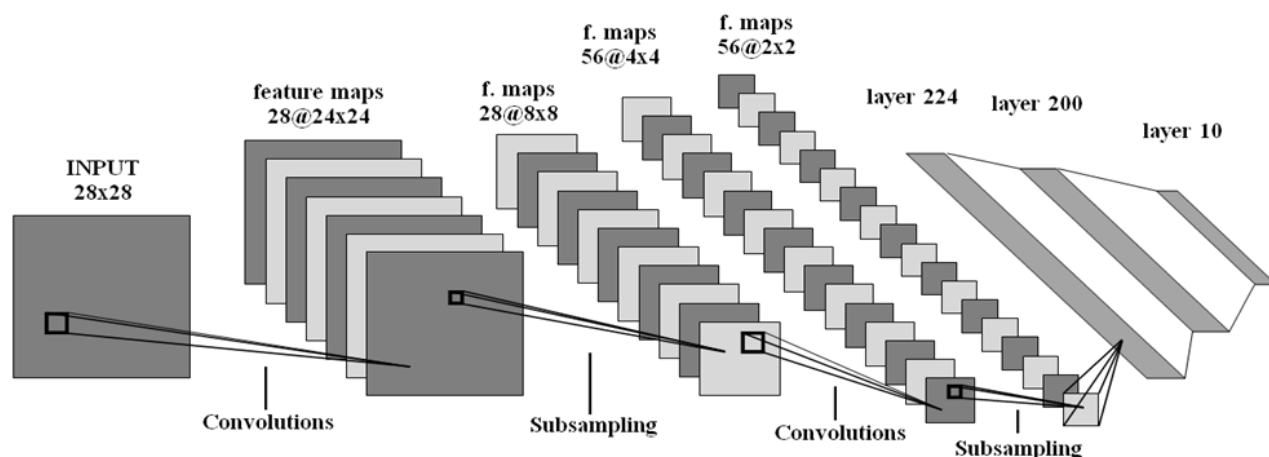


Рисунок 2.17 – Типова структура згорткової нейронної мережі [34]

Чорний квадрат зображений на рисунку є ядро, приклад використання якого зображено на рисунку 2.16. Як бачимо чергуючи таким чином згорткові шари із субдискретизуючими, на яких частіше всього виконується операція пошуку максимуму або середнього по декільком сусіднім нейронам [35].

Архітектура згорткових мереж включає в себе різні шари, що виконують перетворення вхідного зображення та збирають характеристики про це зображення. Кожен шар згорткової нейронної мережі виконує свою функцію. Згорткова нейронна мережа може включати в себе наступні шари [54]:

- Згорткові шари. Кожен шар містить набори ядер для навчання мережі. Кожен фільтр здійснює згортку по ширині та висоті під час прямого проходу по зображенню. Перші згорткові шари використовуються для визначення простих особливостей, такі як межі об'єктів, ліній та навіть кутів. Усі наступні згорткові шари вивчають складніші особливості – частини об'єктів.

- Шари підвибірки або шари субдискретизації. Вони зазвичай використовують нелінійні функції для створення підвибірки, найбільш популярною серед яких є функція максимізації. Дана функція розділяє вхідне зображення на набір невеликих прямокутників, які не перекривають одне одного (блок з 4 прямокутників по 4 елементи), і для кожного такого прямокутника знаходиться максимум. Ідея використання даного шару полягає в тому, що необхідно запам'ятати положення знайденої ознаки, а не факт її пошуку. Шар підвибірки поступово скорочує просторовий розмір представлення для того, щоб зменшити кількість параметрів для навчання.

- Активаційні шари – шари, які імітують поведінку активації як у аксона нейрона. Здебільшого в згорткових мережах використовують функцію активації ReLU [36].

- Шари виключення, які випадковим чином обнуляють значення активації. Даний хід дозволяє впевнитися, що мережа надає правильну класифікацію, навіть якщо деякий набір активацій виключено або було втрачено.

- Повнозв'язні шари, які призначені для зв'язку між шарами, де всі виходи попереднього шару пов'язані з усіма входами повнозв'язних шарів.

Існує значна кількість варіантів визначення кількості параметрів для кожного шару мережі. Якщо  $n_i$  ( $i = 0, \dots, n$ ) дорівнює кількості вихідних карт ознак шару  $i$ , де при  $i = 0$  це вхідний шар та усі фільтри мають розмір  $3 \times 3$ , тоді розрахунок числа параметрів для наступних шарів мережі виглядає так (3.1):

$$N = \sum_{i=1}^k ((n_{i-1} * 3^2 + 1) * n_i), \quad (2.8)$$

де  $N$  – число параметрів.

Однією із досить великих задач для вирішення є пошук документів на фотографії та їх подальша класифікація. В результаті вирішення даної задачі буде розширено основний функціонал систем сканування за рахунок додаткової класифікації документів за типом на фотографії та можливості сортування документів за даною властивістю.

В результаті використання саме такого типу нейронної мережі дозволяє значно покращити якість розпізнавання об'єктів, а також значно зменшує кількість зв'язків, які треба навчити при цьому не втрачаючи якість розпізнавання, а навіть збільшуючи її.

## 2.5 Вибір засобів розробки

Засоби розробки є ключовим елементом для вирішення поставлених завдань, адже від засобів розробки залежить час та простота реалізації системи для сканування та класифікації документів на фотографії.

Визначившись із типом нейронної мережі необхідно визначити, які саме інструменти краще використати для:

- пошуку прямокутних областей на фотографії;
- навчання, побудови та використання нейронної мережі.

### 2.5.1 Порівняльний аналіз засобів реалізації методу Віоли-Джонса для пошуку областей із ймовірними документами

Метод Віоли-Джонса є найбільш ефективним поєднанням швидкості та якості пошуку прямокутних областей на фотографії як було визначено у першому розділі. Даний метод необхідно використати на етапі підготовки зображення до аналізу та класифікації.

Одним із приклад реалізації методу Віоли-Джонса є бібліотека з відкритим програмним кодом OpenCV [20]. Для того аби навчити систему знаходити межі документів необхідна вибірка фотографій з документами в реальних умовах і вибірка з фотографіями без документів. Під час навчання

використовується так зване «ковзке вікно» [21], яке поступово рухається по зображенню. Змінюючи розмір даного вікна можна знаходити об'єкти різних розмірів.

Звісно існують вже навчені моделі, які використовуються бібліотекою OpenCV для пошуку прямокутних областей, проте для того аби навчитися ідентифікувати межі декількох документів за допомогою OpenCV необхідна значна кількість прикладів та часу для навчання. Тому на противагу відкритої бібліотеки OpenCV існує бібліотека Vision від компанії Apple [22]. Основними недоліками даної бібліотеки є лише закритість програмного коду, що забороняє будь-які розширення наявного функціоналу та відсутність будь-якої інформації про особливості реалізації пошуку прямокутних областей.

Для того аби визначити, яка ж із даних бібліотек краще працює необхідно провести порівняння. Порівняння ефективності використання бібліотек пропонується здійснювати за такими ознаками:

- ідентифікація декількох документів одночасно;
- кількість використовуваної операційної пам'яті;
- завантаження центрального процесора (ЦП);
- якість пошуку документів на фотографії.

Порівняння здійснюється шляхом аналізу потокового відео. Результати порівняння наведені в таблиці 2.1 та додатку А.

Таблиця 2.1 – Порівняння OpenCV та Vision для відео якості 1080p 30 кадрів за секунду

| Кількість об'єктів, шт | Бібліотека       | Початкова кількість ОП, Мб | ОП, Мб | Завантаженість ядер ЦП, % | Якість |
|------------------------|------------------|----------------------------|--------|---------------------------|--------|
| 3                      | Vision, Fast     | 35                         | 60-65  | 75-80                     | 3      |
| 3                      | Vision, Accurate | 35                         | 80-90  | 80-90                     | 2      |

Таблиця 2.1 – Продовження

|   |                  |    |         |         |   |
|---|------------------|----|---------|---------|---|
| 3 | OpenCV           | 50 | 120-130 | 140-150 | 2 |
| 1 | Vision, Fast     | 35 | 55-60   | 75-80   | 5 |
| 1 | Vision, Accurate | 35 | 60-80   | 80-85   | 4 |
| 1 | OpenCV           | 50 | 110-115 | 110-120 | 4 |

Враховуючи зазначені вище результати можна зробити висновок, що бібліотека Vision незважаючи на виявлені недоліки все одно показує кращі результати, до яких ще можна додати їх досить невеликий розмір: розмір застосунку із бібліотекою Vision – 15 Мб, а застосунку із бібліотекою OpenCV – 75 Мб.

Враховуючи результати порівняльної характеристики було вирішено використовувати саме бібліотеку Vision для розробки кінцевого застосунку.

### 2.5.2 Аналіз інструментів побудови та навчання згорткової нейронної мережі

Враховуючи те, що фінальний застосунок буде розроблятися для мобільних пристроїв на базі операційної системи iOS, було вирішено використати готові інструменти із бібліотек, які надає компанія Apple.

Для того аби навчити мобільний пристрій розпізнавати документи різних типів, необхідно тренувати згорткову нейронну мережу. З цією метою компанія Apple розробили бібліотеку Core ML [37, 38] та інструмент Create ML [39, 40], який дозволяє навчати моделі для згорткової нейронної мережі, яка використовується в бібліотеці Vision [41].

Процес навчання нейронної мережі за допомогою інструменту Create ML виглядає досить простим і зображено на рисунку 2.18. Суть навчання базується на підготовці і розділенні вхідних даних за типами документів. Дані приклади із документами відповідно до прикладів створених

компанією Apple краще зменшити до розмірів, не більше ніж 299 на 299 пікселів, для пришвидшення навчання.

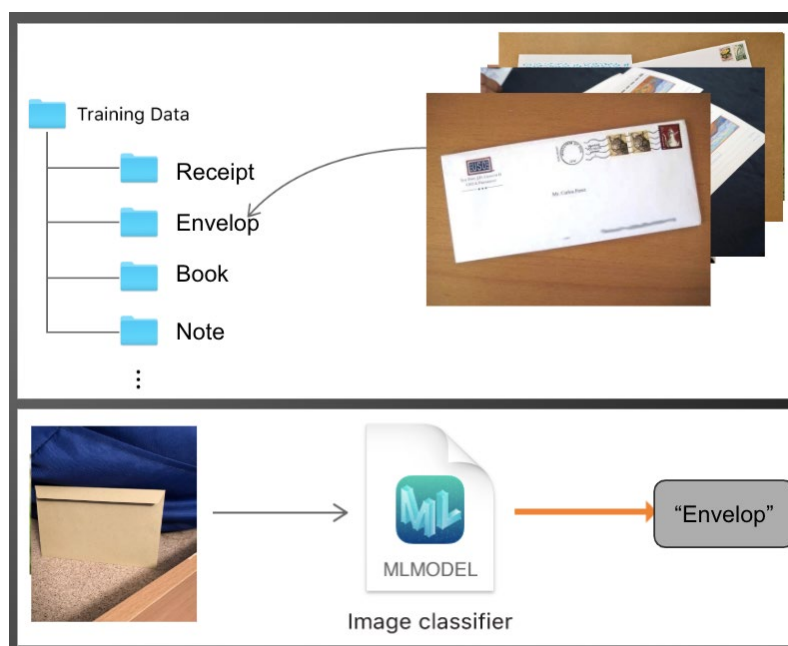


Рисунок 2.18 – Навчання згорткової мережі на наборах даних за допомогою Core ML та інструменту Create ML [39]

Процес використання навченої моделі (файлу із розширенням mlmodel) є досить простим у використанні і представлено на рисунку 2.19.



Рисунок 2.19 – Використання навченої моделі в застосунку

Отже, в результаті використання інструментів Core ML та Create ML, які є необхідними для створення та навчання згорткової нейронної мережі буде можливо визначати типи документів з фотографії.

## 2.6 Висновки до другого розділу

У даному розділі було представлено розроблену технологію для пошуку та ідентифікації документів на фотографії. Було розглянуто різні методи, технології та бібліотеки для пошуку прямокутних областей на фотографії, покращення якості зображення та визначення типів документів на фотографії.

Було визначено, що бібліотека Vision є ідеальним рішенням для вирішення даної задачі, бо поєднує в собі відмінну якість пошуку документів на фотографії, високу швидкість роботи та малий розмір при використанні в застосунку у порівнянні із найбільш популярною бібліотекою з відкритим кодом OpenCV.

Під час розгляду методів покращення зображення було обрано найбільш популярні методи для приведення зображення до відтінків сірого чи до чорно-білого. Але особливістю розробленої технології є визначення розмитості зображення, що дозволяє збільшити ймовірність правильної ідентифікації типу документу та зменшує кількість документів поганої якості.

В результаті аналізу технологій визначення типів документів було обрано згорткові нейронні мережі, які пристосовані для роботи із зображеннями і процес навчання яких є досить простим.

## РОЗДІЛ 3. ОПИС РЕАЛІЗАЦІЇ ЗАСТОСУНКУ

### 3.1 Визначення архітектури згорткової нейронної мережі

Перш ніж визначати архітектуру згорткової нейронної мережі необхідно визначати, які класи документів нейронна мережа повинна класифікувати. В результаті аналізу ринку застосунків для сканування документів та детального аналізу їх можливостей та особливостей було визначено, що система повинна відрізняти наступні типи документів:

- газети;
- журнали;
- книжки;
- друковані тексти;
- рукописні тексти;
- квитки для проїзду в громадському транспорті;
- візитівки;
- дисконтні картки;
- папірці з нотатками;
- чеки з магазинів;
- конверти.

Для деяких типів документів є навіть окремі функції в застосунках, наприклад, розмір документів для друку в застосунку HP Smart та сканування книжок в застосунку FineScanner.

Архітектура згорткової мережі для даної роботи буде включати усі шари зазначені в другому розділі:

- згорткові шари;
- шари підвибірки або шари субдискретизації;
- активаційні шари;
- шари виключення, які випадковим чином обнуляють значення активації;

- повнозв'язні шари, які призначені для зв'язку між шарами

Так відповідно у кінці кожної згортки використовується функція активації ReLU [40] в усіх згорткових шарах, окрім останнього повнозв'язного, в якому застосовується функція softmax [43]. Перед останнім згортковим шаром використовуються шари виключення із dropout [44], значення якого дорівнює 0.5. Детальну архітектуру згорткової нейронної мережі представлено у таблиці 3.1.

Таблиця 3.1 – Архітектура згорткової нейронної мережі

| Назва шару             | Ядра /<br>Нейрони | Функції<br>активації | Канали | Вихідний<br>розмір |
|------------------------|-------------------|----------------------|--------|--------------------|
| Згортковий шар 1       | 3x3               | ReLU                 | 32     | 32x32              |
| Шар виключення 1 (20%) |                   |                      | 32     | 32x32              |
| Згортковий шар 2       | 3x3               | ReLU                 | 32     | 32x32              |
| Шар підвибірки 1       | 2x2               |                      | 32     | 16x16              |
| Згортковий шар 3       | 3x3               | ReLU                 | 64     | 16x16              |
| Шар виключення 2 (20%) |                   |                      | 64     | 16x16              |
| Згортковий шар 4       | 3x3               | ReLU                 | 64     | 16x16              |
| Шар підвибірки 2       | 2x2               |                      | 64     | 8x8                |
| Згортковий шар 5       | 3x3               | ReLU                 | 128    | 8x8                |
| Шар виключення 3 (20%) |                   |                      | 128    | 8x8                |
| Згортковий шар 6       | 3x3               | ReLU                 | 128    | 8x8                |
| Шар підвибірки 3       | 2x2               |                      | 128    | 4x4                |
| Розгалуження           | 2048              |                      |        |                    |
| Шар виключення 4 (20%) | 2048              |                      |        |                    |
| Повнозв'язний шар 1    | 1024              | ReLU                 |        |                    |

Таблиця 3.1 – Продовження

|                        |      |         |  |  |
|------------------------|------|---------|--|--|
| Шар виключення 5 (20%) | 1024 |         |  |  |
| Повнозв'язний шар 2    | 512  | ReLU    |  |  |
| Шар виключення 6 (20%) | 512  |         |  |  |
| Повнозв'язний шар 3    | 12   | softmax |  |  |

Як ми бачимо на виході буде 12 класів документів, які безпосередньо класифікуються. Кожний із виходів буде вказувати на ймовірність приналежності зображення до одного із класів документів.

Для побудови даної моделі використовується інструмент turicreate [45, 46], який дозволяє побудувати модель та потім конвертувати її для подальшого навчання та використання у інструменті Create ML та для мобільних пристроїв на базі iOS. Відповідно за допомогою даного інструменту модель конвертується у файл із розширенням “mlmodel”, який вже підходить для подальшого навчання.

### 3.2 Навчання згорткової нейронної мережі

Після визначення архітектури згорткової нейронної мережі відбувається процес навчання. Для навчання згорткової нейронної мережі було обрано раніше описаний інструмент Create ML [39] (рисунок 3.1).

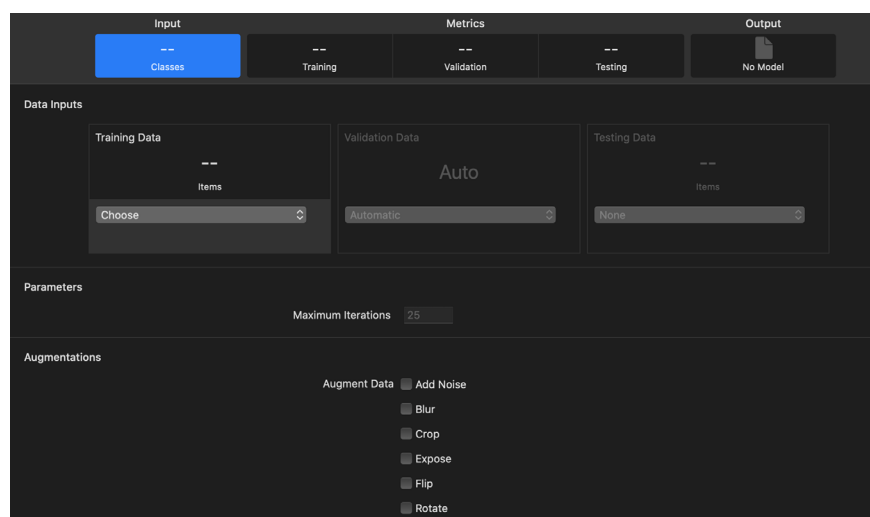


Рисунок 3.1 – Приклад інтерфейсу Create ML

Даний інструмент дозволяє визначити кількість ітерацій для навчання згорткової нейронної мережі; визначити вхідні зображення для навчання, валідації та тестування, а у випадку відсутності валідаційних та зображень для тестування автоматично розділяє вибірку та виділяє 10% на тестування чи валідацію. Також даний інструмент надає можливість наповнити вхідні дані додатковими прикладами за рахунок додавання шумів, розмитості, вирізання частин зображення, різного рівня чіткості, поворотів та переворотів зображення. Наприклад, додавши останніх чотири модифікації кількість вхідних даних збільшилися у 200 разів, але і час навчання збільшився у 240 разів.

Як додаткові метрики даний інструмент надає дані про правильність визначення типів документів під час валідації і тестування та надає фінальні дані валідації і тестування навченої нейронної мережі. Саме ці дані будуть використовуватись під час вибору найкращої навченої нейронної мережі.

Для навчання нейронної мережі було зібрано 1544 навчальних зображення [47] для 12 класів документів та 168 зображень для автоматичного тестування навченості нейронної мережі. Зібрані зображення було підготовлено для навчання:

- кожне зображення було зменшено до 299 пікселів по одному із розмірів та пропорційно по другому розміру;
- кожне зображення було зменшено до 600 пікселів по одному із розмірів та пропорційно по другому розміру;
- було взято сирі не підготовлені дані.

Такі розміри було обрано по причині того, що розмір вхідного зображення, який використовує Create ML є 299 на 299 пікселів, тому було необхідно визначити як швидко відбувається навчання моделі в залежності від розміру вхідних зображень та як це впливає на якість навчання.

Навчання проводилося 4 різними методами:

- 100 ітерацій без модифікацій;
- 1000 ітерацій без модифікацій;
- 1000 ітерацій з 4 модифікаціями (crop, expose, flip, rotate);
- 1000 ітерацій з 2 модифікаціями (flip, rotate).

Саме такі модифікації були обрані по причині того, що вони найменше псують якість модифікованих зображень. Результати навчання представленні у таблицях 3.2 – 3.4.

Таблиця 3.2 – Порівняння якості розпізнавання документів під час навчання на зображеннях 299 на 299 пікселів

|                                                                             | 100 ітерацій<br>без<br>модифікацій | 1000 ітерацій<br>без<br>модифікацій | 1000 ітерацій<br>з 4 типами<br>модифікацій | 1000 ітерацій<br>з 2 типами<br>модифікацій |
|-----------------------------------------------------------------------------|------------------------------------|-------------------------------------|--------------------------------------------|--------------------------------------------|
| Час навчання                                                                | 1 хв 40 сек                        | 1 хв 44 сек                         | 9 год 7 хв                                 | 1 год 7 хв                                 |
| Результат<br>опрацювання<br>вхідних даних,<br>% правильної<br>ідентифікації | 100                                | 100                                 | 47                                         | 100                                        |

Таблиця 3.2 – Продовження

|                                                                       |                |                   |                   |                   |
|-----------------------------------------------------------------------|----------------|-------------------|-------------------|-------------------|
| Валідація даних                                                       | 90             | 92                | 78                | 86                |
| Тестування навченості автоматично, % (правильні / загальна кількість) | 82 (137 / 168) | 83 (139 / 168)    | 77 (129 / 168)    | 86 (144 / 168)    |
| Тестування навченості вручну, % (правильні / загальна кількість)      | 79 (237 / 300) | 77.25 (232 / 300) | 71,41 (214 / 300) | 80.75 (242 / 300) |

Таблиця 3.3 – Порівняння якості розпізнавання документів під час навчання на зображеннях 600 на 600 пікселів

|                                                                 |                              |                               |                                      |                                      |
|-----------------------------------------------------------------|------------------------------|-------------------------------|--------------------------------------|--------------------------------------|
|                                                                 | 100 ітерацій без модифікацій | 1000 ітерацій без модифікацій | 1000 ітерацій з 4 типами модифікацій | 1000 ітерацій з 2 типами модифікацій |
| Час навчання                                                    | 1 хв 44 сек                  | 4 хв 28 сек                   | 9 год 20 хв                          | 1 год 14 хв                          |
| Результат опрацювання вхідних даних, % правильної ідентифікації | 100                          | 100                           | 43                                   | 100                                  |
| Валідація даних                                                 | 92                           | 93                            | 81                                   | 95                                   |
| Тестування навченості, % (правильні / загальна кількість)       | 86 (144 / 168)               | 86 (144 / 168)                | 74 (124 / 168)                       | 85 (143 / 168)                       |

Таблиця 3.3 – Продовження

|                                                                  |                   |                   |                   |                   |
|------------------------------------------------------------------|-------------------|-------------------|-------------------|-------------------|
| Тестування навченості вручну, % (правильні / загальна кількість) | 79.25 (238 / 300) | 79.75 (239 / 300) | 71,83 (216 / 300) | 80.75 (250 / 300) |
|------------------------------------------------------------------|-------------------|-------------------|-------------------|-------------------|

Таблиця 3.4 – Порівняння якості розпізнавання документів під час навчання на зображеннях оригінальної розмірності

|                                                                  | 100 ітерацій без модифікацій | 1000 ітерацій без модифікацій | 1000 ітерацій з 4 типами модифікацій | 1000 ітерацій з 2 типами модифікацій |
|------------------------------------------------------------------|------------------------------|-------------------------------|--------------------------------------|--------------------------------------|
| Час навчання                                                     | 3 хв 1 сек                   | 3 хв 7 сек                    | 11 год 28 хв                         | 43 хв 51 сек                         |
| Результат опрацювання вхідних даних, % правильної ідентифікації  | 100                          | 100                           | 52                                   | 100                                  |
| Валідація даних                                                  | 92                           | 83                            | 86                                   | 90                                   |
| Тестування навченості, % (правильні / загальна кількість)        | 85 (143 / 168)               | 85 (143 / 168)                | 68 (114 / 168)                       | 85 (143 / 168)                       |
| Тестування навченості вручну, % (правильні / загальна кількість) | 86.36 (238 / 300)            | 86.9 (239 / 300)              | 70,45 (216 / 300)                    | 91.63 (250 / 300)                    |

Як ми бачимо із результатів в таблицях 3.2 – 3.4 найкращі результати було отримано під час навчання на більшій кількості ітерацій та з двома додатковими модифікаціями. Із результатів помітно, що часові витрати збільшилися в середньому у 20 - 30 разів в порівнянні з найбільш простим варіантом навчання, але якість розпізнавання значною мірою покращилась (рисунки 3.2 – 3.4).



Рисунок 3.2 – Графік залежності часу навчання та якості розпізнавання для різних моделей навчання згорткової нейронної мережі для зображень 299x299 пікселів

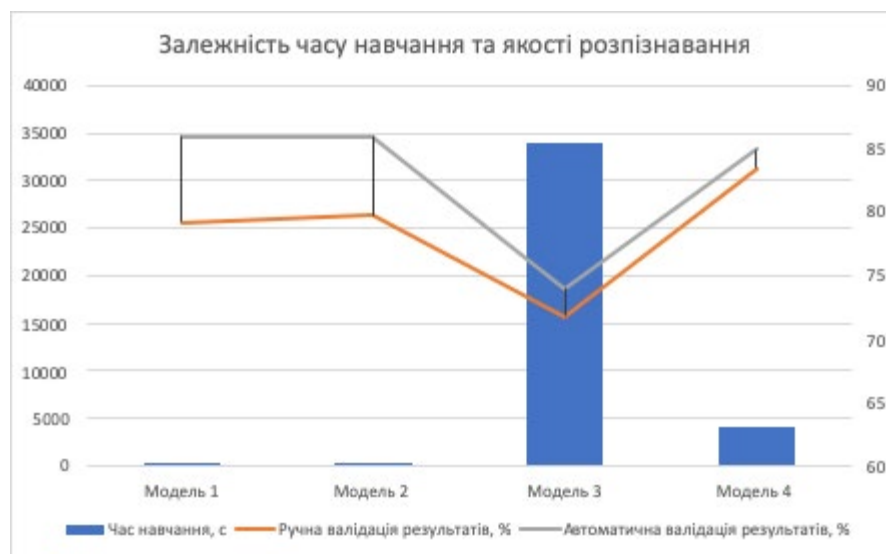


Рисунок 3.3 – Графік залежності часу навчання та якості розпізнавання для різних моделей навчання згорткової нейронної мережі для зображень 600x600 пікселів



Рисунок 3.4 – Графік залежності часу навчання та якості розпізнавання для різних моделей навчання згорткової нейронної мережі для оригінальних зображень

На даних графіка помітно, що найкращим способом навчання згорткової мережі є навчання на більшій кількості ітерацій – ідеально 1000 ітерацій та більше та доповнення вхідних даних за допомогою поворотів та переворотів. Даний спосіб навчання дозволить наблизитись до похибки у 10%, що є прийнятною похибкою для вирішення даної задачі. Із даних наведених у додатку Б також слідує, що найкращими вхідними даними для навчання згорткової мережі є саме сирі дані, які дозволяють виділити більше детальні характеристики кожного із об'єктів через те, що якість вхідного є кращою.

Також в ході аналізу отриманих результатів було вирішено розширити навчальні дані додатковими прикладами для деяких класів документів – квитки для проїзду в громадському транспорті, конверти, рукописні документи, журнали, паспорти, печатні документи. Даний крок обґрунтовано тим, що дані класи документів мали значну похибку, яка іноді дорівнювала більше 20%.

Після розширення бази навчальних даних була навчена згорткова нейронна мережа із 2076 вхідними зображеннями для 12 класів документів

за 1000 ітерацій та додатковими модифікаціями у вигляді поворотів та переворотів зображень. Результат навчання представлено в таблиці 3.5.

Таблиця 3.5 – Якість розпізнавання різного типу документів покращеною моделлю

| Тип документа                                | Кількість правильно розпізнаних документів, штук | Кількість невірно розпізнаних документів, штук | Відсоток правильних розпізнавань, % |
|----------------------------------------------|--------------------------------------------------|------------------------------------------------|-------------------------------------|
| Книга                                        | 47                                               | 3                                              | 94                                  |
| Квиток для проїзду в громадському транспорті | 50                                               | 0                                              | 100                                 |
| Візитівка                                    | 40                                               | 10                                             | 80                                  |
| Дисконтна картка                             | 45                                               | 5                                              | 90                                  |
| Конверт                                      | 47                                               | 3                                              | 94                                  |
| Рукописний текст                             | 48                                               | 2                                              | 96                                  |
| Журнал                                       | 44                                               | 6                                              | 88                                  |
| Газета                                       | 44                                               | 6                                              | 88                                  |
| Папірець з нотатками                         | 40                                               | 10                                             | 80                                  |
| Паспорт                                      | 45                                               | 5                                              | 90                                  |
| Друкований текст                             | 48                                               | 2                                              | 96                                  |
| Чек з магазину                               | 47                                               | 3                                              | 94                                  |

В результаті із отриманих показників можна побудувати наступний графік (рисунок 3.5).

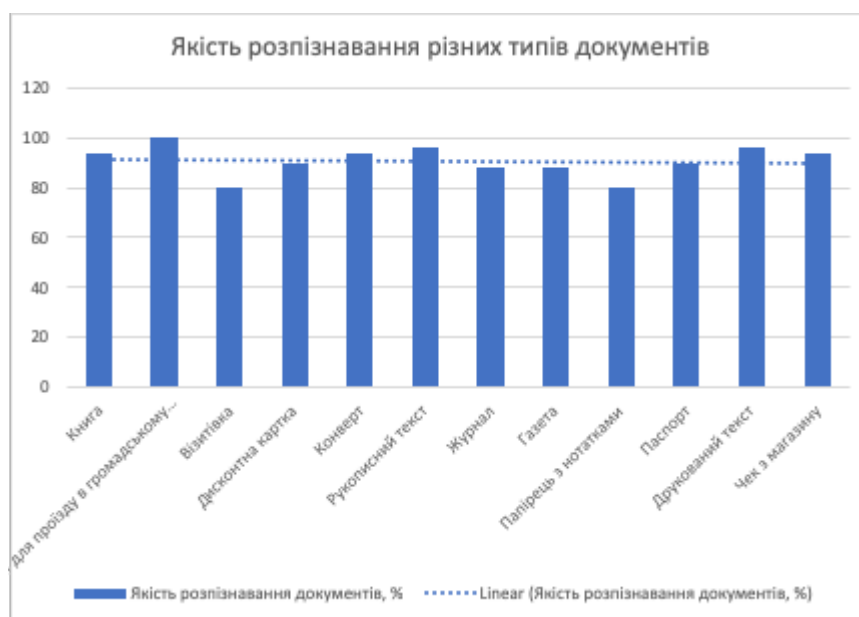


Рисунок 3.5 – Якість розпізнавання різних типів документів навченою нейронною мережею

З отриманих показників випливає, що в результаті навчання нейронної мережі було досягнуто похибки у 9.17 %, що є прийнятним показником похибки для даної задачі.

### 3.3 Архітектура застосунку для пошуку та ідентифікації документів на фотографії

Для реалізації розробленої технології пошуку та ідентифікації документів на фотографії було створено мобільний застосунок для пристроїв на базі iOS.

Для розробки застосунку було використано мову програмування Swift [49, 50] та бібліотеки Vision, Core ML для роботи з нейронними мережами. Також для роботи з графікою було використано вбудовану бібліотеку Core Graphics [51], яка дозволяє працювати із сирими даними та дозволяє малювати різні геометричні об'єкти на зображеннях.

Програмне забезпечення застосунку для пошуку та ідентифікації документів на фотографії у відповідності до запропонованої технології складається з таких підсистем:

- підсистема отримання зображення із можливим документом;
- підсистема підготовки зображення до аналізу та класифікації;
- підсистема сканування документу та класифікації за типом.

На рисунку 3.6 представлено загальну схему із усіма модулями, які використовуються для кожної з підсистем.

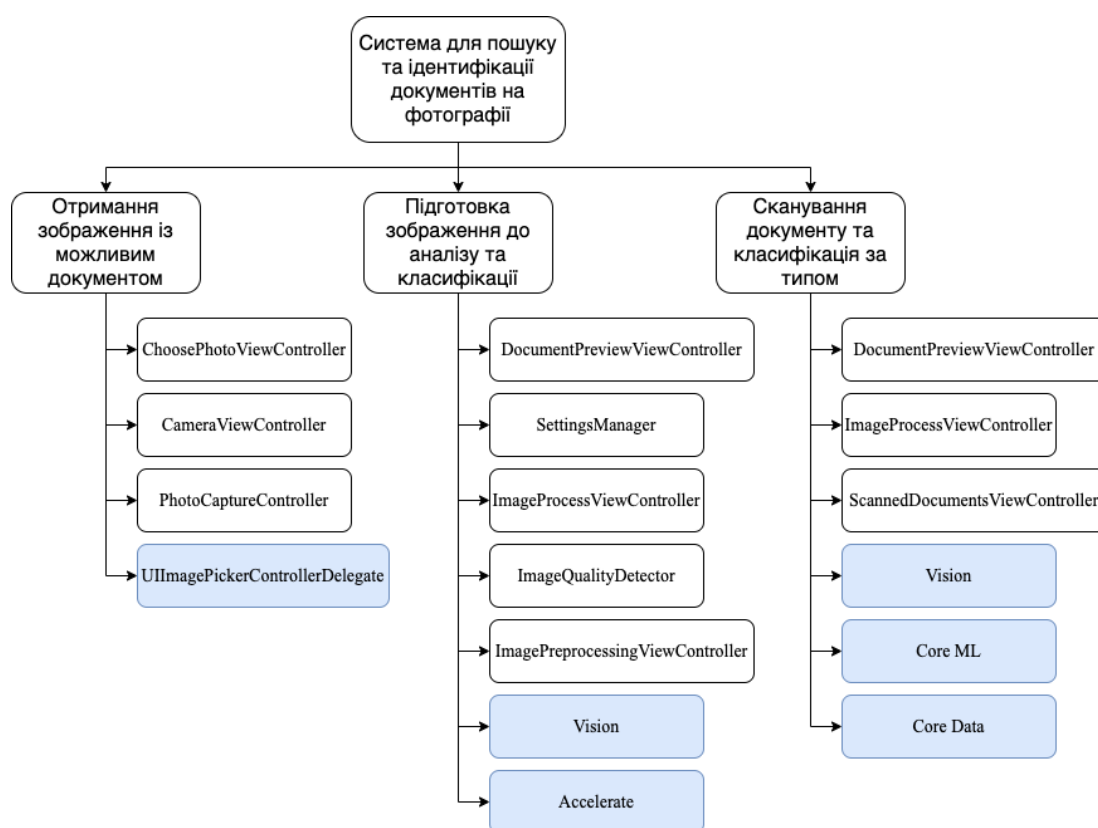


Рисунок 3.6 – Загальна архітектура програмного продукту

Для підсистеми отримання зображення із можливим документом було розроблено 3 модулі та використано один системний модуль (рисунок 3.7):

- ChoosePhotoViewController;
- UIImagePickerControllerControllerDelegate (системний модуль);
- CameraViewController;
- PhotoCaptureController.

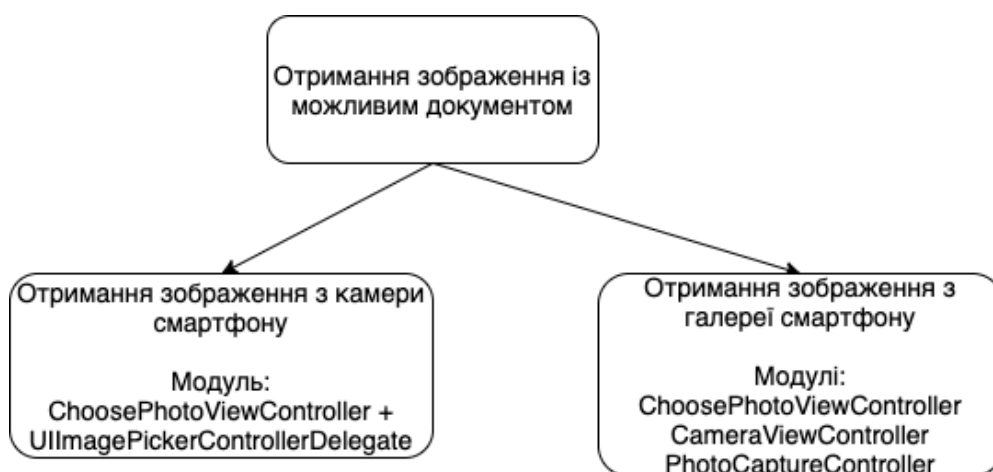


Рисунок 3.7 – Підсистема отримання зображення з можливим документом

Модуль ChoosePhotoViewController дозволяє вибрати метод отримання зображення – з галереї чи з камери – та перенаправляє отримане зображення за допомогою системного модулю UIImagePickerControllerControllerDelegate з галереї на подальшу обробку далі.

Модуль CameraViewController та PhotoCaptureController відповідають за отримання зображення з камери смартфону. PhotoCaptureController також відповідає за отримання сирих даних з камери, які потім використовуються для визначення якості зображення.

Для підсистеми підготовки зображення до аналізу та класифікації було розроблено 5 модулів та використано 2 системні бібліотеки:

- DocumentPreviewViewController;
- Vision (системна бібліотека);
- SettingsManager;
- ImageProcessViewController;
- ImageQualityDetector;
- Accelerate (системна бібліотека);
- ImagePreprocessingViewController;

На рисунку 3.8 зображено, які модулі використовуються для пошуку областей з імовірними документами, а які для попередньої обробки зображення.

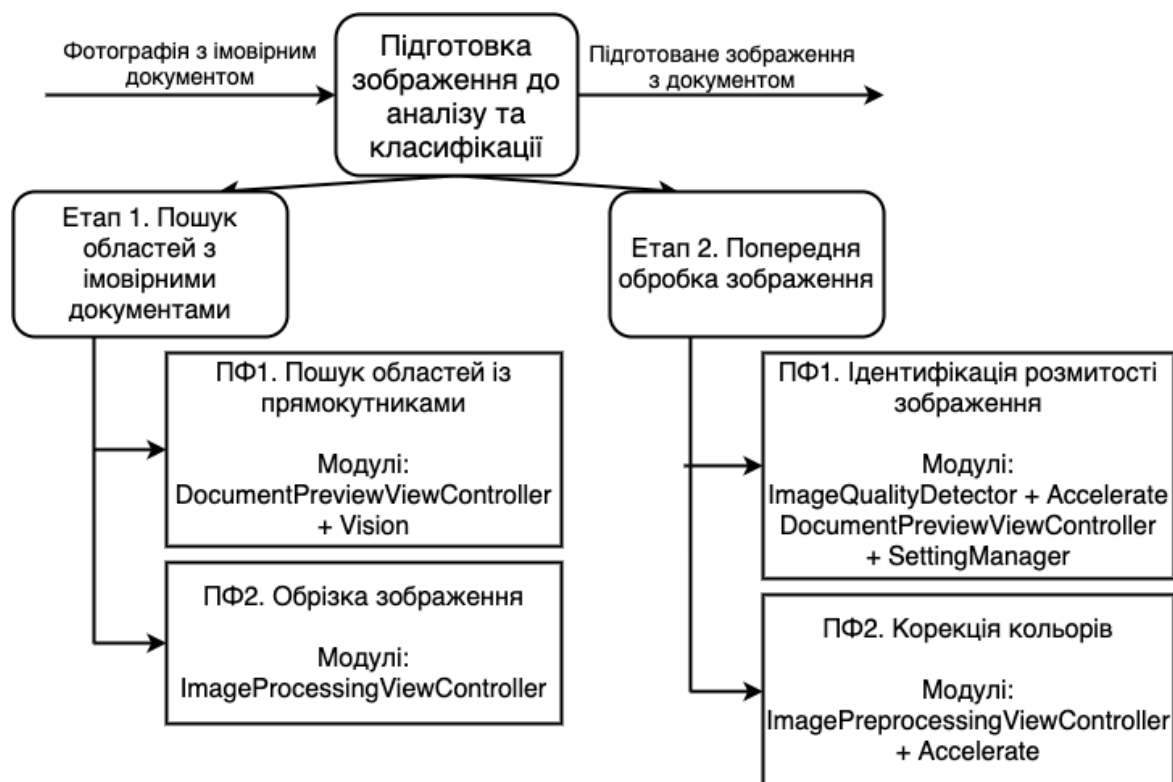


Рисунок 3.8 – Підсистема підготовки зображення до аналізу та класифікації

Модуль `DocumentPreviewViewController` отримує зображення із імовірним документом з модулів `CameraViewController` та `ChoosePhotoViewController`. Даний модуль за допомогою системної бібліотеки `Vision` займається пошуком прямокутних областей на фотографії.

Модуль `ImageProcessViewController` призначений для вирізання зображення за знайденими межами за допомогою модулю `DocumentPreviewViewController`.

Модуль `ImageQualityDetector` використовує сирі дані з фотографії отримані з модулю `PhotoCaptureController` та використовуючи модуль `Accelerate` визначає якість отриманого зображення за допомогою ядра Лапласа. Отриману якість модуль передає для відображення та обробки на модуль

DocumentPreviewViewController. В свою чергу модуль DocumentPreviewViewController забезпечує фільтрування зображень за допомогою модулю SettingsManager, який надає інформацію про мінімальну допустиму якість.

Модуль ImagePreprocessingViewController за допомогою модулю Accelerate дозволяє перетворити зображення у відтінки сірого, чорно-біле та дозволяє коригувати яскравість та контрастність зображення.

Для підсистеми сканування документа та класифікації за типом було розроблено 3 модулі, використано 2 системні бібліотеки і 1 системний модуль:

- DocumentPreviewViewController;
- Vision (системна бібліотека);
- Core ML (системна бібліотека);
- ImageProcessViewController;
- ScannedDocumentsViewController;
- Core Data (системний модуль).

На рисунку 3.9 зображено, які модулі використовуються для пошуку документів на фотографії, які для сканування та класифікації документа, а які для групування документів за типом.

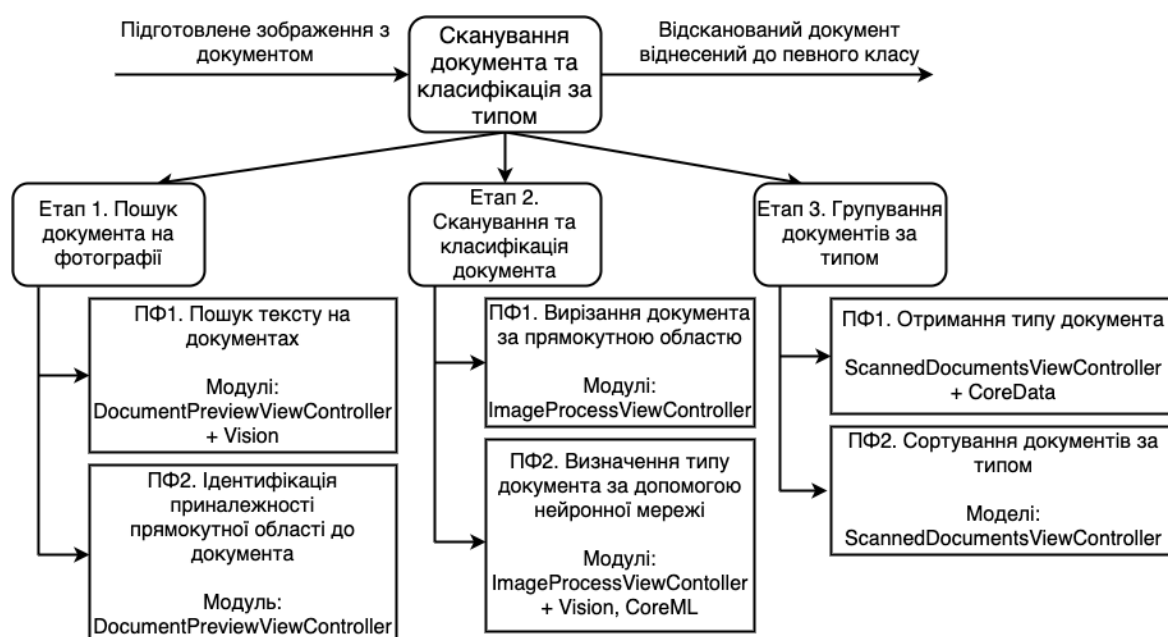


Рисунок 3.9 – Підсистема сканування документа та класифікації за типом

Окрім заявлених вище функцій, модуль `DocumentPreviewViewController` призначений також для пошуку тексту на прямокутних областях та ідентифікації приналежності прямокутної області до документа за рахунок наявності більше однієї групи слів. Пошук тексту відбувається за рахунок використання системної бібліотеки `Vision`.

Окрім заявлених вище функцій, модуль `ImageProcessViewController` займається фінальним скануванням документа та визначенням типу документка за допомогою системних бібліотек `Vision` та `Core ML`, які використовують навчену модель нейронної мережі.

Модуль `ScannedDocumentsViewController` призначений для відображення відсканованих документів, відомості про які зберігаються у базі даних. Для отримання інформації про документи та їх сортування, використовується системний модуль `Core Data` [48], який отримує інформацію про документи із бази даних `SQLite`.

Щоб зберігати інформацію про документи та впорядковувати їх, була створена досить проста база даних, яка зберігає інформацію про відскановані документи (рисунок 3.10).

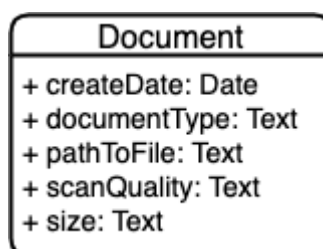


Рисунок 3.10 – Фізична модель бази даних

Для того аби впорядкувати відскановані документи необхідно зберігати інформацію про час їх сканування, тип документу, шлях до документу, якість сканованого документу, якщо доступна, та розмір відсканованого документу. Даної інформації цілком достатньо для сортування документів за різними параметрами – дата, тип документа та якість відсканованого документа.

### 3.4. Опис програмного інтерфейсу користувача застосунку для пошуку та ідентифікації документів на фотографії

Для побудови графічного інтерфейсу мобільного застосунку для пошуку та ідентифікації документів на фотографії було використано стандартну бібліотеку UIKit [52], яка дозволяє додавати елементи навігації, текст, зображення та різного роду кнопки чи вимикачі, які потрібні під час розробки графічного інтерфейсу.

Загальну схему інтерфейсу користувача із модулями, які представляють дані екрани представлено на рисунку 3.11.

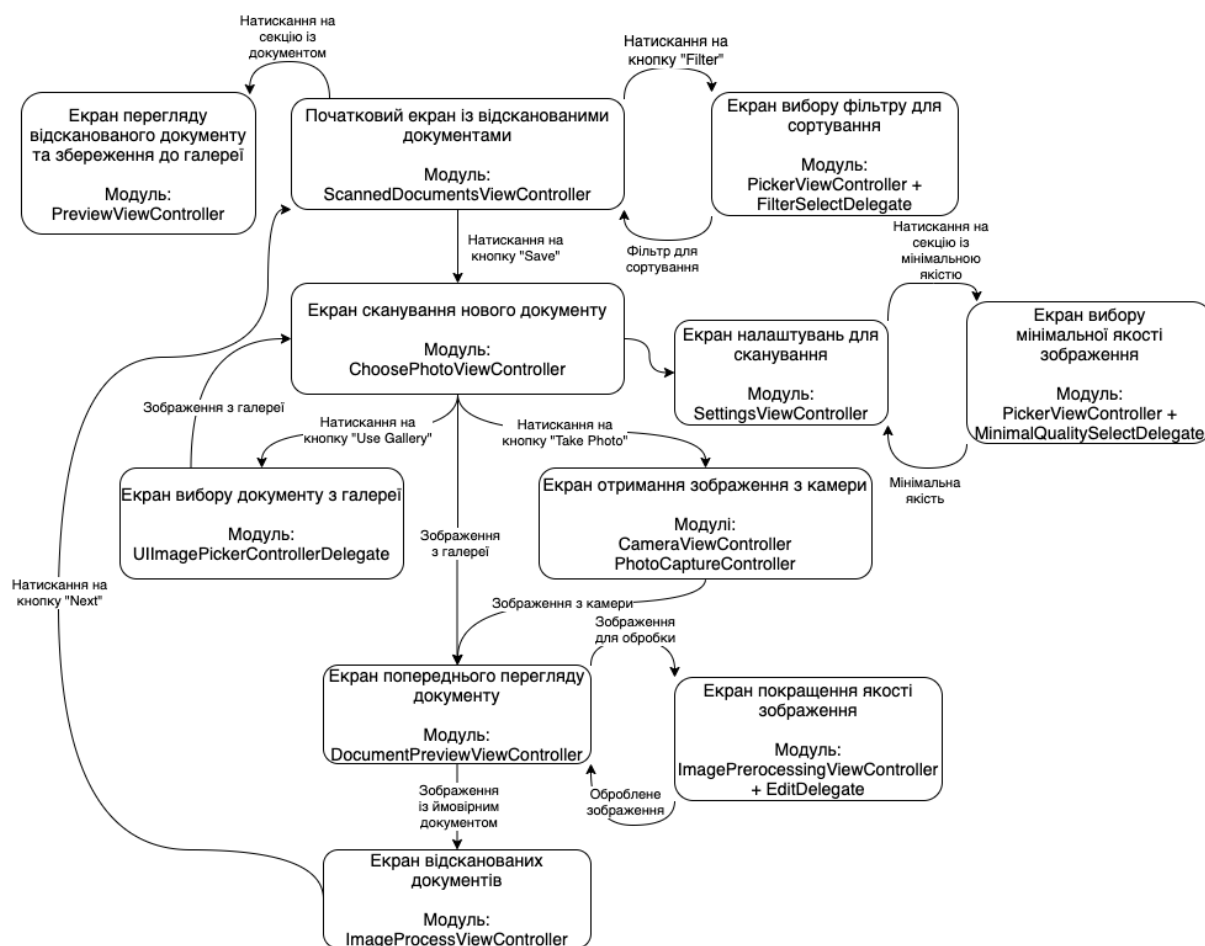


Рисунок 3.11 – Схема програмного інтерфейсу користувача

Коли користувач запускає застосунок, то його зустрічає екран із вже відсканованими документами (рисунок 3.12).

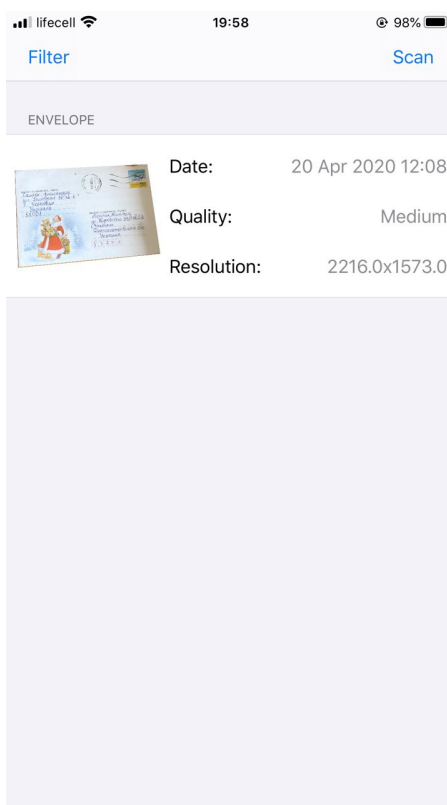


Рисунок 3.12 – Початковий екран застосунку

Як видно з даного екрану існує декілька кнопок, які ведуть на інші екрани:

- кнопка “Filter”, яка відкриває екран із трьома можливими фільтрами для сортування відсканованих документів – сортування за типом документів, датою створення та якістю зображення.
- кнопка “Scan”, яка відкриває наступний екран в застосунку, де користувач може обрати варіант завантаження зображення для роботи (рисунок 3.13).

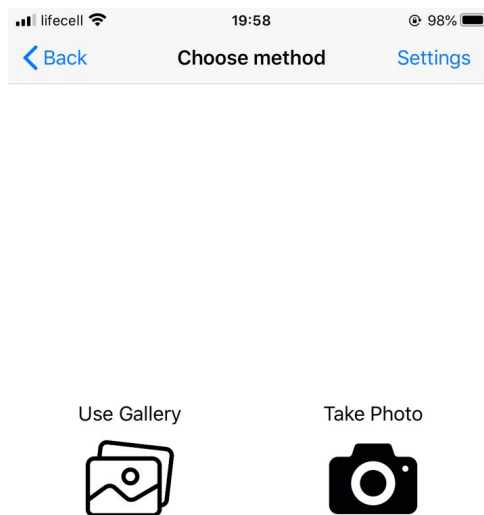
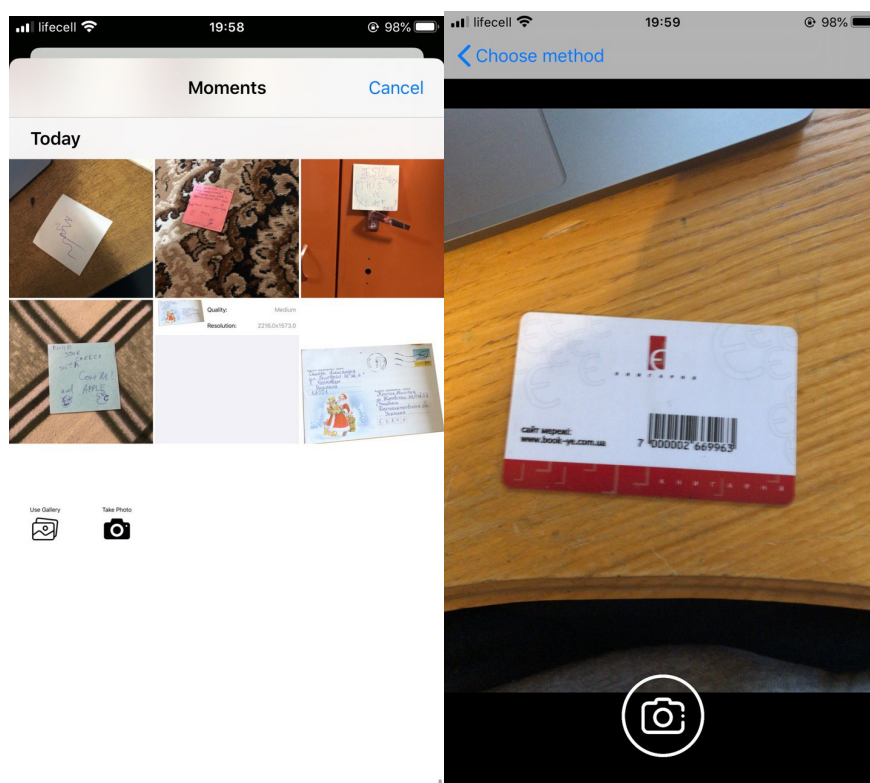


Рисунок 3.13 – Екран вибору методу завантаження зображення до застосунку

На даному екрані у користувача є також можливість виконати налаштування – при натисканні клавіші “Settings”, користувачу відкриється екран із можливістю вибору мінімальної допустимої якості для отриманих зображень.

Після вибору методу завантаження документу користувачу відкриється необхідний екран (рисунки 3.14 – 3.15).



Рисунки 3.14 – 3.15 – Отримання зображення для обробки із галереї пристрою та камери відповідно

Наступним етапом роботи застосунку є екран, на якому проводиться пошук прямокутних областей, які можуть бути документами, пошук тексту в даних областях, які вказують на приналежність прямокутної області до документу та визначається якість отриманого зображення (рисунок 3.16).

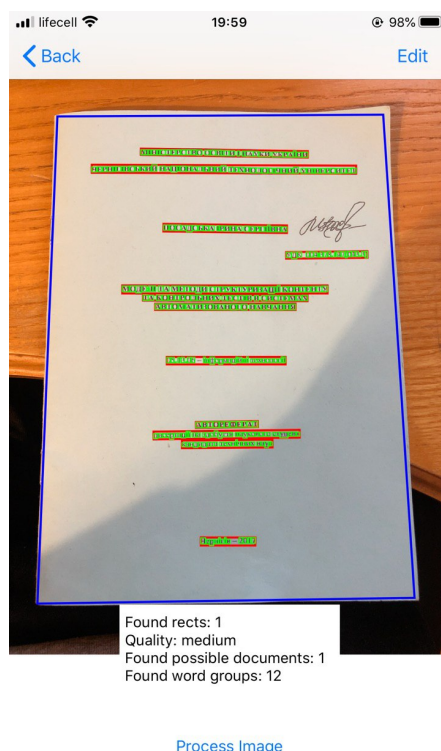


Рисунок 3.16 – Екран із пошуком документу на фотографії

Даний етап є досить важливим, адже якщо зображення буде мати якість меншу за встановлену користувачем, то користувач буде повернутий до попереднього етапу із вибору зображення. Також у випадку відсутності документів із текстом всередині прямокутної області, користувач не матиме можливості визначити тип документу та відсканувати його.

Додатковою опцією для користувача є можливість попереднього редагування зображення перед відправкою на сканування. Дана опція доступна при натисканні кнопки “Edit” у правому верхньому кутку (рисунок 3.17).

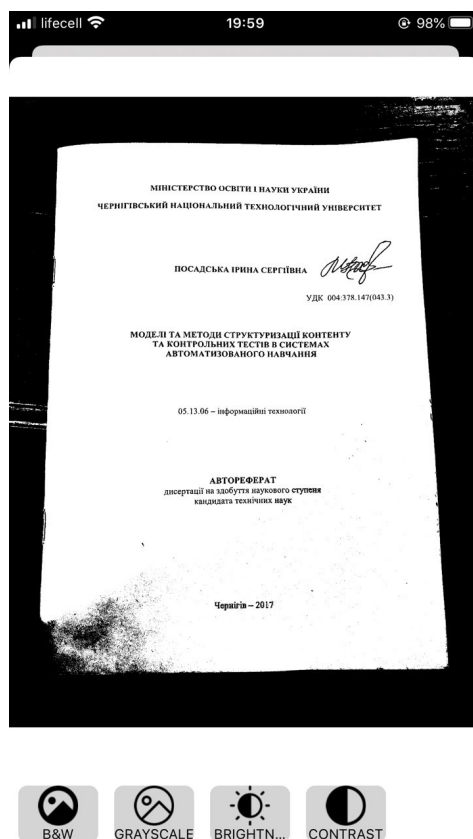


Рисунок 3.17 – Екран редагування зображення перед сканування документу

На даному екрані користувачу надається можливість привести зображення до чорно-білого, відтінків сірого та можливість зміни яскравості та контрастності.

Коли користувач натискає кнопку “Process Image” він потрапляє на наступний екран, на якому він отримує відскановані документи відповідно до знайдених меж та їх визначені типи (рисунок 3.18). Як додаткова опція на даному екрані існує можливість збільшення зображення для детального аналізу.

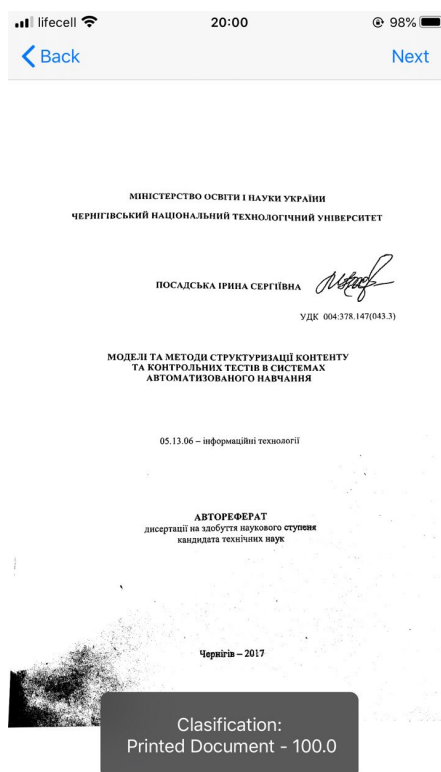


Рисунок 3.18 – Екран з відсканованим документ та його типом

Як бачимо з рисунку 3.17 для визначеного типу документу вказується коефіцієнт впевненості в тому, що документ відноситься до даного типу. У випадку, якщо система невірно визначила тип документу, користувач має можливість змінити тип документу, натиснувши на поле із класифікацією. Також на даному екрані, користувач має можливість натиснути кнопку “Next” та перейти до початкового екрану із усіма відсканованими документами.

На початковому екрані користувачу надається інформація про відсканований документ:

- дата сканування;
- якість оброблюваного зображення;
- розміри відсканованого документу;
- документ автоматично відноситься до певного класу документів.

При натисканні на окремий документ відкривається вікно з документом та його типом для детального перегляду (рисунок 3.19).

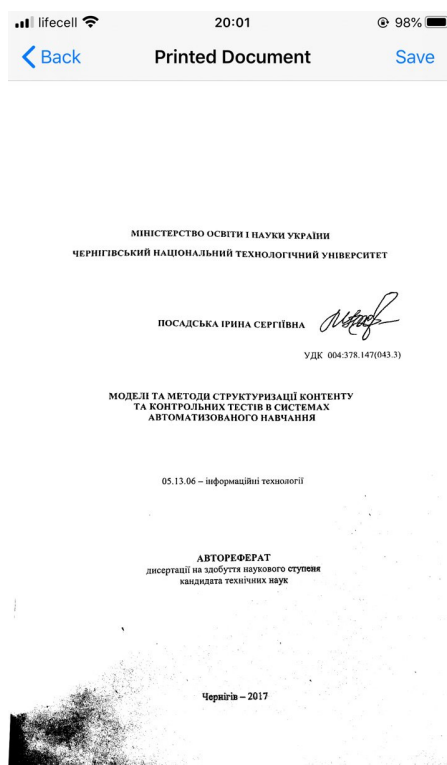


Рисунок 3.19 – Екран перегляду відсканованого документу

На даному екрані наявна кнопка “Save”, яка дозволяє зберегти зображення до галереї мобільного пристрою.

Розроблений інтерфейс є досить простим, адже він є більш демонстративним та призначений для перевірки роботи застосунку. В подальшому планується розробити більш зручний інтерфейс для користувача, який відповідав би усім сучасним трендам.

### 3.5 Аналіз результатів роботи розробленого застосунку

Розроблений застосунок для пошуку та ідентифікації документів на фотографії показав відмінні результати в ході перевірки його роботи.

Аналізуючи функціональні можливості необхідно зазначити, що розроблений застосунок повністю задовольняє поставленим умовам, адже він дозволяє:

- сканування документів з вже зроблених фотографій;
- сканування документів зроблених з камери;
- сканування кількох документів одночасно;
- корекція яскравості та контрастності;
- визначення якості документів;
- визначення класів відсканованих документів;
- перегляд відсканованих документів;
- збереження відсканованих документів;
- сортування документів за різними параметрами: датою, класом документу, якістю.

Слід зазначити, що відповідно до технічних вимог застосунок було розроблено для мобільних пристроїв на базі операційної системи iOS не нижче версії 12.2 та коректно працює на мобільних пристроях із різними розмірами екрану. Також однією із технічних вимог була робота застосунку у вертикальному положенні без зміни орієнтації інтерфейсу при повороті екрану, що повною мірою було реалізовано.

Щоб перевірити застосунок на коректність роботи, необхідно перевірити наступні показники:

- якість знаходження меж документів (як часто застосунок обрізав частину документів);
- точність розпізнавання якості фотографії;
- якість визначення типів документів;
- час пошуку документів на фотографії;
- обсяг використовуваної оперативної пам'яті.

Якість знаходження меж документів визначається у тому, щоб система не обрізала частину документу або навпаки не виділяє значну частину об'єктів простору, які не відносяться до документів. Очікується, що із 100 вимірів у 85 випадках буде прийнятний результат.

В результаті перевірки показника якості знаходження меж документів було сформовано таблицю 3.6.

Таблиця 3.6 – Результати пошуку меж документів із використанням застосунку

| Критерій                             | Кількість коректних випадків | Кількість невірних випадків | Загальна кількість |
|--------------------------------------|------------------------------|-----------------------------|--------------------|
| Точне виявлення меж одного документу | 43                           | 7                           | 50                 |
| Виявлення меж декількох документів   | 42                           | 8                           | 50                 |

Як бачимо із результатів 85% сканувань були вірними. Але слід врахувати наступні зауваження:

1) Під час виявлення меж декількох документів зазвичай у документів, які мають менший розмір межі виявлялися не точно. Дана помилка може бути пов'язана із особливостями роботи бібліотеки Vision.

2) Часто не вірно визначалися межі розгорнутих книг. Це цілком можливо, адже книга у розгорнутому вигляді зазвичай має нечіткі межі і для цього необхідні покращення пошуку меж документів у вигляді мультиполігонального пошуку меж об'єктів, адже пошук прямокутних меж має свої обмеження.

3) Дисконтні картки та візитки дуже часто мають не прямі кути, тому розпізнавання їх меж також часто давало невірні результати, які можна вирішити мультиполігональним пошуком меж об'єктів.

В результаті, якщо взяти до уваги всі коментарі вище, то можна зробити висновок, що пошук меж документів працює добре і помилка у 15% є прийнятною, але щоб підвищити це показник необхідно робити деякі покращення алгоритмів та збільшити кількість навчальних прикладів.

Знаходження якості фотографії є досить важливим показником, який дозволяє покращити якість сканування документів за рахунок сортування тих фотографій, які є досить розмиті. Результат буде вважатися прийнятним, якщо відсоток правильної ідентифікації якості буде більшим за 85%.

В результаті перевірки точності знаходження якості фотографії було сформовано таблицю 3.7.

Таблиця 3.7 – Результати визначення якості фотографії

| Якість | Вірно визначення | Не вірно визначено | Кількість вимірів |
|--------|------------------|--------------------|-------------------|
| Low    | 29               | 4                  | 33                |
| Medium | 27               | 6                  | 33                |
| High   | 30               | 4                  | 34                |

До зображень із якістю “Low” відносяться зображення, які сильно розмиті або зображення є досить зашумленим [53]. До зображень із якістю “Medium” відносяться зображення, які не розмиті, але мають незначні шуми. До зображень із якістю “High” відносяться зображення, які не розмиті та зовсім не мають шумів.

На рисунку 3.20 зображено приклад із 4 зображеннями, де перше зображення має найкращу якість і чіткість, а останнє є досить розмитим та має деякі шуми.

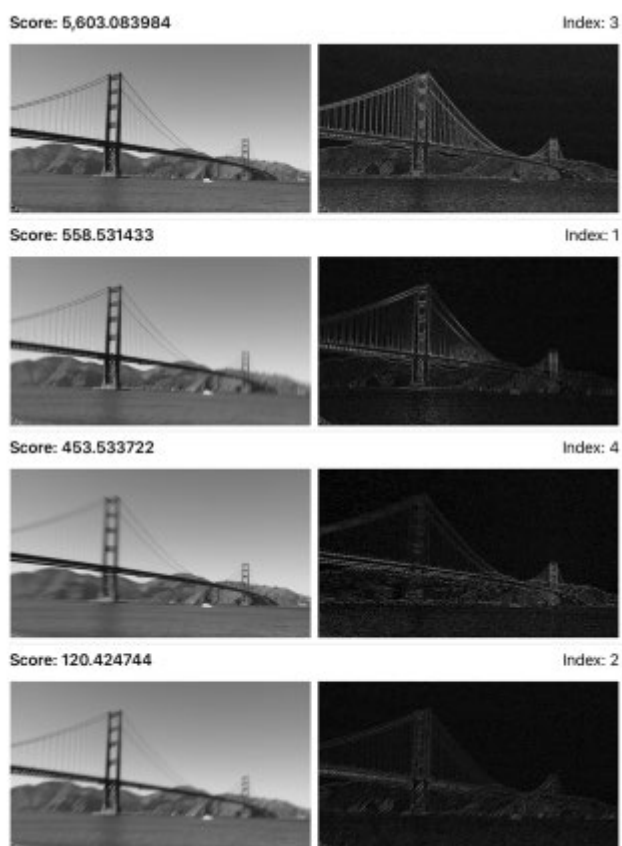


Рисунок 3.20 – Приклад зображень різної якості [53]

На рисунках 3.21 – 3.23 показано приклади роботи застосунку для зображень різної якості.

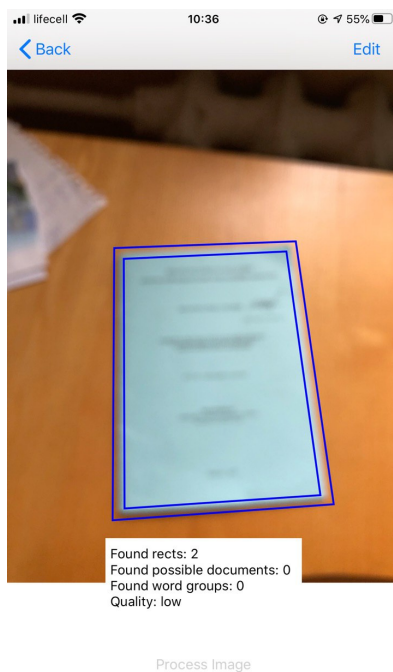


Рисунок 3.21 – Приклад визначення поганої якості для документу

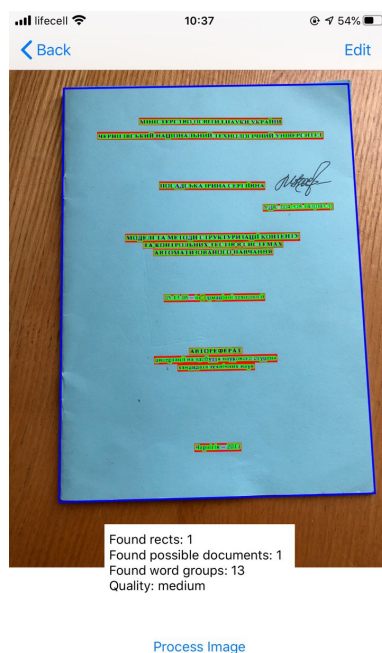


Рисунок 3.22 – Приклад визначення середньої якості для документа

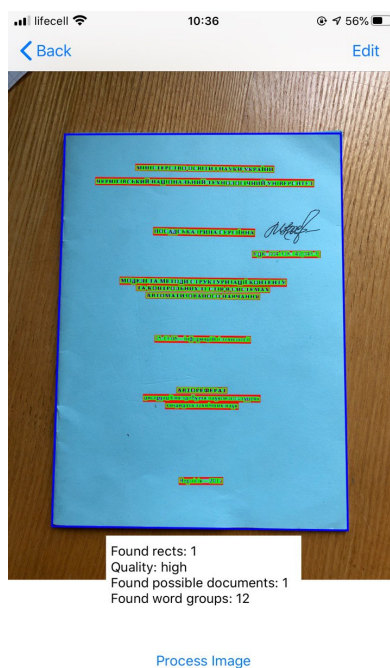


Рисунок 3.23 – Приклад визначення високої якості для документу

З рисунку 3.21 видно, що зображення поганої якості переважно дуже сильно розмиті та при поганому освітленні. На рисунку 3.22 зображення середньої якості має легку зашумленість та не досить добре освітлення навколо. На рисунку 3.23 зображення високої якості має добре освітлення та відсутні шуми.

Як бачимо показник правильної ідентифікації якості дорівнює 86%, що повністю задовольняє поставленим умовам.

Якість знаходження типів документів є досить важливим показником, адже даний функціонал дозволяє визначати тип документа, який в подальшому буде використовуватися для сортування документів. Очікується, що із 100 вимірів у 85 випадках буде вірний результат визначення типу документів.

В результаті перевірки якості визначення типів документів було визначено, що у 83% документів було вірно визначено тип. Переважно з високою ймовірністю було правильно визначено тип документу у книжок, чеків із магазинів, паспортів, рукописних текстів. Але найчастіше помилково визначався невірний тип документів для папірців з нотатка та візиток, адже папірці з нотатками досить легко віднести до рукописних документів, а візитки не мають якогось стандартного патерну, тому згортковій нейронній мережі необхідно більше прикладів, щоб навчитись відрізняти їх від інших класів документів.

Одним із кількісних показників, які необхідно проаналізувати є час пошуку документів на фотографії. Щоб краще розуміти ефективність роботи застосунку, необхідно проаналізувати даний показник з різних сторін, а саме визначити:

- середній час пошуку прямокутних областей;
- середній час пошуку простого документу;
- середній час пошуку двох документів;
- середній час пошуку трьох документів;
- середній час пошуку одного складного документу з великою кількістю тексту.

Відповідно до аналізованих параметрів було створено узагальнену таблицю 3.8.

Таблиця 3.8 – Аналіз часу пошуку різного класу та кількості документів на фотографії

| Показник                                           | Середній час виконання, сек | Кількість вимірів, шт |
|----------------------------------------------------|-----------------------------|-----------------------|
| Пошук прямокутних областей (від 1 до 3 документів) | 0.17                        | 80                    |
| Пошук простого документу                           | 0.33                        | 20                    |
| Пошук двох документів                              | 0.28                        | 20                    |
| Пошуку трьох документів                            | 0.39                        | 20                    |
| Пошук одного складного документу                   | 0.35                        | 20                    |

Відповідно з таблиці 3.8 ми бачимо, що час пошуку документу на фотографії задовольняє необхідний технічний показник у 0.4 секунди. В основному, для одного документу, час пошуку трохи більший, ніж для декількох документів, що може бути пов'язано з тим, що документ займає значну частину зображення.

На рисунках 3.24 – 3.27 наведено приклади роботи застосунку для пошуку від одного простого документу до декількох документів або одного документа із значною кількістю тексту.

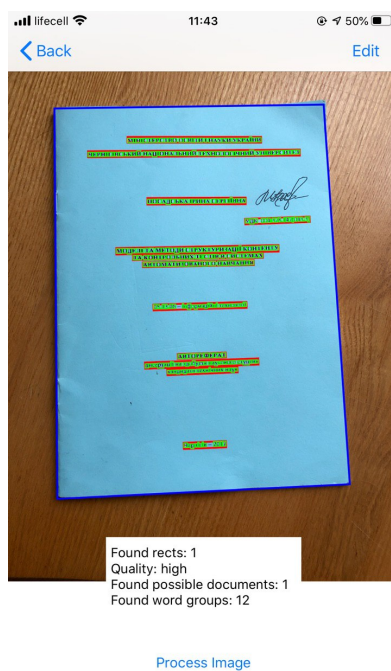


Рисунок 3.24 – Приклад пошуку одного простого документа

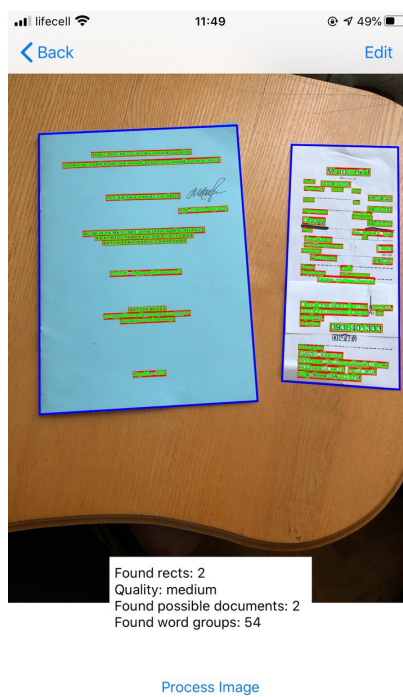


Рисунок 3.25 – Приклад пошуку двох документів на фотографії

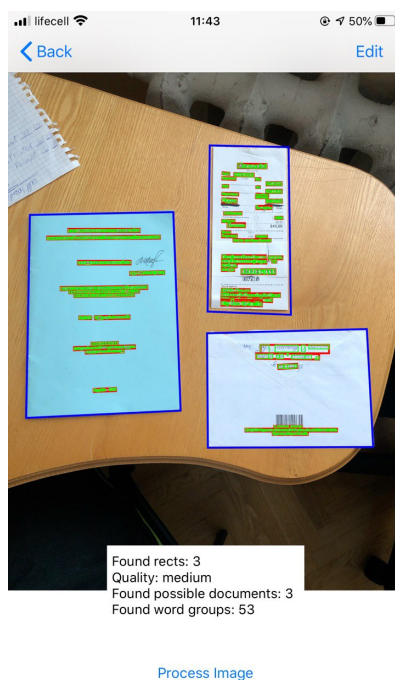


Рисунок 3.26 – Приклад пошуку трьох документів на фотографії

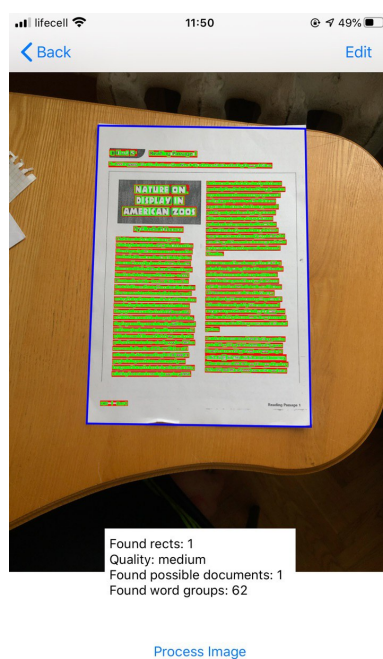
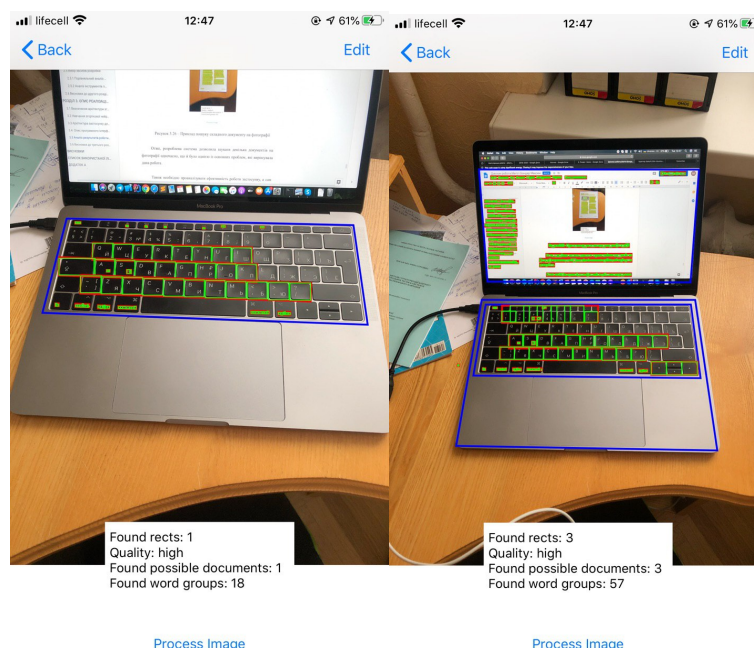


Рисунок 3.27 – Приклад пошуку складного документу на фотографії

Отже, розроблена система дозволила шукати декілька документів на фотографії одночасно, що й було однією із основних проблем, які вирішувала дана робота.

Проте не вдалося вирішити проблему з віднесенням клавіатури до документів (рисунок 3.28).



Рисунки 3.28 – Приклад віднесення клавіатури до документу

Необхідно проаналізувати ефективність роботи застосунку, а саме обсяг оперативної пам'яті, яку він використовує. Щоб краще розуміти ефективність роботи застосунку, необхідно проаналізувати даний показник з різних сторін, а саме визначити:

- кількість використовуваної оперативної пам'яті при запуску;
- кількість використовуваної оперативної пам'яті при роботі з галереєю;
- кількість використовуваної оперативної пам'яті при роботі з камерою;
- кількість використовуваної оперативної пам'яті при пошуку документів та визначенні якості;
- кількість використовуваної оперативної пам'яті при обробці документу;
- кількість використовуваної оперативної пам'яті при скануванні документів;
- кількість використовуваної оперативної пам'яті при повторному поверненні на початковий екран.

Усі зібрані дані під час аналізу представлено в таблиці 3.9.

Таблиця 3.9 – Аналіз кількості використовуваної оперативної пам'яті на різних етапах роботи застосунку

| Етап роботи застосунку                 | Кількість оперативної пам'яті, Мб |
|----------------------------------------|-----------------------------------|
| Запуск                                 | 17.5                              |
| Запуск галереї                         | 21.8                              |
| Запуск камери                          | 27.7                              |
| Пошуку документів та визначення якості | 72.3                              |
| Обробка документу                      | 115                               |
| Сканування документу                   | 160                               |
| Повернення на початковий екран         | 120                               |

З результатів дослідження в таблиці 3.9 випливає, що на етапі сканування документу кількість використовуваної оперативної пам'яті перевищує необхідний максимум у 150 Мб. Але на етапі повернення на початковий екран ми бачимо, що кількість використовуваної оперативної пам'яті дорівнює 120 Мб, що занадто велика кількість для такого простого екрану. Це вказує на те, що застосунок має проблеми з контролем пам'яті та потребує аналізу для усунення витоків пам'яті.

З отриманих результатів аналізу роботи застосунку можна зробити висновок, що розроблена система та технологія цілком задовольняє поставленим вимогам, хоча й має незначні недоліки, які треба усунути в подальших версіях.

### 3.6 Висновки до третього розділу

У даному розділі було розглянуто архітектуру згорткової нейронної мережі, процес її навчання та детально розглянуто та проведено тестування застосунку, побудованого на базі розробленої технології пошуку та ідентифікації документів на фотографії.

Було визначено, що сирі зображення найкраще підходять для навчання за допомогою інструменту Create ML.

Під час тестування фінальної версії застосунку було визначено, що бібліотека Vision має незначні обмеження під час визначення меж документів та точність визначення якості документів повністю відповідає очікуванням. Також було виявлено декілька класів документів, які потребують додаткових навчальних прикладів, для кращого визначення типів документів.

Слід виділити, що розроблена технологія ідентифікації документів дозволила подолати виявлені недоліки у застосунків на ринку – сканування декількох документів одночасно та зниження кількості випадків сканування документів поганої якості. А за рахунок визначення типів відсканованих документів вдалося розширити можливості фільтрування. Проте не вдалося вирішити проблему із віднесенням клавіатури до документу, тому необхідно ще покращувати розроблену технологію.

## ВИСНОВКИ

В результаті виконання дипломного проекту було розроблено покращену технологію пошуку та ідентифікації документів на фотографії. Розроблено технологія дозволяє пришвидшити процес сканування документів за рахунок одночасного сканування декількох документів, покращує якість отриманих документів за рахунок аналізу чіткості граней об'єктів на зображенні та блокуванню можливості сканування документів низької якості. Також розроблена технологія дозволяє аналізувати прямокутні області на наявність текстових областей, щоб з упевненістю відносити їх до документів.

Для перевірки ефективності розробленої технології на практиці було розроблено застосунок для мобільних пристроїв на базі операційної системи iOS. Даний застосунок дозволяє сканувати документи із зображень з галереї та камери смартфона. Для ефективної роботи із зображеннями використовується формат JPEG та PNG, які компанія Apple рекомендує для роботи в своїх застосунках.

В ході розробки програмного продукту було побудовано та натреновано згорткову нейронну мережу, яка дозволяє класифікувати отримані документи до певних класів для розширення можливостей сортування та групування відсканованих документів за класами. В процесі навчання було досягнуто похибки у 9.17 %, що дозволяє з високою точністю визначати класи документів.

В ході перевірки результатів роботи розробленого програмного продукту було визначено, що він повністю відповідає функціональним вимогам. Щодо технічних вимог, то програмний продукт повністю відповідає заявленим вимогам. Проте було визначено декілька недоліків:

- під час роботи було виявлено втрати оперативної пам'яті, які необхідно буде виправити;

- застосунок дозволяє сканувати клавіатуру ноутбуку, яка не відноситься до документів.

Виявлені недоліки необхідно виправити у майбутніх версіях продукту.

З іншого боку, за рахунок розробленої технології вдалося подолати недолік із скануванням декількох документів одночасно, що значно пришвидшило процес сканування документів.

За рахунок визначення якості зображення вдалося зменшити кількість хибних сканувань. А за рахунок визначення класів документів вдалося розширити базові можливості фільтрування документів, що поліпшує процес роботи з документами.

## СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

- 1) Бондар М.О., Красовська Г.В. Аналіз підходів до створення інформаційної системи пошуку та ідентифікації документів на фотографії для мобільних пристроїв на базі операційної системи iOS. *Обчислювальний інтелект (результати, проблеми, перспективи)*: матеріали міжнар. наук.-практ. конф. (Міжнародний науковий симпозіум «Інтелектуальні рішення»), м. Ужгород, 15-20 квітня 2019 р. / М-во освіти і науки України, ДВНЗ «Ужгородський національний університет», та [ін.] – с. 173-175.
- 2) Бондар М.О., Красовська Г.В. Технологія пошуку та ідентифікації документів на фотографії для мобільних пристроїв на базі операційної системи iOS. *Інформаційні технології та взаємодії*: матеріали VI Міжнар. наук.-практ. конф., м.Київ, 20 грудня 2019 р. / М-во освіти і науки України, ДВНЗ «Київський національний університет імені Тараса Шевченка», та [ін.] – с. 213-214.
- 3) Nixon D. User Interfaces. In: *Beginning Unreal Game Development* / D. Nixon. – Berkeley: Apress, 2020. – 389 с. – (ISBN: 978-1-4842-5639-8).
- 4) Adobe scan – сканирование и создание PDF [Електронний ресурс] // App Store – Режим доступу до ресурсу: <https://apple.co/2vjYRfS>.
- 5) HP Smart – Производительность [Електронний ресурс] // App Store – Режим доступу до ресурсу: <https://apple.co/2OEr1Jq>.
- 6) Установка и использование приложения HP Smart (Android, Apple iOS, iPadOS) [Електронний ресурс] // HP – Режим доступу до ресурсу: <https://bit.ly/37gjhEc>.
- 7) FineScanner Сканер документов [Електронний ресурс] // App Store – Режим доступу до ресурсу: <https://apple.co/2Sb6E8W>.
- 8) iScanner – Сканер Документов. [Електронний ресурс] // App Store – Режим доступу до ресурсу: <https://apple.co/2H56FEW>.

9) Notes – Take note of almost anything [Электронный ресурс] // App Store – Режим доступа до ресурсу: <https://apple.co/31AOyjG>.

10) Use Notes on your iPhone, iPad, and iPod touch [Электронный ресурс] // Apple – Режим доступа до ресурсу: <https://support.apple.com/en-us/HT205773>.

11) ABBYY FineScanner AI [Электронный ресурс] // ABBYY – Режим доступа до ресурсу: <https://www.finescanner.com/ru/>.

12) FineScanner iPhone App Review [Электронный ресурс] // Youtube. – 2015. – Режим доступа до ресурсу: <https://bit.ly/2VECoFi>.

13) FineScanner iPad App Demo [Электронный ресурс] // Youtube. – 2016. – Режим доступа до ресурсу: <https://cutt.ly/tya6HRb>.

14) Алгоритмы выделения контуров изображений [Электронный ресурс] // Хабр. – 2011. – Режим доступа до ресурсу: <https://habr.com/ru/post/114452/>.

15) Алгоритм Кенні [Электронный ресурс] // Вікіпедія - вільна енциклопедія. – 2020. – Режим доступа до ресурсу: <https://cutt.ly/oya6H2u>

16) Великий Я. О. Анализ принципа распознавания объектов на изображении методом Виолы–Джонса / Я. О. Великий. // Открытые информационные и компьютерные интегрированные технологии. – 2015. – №68. – С. 162–166.

17) Метод Виолы-Джонса (Viola-Jones) как основа для распознавания лиц. [Электронный ресурс] // Хабр – Режим доступа: <https://habrahabr.ru/post/133826/>.

18) Viola P., Jones M.J., Snow D. Detecting pedestrians using patterns of motion and appearance // Proceedings of the 9th International Conference on Computer Vision (ICCV'03). 2003. V. 1. P. 734–741.

19) UIImage - UIKit [Электронный ресурс] – Режим доступа до ресурсу: <https://developer.apple.com/documentation/uikit/uiimage>.

20) Open Source Computer Vision Library [Электронный ресурс] // OpenCV. – 2020. – Режим доступа до ресурсу: <https://opencv.org>.

21) Алгоритм сегментации по принципу скользящего окна и вычисление контраста [Электронный ресурс] // Compvision.ru. – 2017. – Режим доступа до ресурсу: <https://bit.ly/355zRaa>.

22) Framework Vision [Электронный ресурс] – Режим доступа до ресурсу: <https://developer.apple.com/documentation/vision?language=objc>

23) Three algorithms for converting color to grayscale [Электронный ресурс] // JOHN D. COOK CONSULTING. – 2009. – Режим доступа до ресурсу: <https://www.johndcook.com/blog/2009/08/24/algorithms-convert-color-grayscale/>.

24) Mordvintsev A. Image Thresholding [Электронный ресурс] / A. Mordvintsev, R. C. Gonzalez // OpenCV Official Documentation. – 2013. – Режим доступа до ресурсу: <https://cutt.ly/0ya6Jzk>

25) Loch F. G. IMAGE PROCESSING ALGORITHMS PART 4: BRIGHTNESS ADJUSTMENT [Электронный ресурс] / Francis G. Loch // Dreamland Fantasy Studios. – 2010. – Режим доступа до ресурсу: <https://www.dfstudios.co.uk/articles/programming/image-programming-algorithms/image-processing-algorithms-part-4-brightness-adjustment/>.

26) Loch F. G. IMAGE PROCESSING ALGORITHMS PART 5: CONTRAST ADJUSTMENT [Электронный ресурс] / Francis G. Loch // Dreamland Fantasy Studios. – 2010. – Режим доступа до ресурсу: <https://www.dfstudios.co.uk/articles/programming/image-programming-algorithms/image-processing-algorithms-part-5-contrast-adjustment/>.

27) Laplace Operator [Электронный ресурс] // Open Source Computer Vision Documentation. – 2020. – Режим доступа до ресурсу: [https://docs.opencv.org/3.4/d5/db5/tutorial\\_laplace\\_operator.html](https://docs.opencv.org/3.4/d5/db5/tutorial_laplace_operator.html).

28) Yiu T. Understanding Neural Networks [Электронный ресурс] / Tony Yiu // Towards Data Science. – 2019. – Режим доступа до ресурсу: <https://bit.ly/3cMHBAo>.

29) Применение нейросетей в распознавании изображений [Электронный ресурс] // Хабр. – 2009. – Режим доступа до ресурсу: <https://habr.com/ru/post/74326/>.

30) Parallel distributed processing model with local space-invariant interconnections and its optical architecture / Z.Wei, I. Kazuyoshi, T. Jun, I. Yoshiki. // APPLIED OPTICS. – 1990. – С. 4790–4797.

31) Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard and L. D. Jackel: Backpropagation Applied to Handwritten Zip Code Recognition, Neural Computation, 1(4):541-551, Winter 1989.

32) Convolutional Neural Networks (LeNet) - DeepLearning 0.1 documentation [Электронный ресурс] // DeepLearning 0.1. LISA Lab. – Режим доступа до ресурсу: <http://deeplearning.net/tutorial/lenet.html>.

33) Свёрточная нейронная сеть [Электронный ресурс] // Википедии — свободная энциклопедия. – 2018. – Режим доступа до ресурсу: [https://ru.wikipedia.org/wiki/Свёрточная\\_нейронная\\_сеть#cite\\_note-1](https://ru.wikipedia.org/wiki/Свёрточная_нейронная_сеть#cite_note-1).

34) Сравнение библиотек глубокого обучения на примере задачи классификации рукописных цифр [Электронный ресурс] / [Д. Кручинін, Є. Долотов, В. Кустикова та ін.] // Habr. – 2015. – Режим доступа до ресурсу: <https://habr.com/ru/company/intel/blog/254747/>.

35) Subject independent facial expression recognition with robust face detection using a convolutional neural network / [M. Masakazu, M. Katsuhiko, M. Yusuke та ін.]. // Neural Networks : journal. – 2003. – С. 555—559.

36) Eleonora Cagli1, Cecile Dumas, Emmanuel Prouff. Convolutional Neural Networks with Data Augmentation against Jitter-Based Countermeasure International on Conference on Multimedia, pages 6-7. ACM, 2014.

37) Framework Core ML [Электронный ресурс] // Developer – Режим доступа до ресурсу: <https://developer.apple.com/documentation/coreml>.

38) Thakkar M. Introduction to Core ML Framework. In: Beginning Machine Learning in iOS. / Mohit Thakkar. – Berkeley, CA: Apress, 2019. – 157 с. – (Apress). – (DOI 10.1007/978-1-4842-4297-1).

39) Create ML – Creating an Image Classifier Model [Электронный ресурс] // Developer – Режим доступа до ресурсу:

[https://developer.apple.com/documentation/coreml/creating\\_an\\_image\\_classifier\\_model](https://developer.apple.com/documentation/coreml/creating_an_image_classifier_model).

40) Thakkar M. Custom Core ML Models Using Create ML: CoreML Framework [Электронный ресурс] / Mohit Thakkar // ResearchGate GmbH. – 2019. – Режим доступа до ресурсу: <https://bit.ly/35aRghM>.

41) Framework Vision [Электронный ресурс] // Developer – Режим доступа до ресурсу: <https://developer.apple.com/documentation/vision>.

42) Adaptive Convolutional ReLUs [Электронный ресурс] / J.Ning, A. Ndagijimana, J. Ibare, R. Nkusi // Research Gate GmbH. – 2019. – Режим доступа до ресурсу: <https://bit.ly/2yPrGmc>.

43) Understand the Softmax Function in Minutes [Электронный ресурс] // Medium - Data Science Bootcamp. – 2018. – Режим доступа до ресурсу: <https://medium.com/data-science-bootcamp/understand-the-softmax-function-in-minutes-f3a59641e86d>.

44) Budhiraja A. Dropout in (Deep) Machine learning [Электронный ресурс] / Amar Budhiraja // Medium - Data Science Bootcamp. – 2016. – Режим доступа до ресурсу: <https://cutt.ly/jya6KSR>.

45) Turi Create API Documentation [Электронный ресурс] // Turi Create API – Режим доступа до ресурсу: <https://apple.github.io/turicreate/docs/api/> (дата звернення: 20.04.2020).

46) Custom Core ML Models Using Turi Create: CoreML Framework. // Beginning Machine Learning in iOS / – Berkeley, CA: Apress, 2019. – (Apress). – С. 51–94.

47) Konstantinos Mersinas, Theresa Sobb, Jonathan Bakdash. Training Data and Rationality. Conference Paper, European Conference on the Impact of AI and Robotics (ECIAIR), April 2020. Режим доступа до ресурсу: <https://cutt.ly/Mya6LmH>

48) Core Data Framework [Электронный ресурс] // Apple Developer. – 2020. – Режим доступа до ресурсу: <https://developer.apple.com/documentation/coredata>.

49) Swift. A powerful open language that lets everyone build amazing apps. [Електронний ресурс] // Apple Inc. – 2020. – Режим доступу до ресурсу: <https://www.apple.com/swift/>.

50) Goodwill J. The Swift Programming Language / J. Goodwill, W. Matlock // Beginning Swift Games Development for iOS / J. Goodwill, W. Matlock., 2015. – (Apress). – (DOI 10.1007/978-1-4842-0400-9).

51) Framework Core Graphics [Електронний ресурс] // Developer – Режим доступу до ресурсу: <https://developer.apple.com/documentation/coregraphics>

52) Framework UIKit [Електронний ресурс] // Developer – Режим доступу до ресурсу: <https://developer.apple.com/documentation/uikit> (дата звернення: 20.04.2020).

53) Finding the Sharpest Image in a Sequence of Captured Images [Електронний ресурс] // Apple Developer. – 2020. – Режим доступу до ресурсу: [https://developer.apple.com/documentation/accelerate/finding\\_the\\_sharpest\\_image\\_in\\_a\\_sequence\\_of\\_captured\\_images](https://developer.apple.com/documentation/accelerate/finding_the_sharpest_image_in_a_sequence_of_captured_images).

54) Буц В. В. Способи організації засобів нейромережевого розпізнавання об'єкта на зображенні: магістр, Комп'ютерна інженерія, Системне програмування, КПІ ім. Сікорського : 123 / Буц Вікторія Віталіївна – Київ, 2018. – 129 с.

## ДОДАТОК А

В даному додатку наведено таблиці із порівняльною характеристикою бібліотек OpenGL та Vision. Параметр якість вказує на якість пошуку прямокутних областей на відео/фотографії, а завантаженість ЦП більше 100% вказує на завантаженість більше ніж одного ядра процесора.

Таблиця А.1 – Порівняння OpenCV та Vision для відео якості 1080p 60 кадрів секунду

| Кількість<br>об'єктів,<br>шт | Бібліотека          | Початков<br>а<br>кількість<br>ОП, Мб | ОП,<br>Мб   | Завантаженіс<br>ть<br>ядер ЦП,<br>% | Якіст<br>ь |
|------------------------------|---------------------|--------------------------------------|-------------|-------------------------------------|------------|
| 3                            | Vision, Fast        | 35                                   | 55-60       | 60                                  | 3          |
| 3                            | Vision,<br>Accurate | 35                                   | 75-90       | 65-70                               | 4          |
| 3                            | OpenCV              | 50                                   | 120-<br>130 | 120-140                             | 2          |
| 1                            | Vision, Fast        | 35                                   | 45-60       | 55-60                               | 5          |
| 1                            | Vision,<br>Accurate | 35                                   | 60-75       | 60-65                               | 5          |
| 1                            | OpenCV              | 50                                   | 120-<br>125 | 110-120                             | 5          |

Таблиця А.2 – Порівняння OpenCV та Vision для відео якості 4К 30 кадрів секунду

| Кількість<br>об'єктів,<br>шт | Бібліотека          | Початков<br>а<br>кількість<br>ОП, Мб | ОП,<br>Мб   | Завантаженіс<br>ть<br>ядер ЦП,<br>% | Якіст<br>ь |
|------------------------------|---------------------|--------------------------------------|-------------|-------------------------------------|------------|
| 3                            | Vision, Fast        | 42                                   | 70-140      | 120                                 | 5          |
| 3                            | Vision,<br>Accurate | 42                                   | 90-150      | 110-130                             | 4          |
| 3                            | OpenCV              | 60                                   | 150-<br>170 | 140-190                             | 2          |
| 1                            | Vision, Fast        | 42                                   | 60-110      | 55-60                               | 5          |
| 1                            | Vision,<br>Accurate | 42                                   | 80-130      | 60-65                               | 4          |
| 1                            | OpenCV              | 60                                   | 140-<br>150 | 130-150                             | 4          |