

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»**

Навчально-науковий фізико-технічний інститут

Кафедра математичного моделювання та аналізу даних

«На правах рукопису»

УДК 004.932

До захисту допущено:

Завідувач кафедри

_____ Наталія КУССУЛЬ

«__» _____ 20__ р.

Магістерська дисертація

на здобуття ступеня магістра

**за освітньо-професійною програмою «Математичні методи
моделювання, розпізнавання образів та комп'ютерного зору»**

зі спеціальності 113 «Прикладна математика»

на тему: «Вирівнювання зображень розлінованих аркушів»

Виконав (-ла): студентка II курсу, групи ФІ-11мп

Бондар Марія Олександрівна _____

Науковий керівник:

Доцент, к.ф.-м.н. Терещенко Іван Миколайович

Консультант:

м.н.с. Кригін Валерій Михайлович _____

Рецензент:

Посада, науковий ступінь, вчене звання,

Прізвище, ім'я, по батькові _____

Засвідчую, що у цій магістерській
дисертації немає запозичень з праць
інших авторів без відповідних посилань.

Студентка _____

Київ – 2022 року

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Навчально-науковий фізико-технічний інститут
Кафедра математичного моделювання та аналізу даних

Рівень вищої освіти – другий (магістерський)

Спеціальність – 113 «Прикладна математика»

Освітньо-професійна програма: «Математичні методи моделювання, розпізнавання образів та комп'ютерного зору»

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____ Наталія КУССУЛЬ

«__» _____ 20__ р.

ЗАВДАННЯ
на магістерську дисертацію студенту

Бондар Марії Олександрівні

1. Тема дисертації «Вирівнювання зображень розлінованих аркушів», науковий керівник дисертації Терещенко Іван Миколайович, доцент, к.ф.-м.н., затверджені наказом по університету від «__» _____ 20__ р. № _____
2. Термін подання студентом дисертації _____
3. Об'єкт дослідження: зображення розлінованих аркушів паперу
4. Предмет дослідження: вирівнювання зображень аркушів паперу
5. Перелік завдань, які потрібно розробити
 1. Ознайомитися з літературою за темою роботи
 2. Провести аналіз існуючих методів вирівнювання зображень
 3. Дослідити методи і правила проєктивної геометрії

4. Дослідити методи обробки зображень
 5. Скласти математичний алгоритм для вирівнювання зображень аркушів та реалізувати його мовою Python
 6. Провести аналіз результатів та зробити висновки
6. Орієнтовний перелік графічного (ілюстративного) матеріалу: 23 ілюстрації, 6 таблиць.
7. Орієнтовний перелік публікацій
8. Консультанти розділів дисертації

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
3	Кригін В. М., м.н.с		

9. Дата видачі завдання _____

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів магістерської дисертації	Примітка

Студент
Науковий керівник

Марія БОНДАР
Іван ТЕРЕЩЕНКО

РЕФЕРАТ

Робота складається з 4 розділів, містить 23 ілюстрації, 6 таблиць, 21 літературних посилань, обсяг роботи — 65 сторінки.

Сучасне навчання поширює процес передачі інформації за допомогою знімків аркушів, що потребують швидкої обробки для комфортного візуального сприйняття. Частина зображень конспектів та письмових контрольних завдань зроблена камерою пересічного смартфона за слабкого освітлення та під несприятливим для зору кутом, тому потребує застосування методів покращення їх якості. Оскільки тема вирівнювання зображень не є новою, наразі існує досить багато комерційних застосунків, котрі дають змогу користувачам вирівнювати фотографії документів і робити їх цифрові скан-копії. Однак більша їх частина орієнтована на пошук аркушу паперу за спеціальних умов зйомки. У цій роботі пропонується метод вирівнювання зображень розлінованих у клітинку або косу лінію аркушів, що може бути доповненням до вже існуючих інструментів.

Мета даної роботи — розробити метод вирівнювання зображень аркушів, які мають розмітку в клітинку або у косу лінію.

Об'єкт дослідження — зображення розлінованих аркушів паперу.

Предмет дослідження — вирівнювання зображень аркушів паперу.

Методами дослідження є аналіз інформаційних джерел, новітніх публікацій за темою дослідження, дослідження методів обробки зображень та правил проективної геометрії.

Актуальність роботи зумовлюється тим, що можливості редагування зображень документів є популярними та потрібними у повсякденному житті, зокрема у сучасній шкільній та університетській освіті, невід'ємною частиною якого є процес передачі інформації за допомогою знімків аркушів.

Наукова новизна роботи полягає в тому, що запропоновано алгоритм вирівнювання аркушів, що мають розмітку. Алгоритми такої спеціалізації нам не відомі, так як інші у своїй роботі використовують границі аркушів або текст.

Практичне застосування полягає в тому, що даний підхід дозволяє вирівнювати зображення, які не задовольняють умовам зйомки, що вимагають існуючі методи, що є їх якісним доповненням.

Роботу оформлено за вимогами та подано на публікацію до міжнародного науково-теоретичного журналу “Кібернетика і системний аналіз”.

Ключові слова: комп’ютерний зір, вирівнювання зображень документів, сканування документів, проективна геометрія, RANSAC, перспективне перетворення.

ABSTRACT

The work consists of 4 sections, 23 illustrations, 6 tables, and 21 references, and the volume of the work is 65 pages.

Modern education spreads the process of transferring information using images of sheets that require fast processing for comfortable visual perception. Some of the images of notes and written control tasks were taken by the camera of a general smartphone in low light and at an inconvenient angle for vision. Therefore, it is necessary to use methods to improve their quality. Since image alignment is not a new topic, currently, there are quite a lot of commercial applications that allow users to rectify photos of documents and make their digital scans. However, most focus on finding a sheet of paper under special shooting conditions. This work proposes a method for aligning images of sheets lined in a cell or diagonal line, which can be an extension of already existing tools.

This work aims to develop a method for rectifying images of sheets with a grid or diagonal line marking.

The object of research is images of lined sheets of paper.

The subject of research is alignment of images of sheets of paper.

The research methods are an analysis of information sources and the latest publications on the research topic, study of image processing methods and rules of projective geometry.

The relevance of the work is determined by the fact that the ability to edit images of documents is prevalent and necessary in everyday life, particularly in modern school and university education, an integral part of which is the process of transmitting information using images of sheets.

The scientific novelty of the work is that an algorithm for aligning sheets with markings is proposed. We do not know algorithms for such specialization, as others use sheet borders or text in their work.

The practical application is that this approach allows you to rectify images that do not meet the shooting conditions required by existing methods, which could be their qualitative addition.

The work has been designed according to the requirements and submitted for publication to the international scientific and theoretical journal "Cybernetics and System Analysis".

Keywords: computer vision, document image alignment, document scanning, projective geometry, RANSAC, perspective transformation.

ЗМІСТ

1. Огляд існуючих методів.....	13
1.1 Наукові роботи.....	14
1.1.1 Auto-rectification of user photos	14
1.1.2 Metric Rectification for Perspective Images of Planes.....	16
1.1.3 Rectification of Camera Captured Document Imagesfor Camera-Based OCR Technology	17
1.2 Алгоритми комерційних застосунків.....	19
Висновок до розділу 1	20
2. Компоненти методу вирівнювання зображень розлінованих аркушів.....	22
2.1 Алгоритм Кенні пошуку країв елементів	22
2.2 Алгоритм Хафа для знаходження ліній.....	24
2.3 Метод RANSAC (Random sample consensus)	26
2.4 Елементи проективної геометрії	28
2.4.1 Задання прямих і точок в проективному просторі	28
2.4.3 Точка і пряма на нескінченності	30
2.4.4 Перспективні перетворення.....	31
Висновок до розділу 2	35
3. Опис алгоритму ректифікації розлінованих аркушів.....	36
3.1 Знаходження ліній на зображенні	37
3.2 Групування (кластеризація) ліній	38
3.2.1 Оцінка відстані між точками у проективній площині.....	39

3.2.2 Виділення груп співнапрямлених ліній та їх точок сходу.....	40
3.3 Побудова матриці перспективного перетворення	42
3.4 Знаходження обмежувальної рамки	44
3.5 Виділення аркуша з фону.....	46
Висновок до розділу 3	50
4. Оцінка результатів та обмеження алгоритму	52
4.1 Оцінка результатів експериментів	52
4.2 Обмеження розробленого алгоритму	57
Висновок до розділу 4	60
Висновки.....	62
Перелік посилань	64

ВСТУП

Актуальною темою є поліпшення якості дистанційної освіти. Багато школярів і студентів вищих учбових закладів були змушені перейти на дистанційну форму навчання, а деякі навіть на заочну. Не зважаючи на те, що наразі існує досить велика кількість програмних застосунків, які полегшують роботу викладача і допомагають оптимізувати навчальний процес в online-форматі, основним способом передачі інформації у письмовому вигляді від учня до вчителя залишаються записи у зошитах — конспекти, домашні завдання та інші контрольні заходи. Це зазвичай спонукає до процесу передачі інформації за допомогою знімків аркушів, що потребують швидкої обробки для комфортного візуального сприйняття.

Значна частина зображень, що надсилається для перевірки, робиться камерою пересічного смартфона за слабкого освітлення та під несприятливим для зору кутом, що ускладнює сприйняття їх вмісту. Приклади таких зображень наведені на рис. 1. Це призводить до потреби користування спеціальними програмними засобами, які дозволяють мінімізувати дефекти і покращити якість фото.

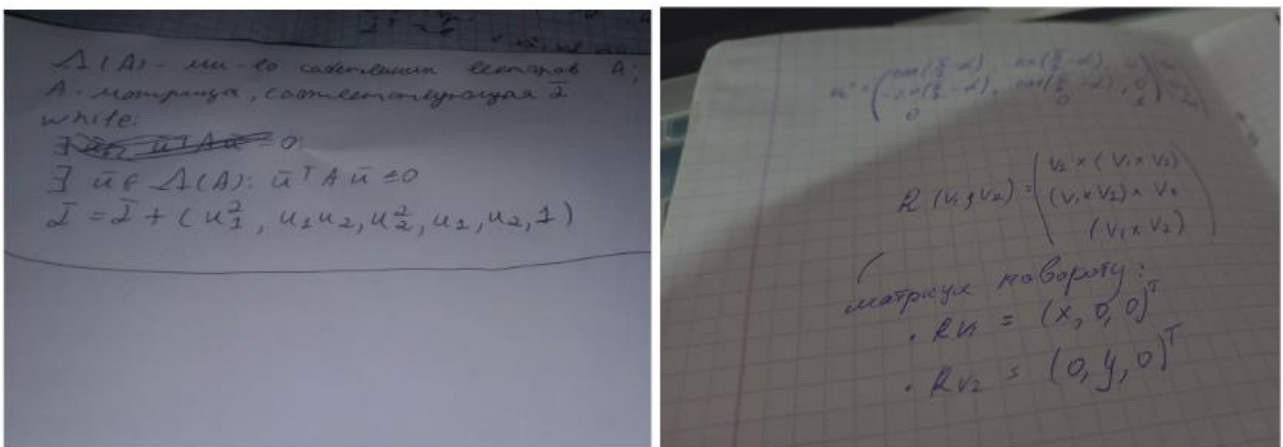


Рис. 1. Приклади неякісного зображення аркуша

Оскільки тема вирівнювання зображень не є новою і має широке застосування у різних сферах, зокрема в опрацюванні зображень паперових документів, наразі існує досить багато комерційних застосунків, котрі дають змогу користувачам вирівнювати фотографії документів і робити їх цифрові скан-копії. Прикладами таких програм є Dropbox [1], PDF-scanner [2], Google Drive [3], Adobe Scan [4] та інші. Більша частина з них орієнтована на пошук аркушу паперу за спеціальних умов зйомки, а саме: зображення містить документ цілком, фон зображення контрастує з кольором документа, ніякий інший документ не перекриває основний. У інших випадках вони часто не в змозі коректно відшукати аркуш на фото і вирівняти його.

У цій роботі ми пропонуємо метод вирівнювання зображень розлінованих у клітинку або косу лінію аркушів. Даний підхід дозволяє вирівнювати зображення, які не підпадають під критерії, описані у попередньому абзаці, що є якісним доповненням вже існуючих методів.



Рис. 2. Фото зліва – фон зображення не контрастує з кольором документа, фото по центру – один документ виглядає з-під іншого, фото справа – зображення містить не весь документ

Структура роботи наступна. Розділ 1 містить огляд існуючих робіт на тему вирівнювання зображень. У розділі 2 наведено основні математичні об'єкти та алгоритми, необхідні для опису нашого методу. У розділі 3 описуються кроки нашого підходу, а саме: знаходження ліній на зображенні, кластеризація ліній, вирівнювання (ректифікація) зображення, обрізання зображення по обмежувальній рамці. Розділ 4 освітлює результати експериментів, обмеження алгоритму та висновки.

РОЗДІЛ 1

ОГЛЯД ІСНУЮЧИХ МЕТОДІВ

Обробка зображень документів охоплює різні сфери діяльності, зокрема аналіз графіки, обробка каталогів цифрових бібліотек, криміналістику тощо. Також можливості редагування зображень є популярними та потрібними у повсякденному житті. Наразі цифрові фотоапарати, відеокамери і телефонні камери можуть замінити сканери та зробити обробку зображень документів більш гнучкою. Однак фотокопії документів зазвичай мають різний характер спотворень. Через неконтрольоване положення камери, можливе таке погіршення якості як нерівномірне освітлення та різноманітні деформації об'єктів зображення, одним з яких є перспективне викривлення, за якого відносні розміри та кути окремих частин зображення не відповідають дійсності.

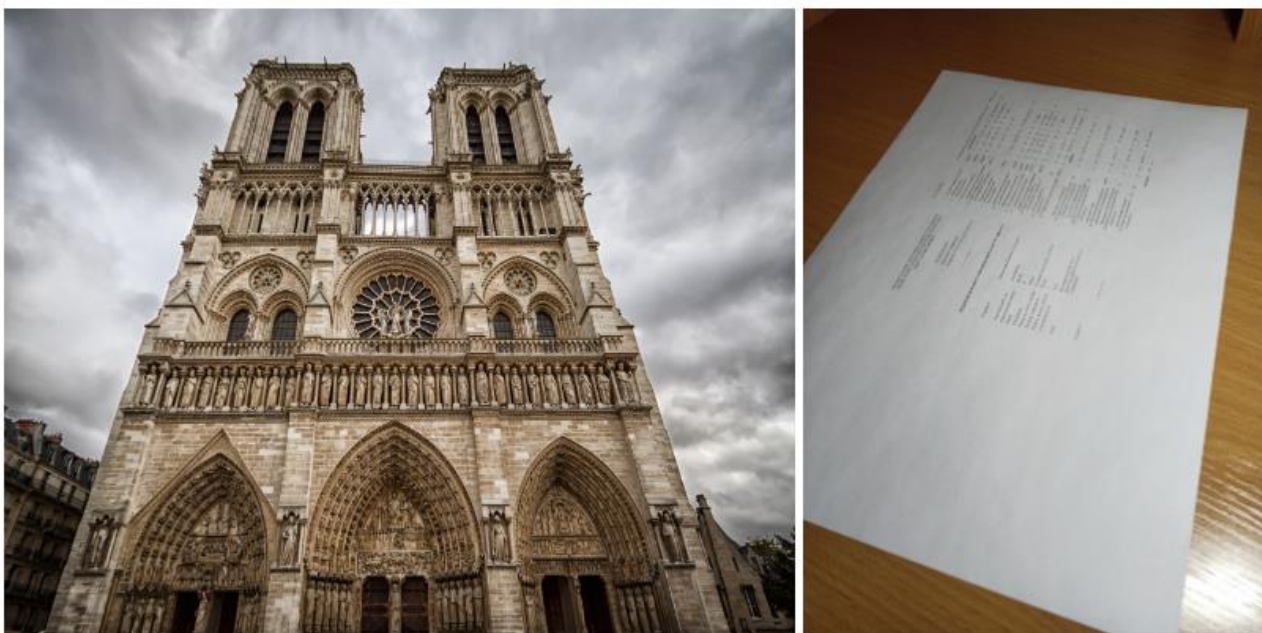


Рис 1.1. Приклади перспективних викривлень зображень [19]

Перспективне вирівнювання зображень, тобто перенесення рисунку у площину, перпендикулярну до оптичної осі камери, є основним предметом багатьох наукових досліджень і комерційних програм [5-6, 8, 11-12]. У цьому розділі подано короткий огляд деяких з них.

1.1 Наукові роботи

Нижче наведено опис наукових праць, які, на наш погляд, є релевантними даній роботі. Вчені пропонують різні методи перспективного вирівнювання зображень, деякі їх етапи використано нами для досягнення мети.

1.1.1 Auto-rectification of user photos

У роботі [6] науковців компанії Google автори запропонували метод автоматичної корекції фото, зроблених звичайними користувачами смартфонів. Такі фото мають невеликі спотворення, спричинені поворотом камери відносно оптичної осі. Цей метод має два основні кроки: знаходження двох точок сходу на зображенні та побудова матриці гомографії для переведення зображення у його фронтально-паралельний вид.

На першому кроці знаходяться так звані еджлети (edgelets) — вектори, структура яких містить координати пікселя в однорідних координатах, що відображає границю на зображенні, його напрям в однорідних координатах та інтенсивність кольору пікселя. Далі запускається алгоритм на базі методу RANSAC [7] для знаходження точок сходу. Особливість цього підходу полягає в тому, що при використанні edgelets немає потреби в пошуку ліній для оцінки точок сходу.

У нашій роботі ми також використовуємо метод RANSAC для пошуку точок сходу, але на вхід йому подаємо саме лінії, знайдені на зображенні алгоритмом Хафа [10], що дає можливість виділяти групи паралельних ліній із загального масиву для подальшого їх використання при обрізанні зображення.

На другому кроці методу, що розглядається в цьому пункті, будується матриця вирівнювання зображення на основі точок сходу з попереднього кроку та кута нахилу до вертикальної осі найближчої до неї точки.

Обмеженням даного алгоритму є те, що він спроможний відновлювати лише такі зображення, в яких точки сходу відповідають ортогональним лініям. На рис. 3 можна побачити результат роботи цього алгоритму.



Рис. 1.4. Результат відновлення зображення методом науковців компанії Google

У запропонованому нами методі також використовується орієнтація точок сходу відносно головних осей координат, а також апріорі відомий кут між групами ліній. Це дає змогу вирівнювати фото не тільки на основі ортогональних ліній, а й ліній, котрі перетинаються під будь-яким наперед заданим кутом.

1.1.2 Metric Rectification for Perspective Images of Planes

У роботі Д. Забовіца і А. Циссермана [5] описано геометричні обмеження та алгоритмічну імплементацію метричного випрямлення площин, яке дозволяє вимірювати кути та співвідношення довжин на світовій площині з перспективного зображення. Покроково описано декомпозицію матриці перспективного перетворення на декілька груп окремих перетворень, а саме: чисте проєктивне перетворення, яке робить паралельними ті лінії зображення, котрі є паралельними на площині, перпендикулярній оптичній осі камери; афінне перетворення, яке відновлює кути та відношення довжин зображення; перетворення подібності – масштабування та поворот зображення. Одним з важливих результатів цієї роботи є те, що в ній показується, що ці метричні властивості об'єктів зображення можна представити як кругові обмеження на параметри афінного перетворення площини, що дозволяє створити загальний алгоритм ректифікації. На рис. 1 та 2 показано результати кроків даного алгоритму. Більш детально про види перетворення зображень описано у розділі 2 цієї роботи.

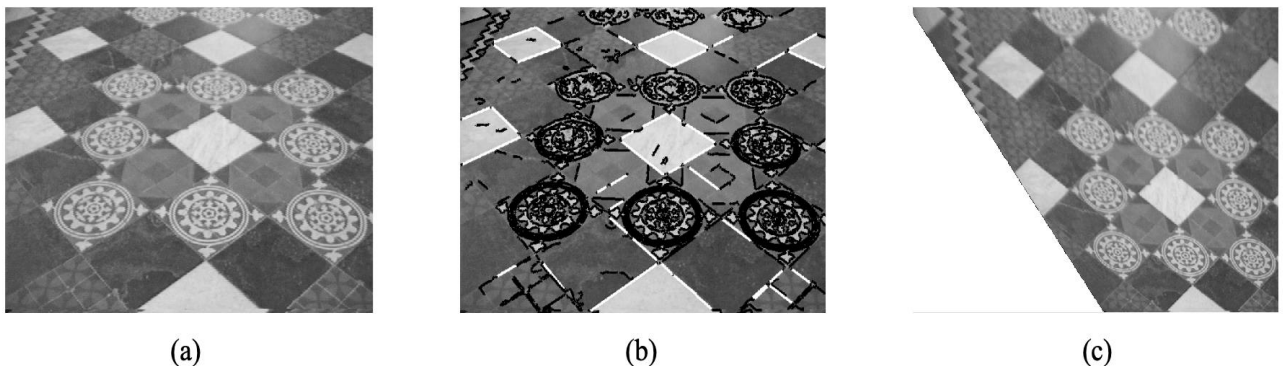


Рис. 1.2. Афінне випрямлення фото візерункової підлоги. (а) Оригінальне зображення. (б) виявлені сегменти ліній (с) Випрямлене афінне зображення

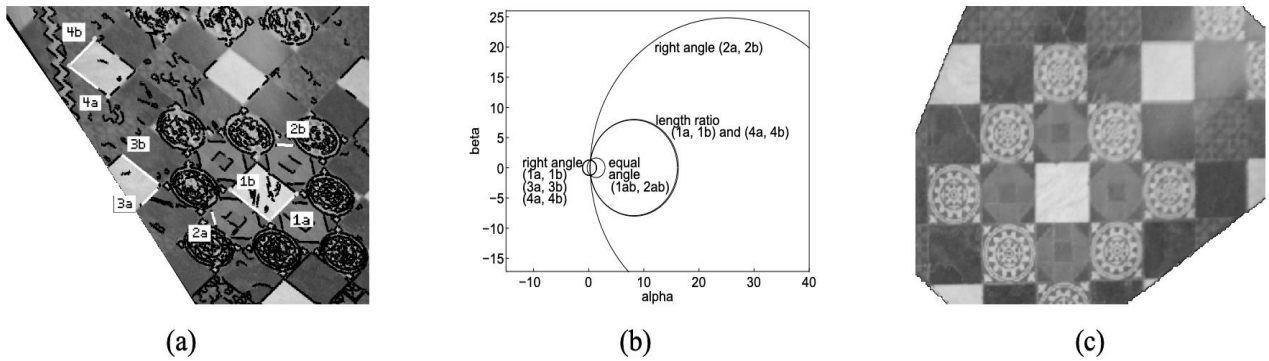


Рис. 1.3. Метричне випрямлення рисунка 1(с). (а) Знайдено відрізки ліній, які використовуються для встановлення обмежень. (б) обмежувальні кола (с) випрямлене метричне зображення

У своїй роботі ми частково використовуємо запропоновану схему побудови матриці перспективної деформації, зокрема проективну її компоненту і перетворення подібності. Ключовим параметром матриці афінної деформації, запропонованої нами, є реальний кут між лініями у фронтально-паралельній площині аркуша [6], які відповідають двом основним точкам сходу на зображенні. Також ми пропонуємо будувати афінну компоненту, використовуючи абсолютне положення точок сходу виправленого зображення на координатній площині. Про матрицю афінного перетворення як про компоненту перспективної деформації йдеться у розділі 2 цієї роботи, а про відтворення реального кута за допомогою точок сходу — у розділі 3.

1.1.3 Rectification of Camera Captured Document Images for Camera-Based OCR Technology

У роботі [8] автори пропонують метод вирівнювання фотографій документів, в основу якого закладено пошук ліній, знайдених на самому тексті. Для виділення горизонтальних ліній з тексту вони застосовують алгоритми Кенні

[9] та Хафа [10] (рис. 1.5.), а для вертикальних — метод RANSAC, який обходить всі крайні точки горизонтальних ліній і апроксимує нахил вертикальних (рис. 1.6.).



Рис. 1.5. Виявлення ліній у роботі [8]. Перше зображення — вхідне. Друге — після попередньої обробки. Третє — після застосування алгоритма Хафа

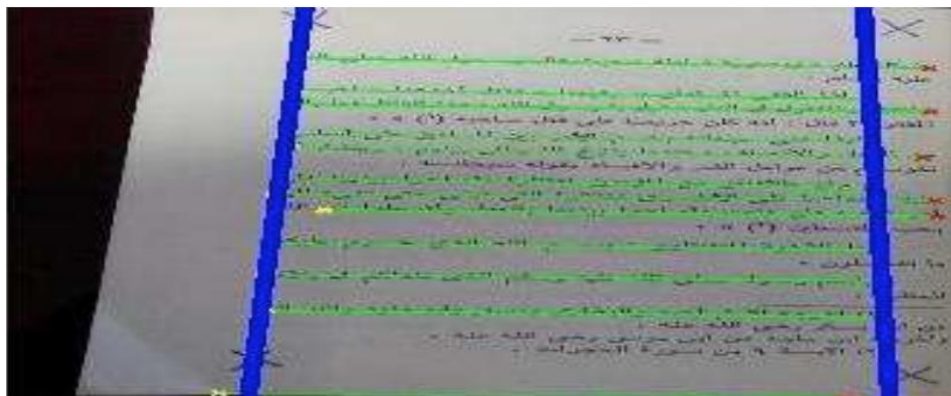


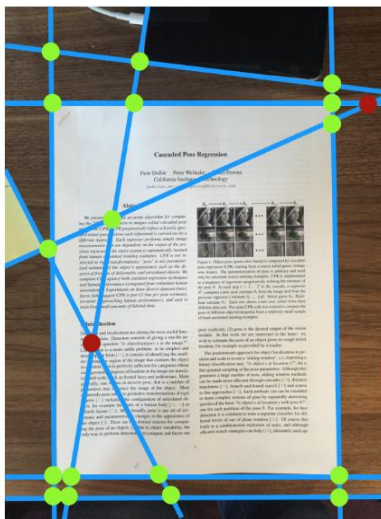
Рис. 1.6. Виявлення вертикальних ліній методом RANSAC

Обмеженням даного алгоритму є те, що він потребує написання дуже рівних рядків тексту, а також знання точних координат початку і кінця горизонтальних ліній, що мають співпадати з початком і кінцем відповідних рядків тексту. У нашій роботі ми використовуємо лінії розмітки акруші, через що перша з вищенаведених проблем не виникає, а також метод вирівнювання зображень, який використовує точки сходу паралельних ліній розмітки, що дозволяє зробити більш коректну ректифікацію.

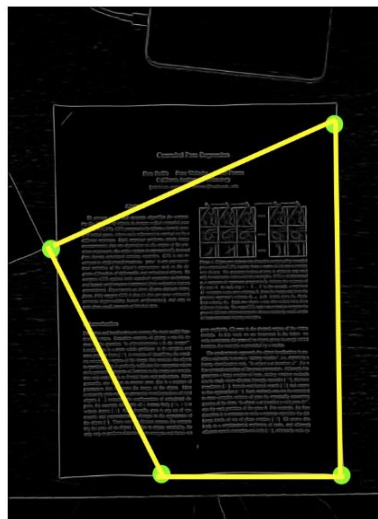
1.2 Алгоритми комерційних застосунків

Наразі існує різні комерційні застосунки, котрі дають змогу користувачам вирівнювати фотографії документів і робити їх цифрові скан-копії. Прикладами таких програм є Dropbox, PDF-scanner, Google Drive, Adobe Scan та інші. Більша частина програм з переліку вище орієнтована на пошук аркушу паперу за сприятливих умов зйомки, а саме: наявність цілого аркушу на зображенні та відсутність зайвих аркушів на фоні. Було досліджено та проаналізовано один з таких алгоритмів ректифікації фото.

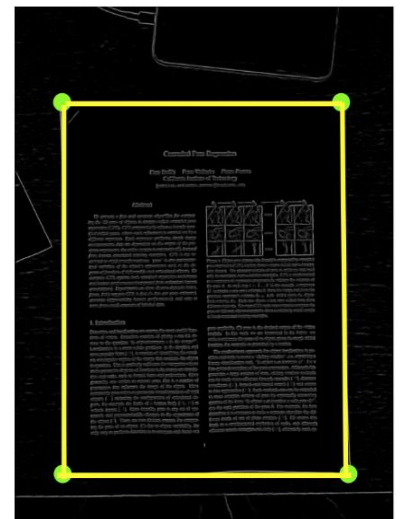
З 2016 року програмний застосунок для обробки документів Dropbox дає змогу своїм користувачам користуватися функцією вирівнювання фотографій документів [11]. Під час розробки автоматичного сканеру для смартфонів компанія керувалася припущенням, що всі документи здебільшого прямокутні. Виходячи з цього, для знаходження країв на зображенні вони використовували алгоритм Кенні та алгоритм Хафа для знаходження ліній, а далі, перебираючи точки перетину ліній, оцінювали кути документу за допомогою сумування ймовірності, передбаченої детектором країв, для пікселів вздовж його периметра [12]. Наше рішення спрямоване на уникнення залежності від даного припущення, тому ми використовуємо розмітку на аркушах, що дозволяє опрацьовувати випадки, на яких даний застосунок дав би поганий результат.



а)



б)



в)

Рис. 1.7. Знаходження країв документу алгоритмом, запропонованим компанією Dropbox. а) Знаходження ліній на зображенні та всіх їх точок перетину; б) оцінка ймовірних кутів документу, перебираючи комбінації точок перетину; в) знаходження ліній, точки перетину яких мають найбільшу ймовірність як кандидатів на краї документу

Висновок до розділу 1

У першому розділі розглянуто та проаналізовано наукові роботи та алгоритми комерційних програм, що присвячені вирівнюванню (ректифікації) зображень, зокрема зображень паперових документів. Описано їх переваги та недоліки, а також порівняно наш метод з ними. З огляду даних робіт можна побачити, що наш алгоритм, як і більшість з них, використовують класичні методи обробки зображень, такі як алгоритми Кенні та Хафа, а також метод RANSAC у різних його застосуваннях. Проте на відміну від розглянутих методів, розроблений нами дозволяє вирівнювати зображення, головні напрямки ліній на яких не є строго перпендикулярними фокальній площині камери, а можуть

перетинатися під будь-яким наперед заданим кутом. Також у випадку вирівнювання зображень документів, для нашого підходу не є обов'язковою вимогою те, що аркуш повинен бути повністю вміщеним на зображенні, чи мати рівні лінії тексту, написані на ньому.

РОЗДІЛ 2

КОМПОНЕНТИ МЕТОДУ ВИРІВНЮВАННЯ ЗОБРАЖЕНЬ РОЗЛІНОВАНИХ АРКУШІВ

У цьому розділі наведено основні математичні об'єкти та алгоритми, необхідні для опису нашого методу. Описано такі математичні методи як детектор країв Кенні [9] для знаходження країв елементів на зображенні, алгоритм Хафа [10] для знаходження геометричних елементів, у нашому випадку – прямих ліній, описано основні ідейні положення методу RANSAC (RANdom SAmple Consensus) [7] та елементи проективної геометрії, які використовуються в нашій роботі [13].

2.1 Алгоритм Кенні пошуку країв елементів

Детектор країв Кенні — це алгоритм для виявлення країв на вхідному зображенні, який складається з декількох кроків. Він був запропонований Дж. Кенні у 1961 і його задачею було реагувати на границі, але при цьому ігнорувати помилкові, точно визначати суцільну лінію границі і реагувати на кожний край один раз, що дозволяє уникнути сприйняття широких смуг зміни яскравості як сукупності границь. Зазвичай детектор Кенні використовують на одноканальних зображеннях, для зменшення обчислювальних витрат. Він включає в себе наведені нижче етапи:

1. Видалення шуму у вхідному зображенні за допомогою фільтра Гауса. Алгоритм використовує фільтр Гаусса, що діє на зображення $f(x, y)$ за допомогою оператора згортки. Застосовується двовимірна функція Гаусса $g(x, y, \delta)$ (2.1) з параметром δ , що відповідає за степінь згладжування

$$g(x, y) = \frac{1}{\sqrt{2\pi\delta}} e^{-\frac{x^2+y^2}{2\delta^2}}, \quad (2.1)$$

$$G = g(x, y, \delta) * f(x, y). \quad (2.2)$$

2. Обчислення градієнта інтенсивності пікселів зображення. Градієнти можуть мати різні напрямки, однак алгоритм використовує чотири для знаходження горизонтальних, вертикальних та діагональних країв. Градієнти визначаються за допомогою фільтра Собеля (2.3). Границя виникає, коли змінюється колір зображення, а отже, інтенсивність пікселя також змінюється.

$$G_x = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} I, \quad G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & +2 & +1 \end{bmatrix} I. \quad (2.3)$$

Далі обчислюється величина та кут напрямків градієнтів відповідно за формулами

$$|G| = \sqrt{G_x^2 + G_y^2},$$

$$\theta = \tan^{-1}\left(\frac{G_y}{G_x}\right)$$

і обирається кут θ близький до 0, 45, 90 або 135 градусів.

3. Подавлення не максимумів. На цьому кроці значення пікселя залишається, якщо його значення є найбільшим серед значень сусідніх пікселів, взятих у напрямку його градієнта, інакше значення пікселя прирівнюється до нуля. Таким чином, знайдені на попередньому кроці краї стають тоншими та чіткішими.

4. Подвійна порогова фільтрація. Збереження значень пікселів, що перевищують верхній поріг (так звані “сильні” краї), і прирівняння їх значень до нуля, якщо ті є нижчими за нижнє порогове значення. Для тих пікселів, значення яких знаходиться між двома порогоми застосовується функція гістерезису, яка зануляє ті пікселі, які знаходяться далеко від “сильних” країв.

2.2 Алгоритм Хафа для знаходження ліній

Перетворення Хафа — це алгоритм, що використовується для виділення на зображенні об'єктів певної форми, яку можна задати за допомогою параметрів. Оскільки він вимагає, щоб бажані характеристики були вказані в деякій параметричній формі, класичне перетворення Хафа найчастіше використовується для виявлення регулярних кривих, таких як лінії, кола, еліпси тощо. Для зменшення обчислювальних витрат, на вхід даного алгоритму зазвичай подають мапу країв елементів зображення.

Розглянемо алгоритм Хафа для знаходження прямих ліній на зображенні. У класичному описі алгоритму [10] пряму $y = mx + c$ представлено у просторі параметрів як точку (b, c) (рис. 2.1). Однак за такого представлення виникає проблема з вертикальними лініями, адже для них параметр m приймає значення нескінченності. Таким чином, з у своїй роботі Дуда і Харт [14] запропонували визначати рівняння прямої для алгоритму Хафа у полярних координатах:

$$r = x \cos(\theta) + y \sin(\theta),$$

де $r \geq 0$ — відстань до прямої від початку координат, $\theta \in (0, 2\pi)$ — кут нахилу прямої r до додатного напрямку осі x . За даного представлення ліній кожній точці (x_0, y_0) в просторі зображення буде відповідати синусоїда у просторі параметрів (r, θ) , яка описує всі можливі параметри прямих, які проходять через дану точку (рис 2.2).

У такому випадку, якщо точки (x_1, y_1) і (x_2, y_2) лежать на одній прямій у просторі зображення, яка задається параметрами r_0, θ_0 , то їх синусоїди у просторі параметрів будуть перетинатися у точці з координатами (r_0, θ_0) .

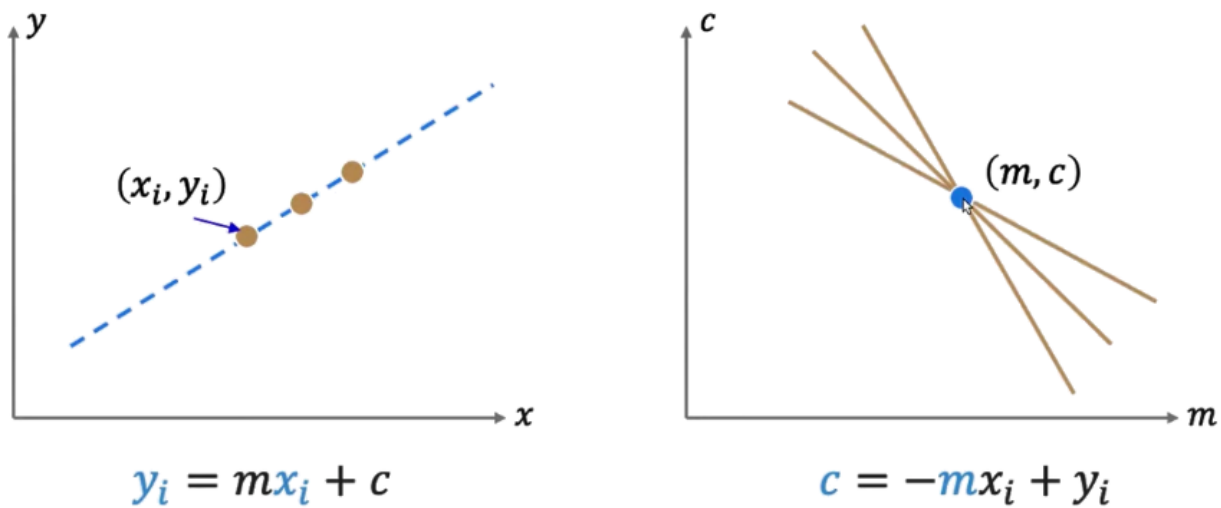


Рис. 2.1 Представлення прямої у просторі зображення (X, Y) і у просторі параметрів (M, C) відповідно

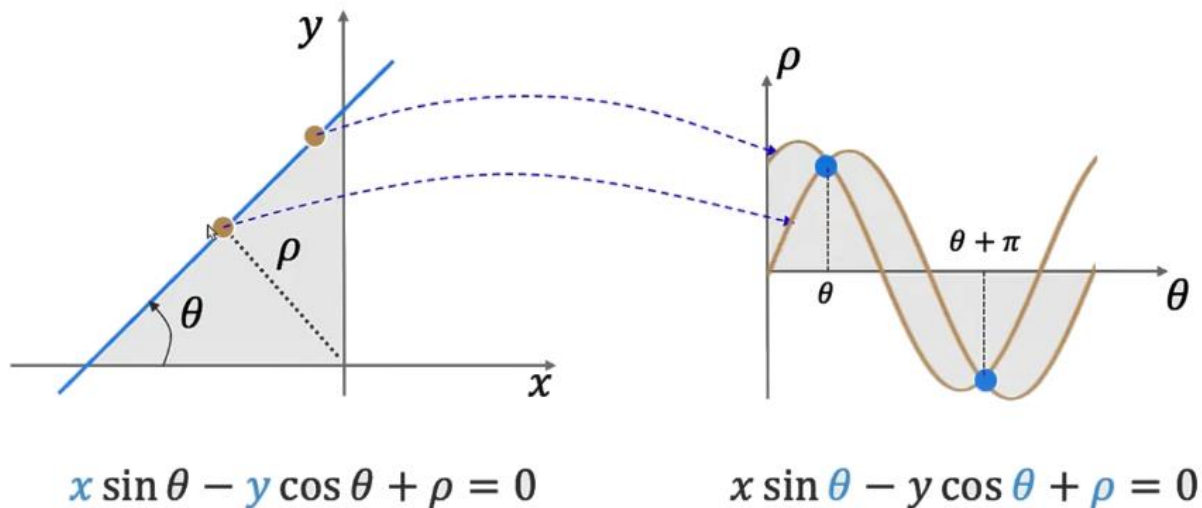


Рис. 2.2. Представлення прямої у просторі зображення (X, Y) і у просторі параметрів полярних координат (r, θ) відповідно

Це означає, що лінію можна виявити, знайшовши кількість перетинів між кривими. Чим більше кривих перетинаються у певній точці, тим лінія, представлена цим перетином, має більше точок. Загалом, ми можемо визначити поріг мінімальної кількості перетинів, необхідних для виявлення лінії.

Отже, алгоритм Хафа відстежує перетин синусоїд кожної точки зображення у просторі параметрів. Якщо кількість перетинів в деякій точці (r_θ, θ) перевищує певний поріг, то він визначає це як лінію з параметрами (r_θ, θ) .

2.3 Метод RANSAC (Random sample consensus)

Метод RANSAC — це ітеративний метод визначення математичної моделі на основі невеликої вибірки з набору даних, який містить викиди. Він виявляє викиди та будує модель використовуючи лише релевантні елементи датасету.

Даний метод був вперше продемонстрований у 1981 році М. Фішером та Р. Боллесом [7]. В порівнянні з традиційними моделями машинного навчання,

RANSAC не використовує всю доступну вибірку для оцінки параметрів моделі, а застосовує невелику кількість її елементів на кожному своєму кроці. Точність методу на пряму залежить від кількості його ітерацій, однак автори пропонують обмежувати її знизу значенням

$$k = \frac{\log(1 - z)}{\log(1 - b)},$$

де z — ймовірність того, що модель знайде підвибірку з n елементів, де не буде жодного викиду, $b = w^n$, w — ймовірність того, що будь-яка вибрана точка знаходиться в межах допустимої помилки моделі. На рис. 2.3. можна побачити результат алгоритму на основі методу RANSAC в порівнянні з класичною лінійною регресією для набору даних, який складається з двовимірних точок, які мають лінійну залежність своїх координат і серед яких є явні викиди.

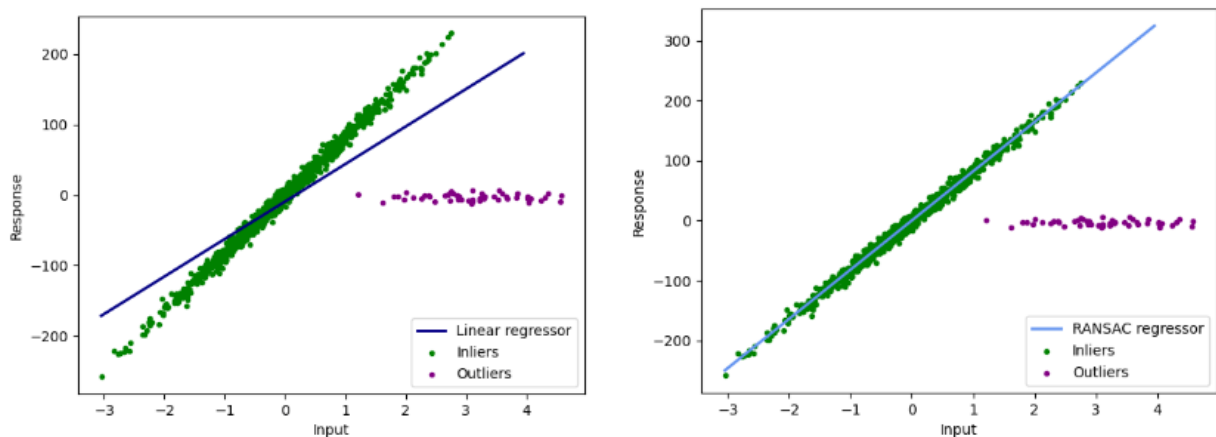


Рис. 2.3. – Порівняння результату на основі методу RANSAC з результатом алгоритму лінійної регресії [17]

Опишемо основні кроки цього методу. На кожній ітерації алгоритму, побудованого на основі RANSAC виконують наступне:

1. Вибирають випадкову підмножину даних з загальної вибірки із попередньо визначеним розміром n .
2. На цих даних тренують модель, яка повинна описувати залежність даних. Це може бути будь-яка модель, наприклад, проста лінійна регресія.
3. Після встановлення моделі перевіряють, чи відповідають їй інші точки. Для цього використовують параметр t як деяке порогове значення для оцінки помилки моделі. Таким чином, створюється множина A точок, що задовольняють модель, створеній на певній ітерації.
4. Якщо потужність множини A більша за деяке задане значення k , це означає, що дана модель є кандидатом на правильну.
5. Серед кандидатів на правильну модель обрати ту, для якої множина A має найбільшу потужність.

2.4 Елементи проективної геометрії

У цьому підрозділі висвітлено визначення та правила проективної геометрії, які використовуються в нашій роботі для побудови матриці ректифікації [13].

2.4.1 Задання прямих і точок в проективному просторі

Визначення прямих в проективному просторі

Пряма на площині представлена рівнянням $ax + by + c = 0$, де різні значення параметрів a , b і c породжують різні лінії. Таким чином, пряму можна представити вектором $(a, b, c)^T$. Відповідність між прямими і векторами $(a, b, c)^T$ не є однозначною, оскільки прямі $ax + by + c = 0$ і $(ka)x + (kb)y + kc = 0$ однакові для будь-якої ненульової константи k . Таким чином, вектори $(a, b, c)^T$ і $k(a, b, c)^T$ представляють ту саму лінію

для будь-якого ненульового k . Такі вектори, пов'язані константним масштабуванням, відносять до одного класу еквівалентності. Клас векторів у цьому відношенні еквівалентності відомий як однорідний вектор $(a, b, c)^T$ і будь-який вектор $k(a, b, c)^T$ є представником цього класу. Множина класів еквівалентності векторів в $R^3 \setminus (0, 0, 0)^T$ утворює проєктивний простір P^2 .

Визначення точок в проєктивному просторі

Точка $p = (x, y)^T$ лежить на прямій $l = (a, b, c)^T$ тоді і тільки тоді, коли $ax + by + c = 0$. Це можна записати в термінах скалярного добутку векторів $(x, y, 1)(a, b, c)^T = (x, y, 1)l = 0$, що представляє точку $(x, y)^T$ в R^2 як 3-вектор шляхом додавання кінцевої координати рівної 1. Варто звернути увагу, що для будь-якої ненульової константи k і лінії l рівняння $(kx, ky, k)l = 0$, тоді і тільки тоді, коли $(x, y, 1)l = 0$. Тому природно розглядати множину векторів $(kx, ky, k)^T$ для змінних значень k як представлення точки $(x, y)^T$ в просторі P^2 .

Таким чином, так само як лінії, точки зображуються однорідними векторами. Довільний вектор, що представляє точку в просторі однорідних координат, має вигляд $x = (x_1, x_2, x_3)^T$, що відповідає точці $(\frac{x_1}{x_3}, \frac{x_2}{x_3})^T$ в просторі R^2 .

Перетин прямих в проєктивному просторі

Дано дві прямі $l = (a, b, c)^T$ і $l' = (a', b', c')^T$, ми хочемо знайти їх перетин. Визначимо вектор як $x = l \times l'$. З тотожності мішаного добутку $l(l' \times l) = l'(l \times l) = 0$ можна побачити, що $l^T x = l'^T x = 0$. Таким чином, якщо x – це точка, то вона лежить на обох прямих l і l' , а отже, є перетином двох прямих.

Задання прямої через дві точки в проєктивному просторі

Вираз для прямої, що проходить через дві точки x і x' , можна отримати аналогічно. Візьмемо дві точки x і x' і визначимо вектор $l = x \times x'$. З тотожності мішаного добутку $x(x \times x') = x'(x \times x') = 0$ можна переконатися, що обидві точки x і x' лежать на прямій l . Таким чином, пряма, яка проходить через дві точки x і x' , дорівнює $l = x \times x'$.

2.4.3 Точка і пряма на нескінченності

Розглянемо дві паралельні лінії $ax + by + c = 0$ і $ax + by + c' = 0$. Вони представлені векторами $l = (a, b, c)^T$ і $l' = (a, b, c')^T$ для яких перші дві координати однакові. Обчисливши точку перетині двох прямих способом, наведеним вище маємо $l \times l' = p = (bc' - bc, ca - c'a, ab - ab) = (c' - c)(b, -a, 0)^T$. І, нехтуючи масштабуючим множником $(c' - c)$, отримуємо точку $(b, -a, 0)^T$. Далі, переведемо знайдену точку в декартову систему координат, поділивши на останню координату. Отримуємо точку $(\frac{b}{0}, -\frac{a}{0}, \frac{0}{0})^T$, яка показує, що дві паралельні лінії перетинаються на нескінченності.

Означення 2.1

Точка в проєктивному просторі P^2 з координатами $(x, y, 0)$ називається ідеальною точкою, або точною на нескінченності. Множину всіх ідеальних точок можна записати $(x_1, x_2, 0)^T$, де конкретна точка визначається співвідношенням $x_1 : x_2$.

Варто звернути увагу, що набір всіх ідеальних точок лежить на одній прямій, що називається прямою на нескінченності і позначається вектором

$l_{\infty} = (0, 0, 1)^T$. Перевірити це можна, порахувавши скалярний добуток $(0, 0, 1) (x_1, x_2, 0)^T = 0$.

На рис. 2.4. Наведено модель проєктивного простору, яка дає уявлення про взаємне розташування вищенаведених елементів в ньому.

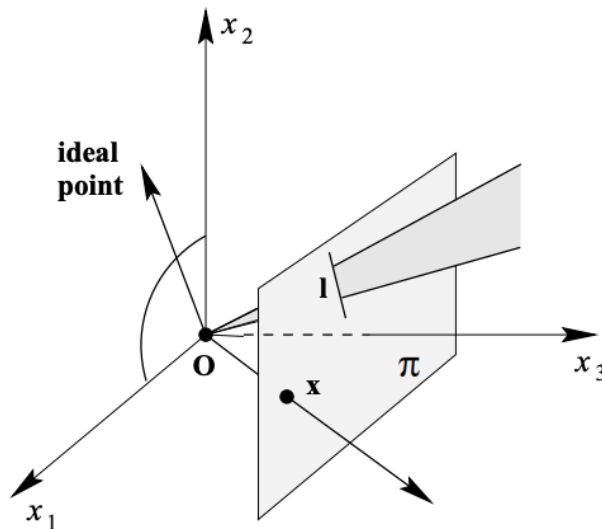


Рис. 2.4 – Модель проєктивного простору [13].

2.4.4 Перспективні перетворення

Означення 2.2

Проективність (проективне перетворення або гомографія) — це бієктивне відображення H з P^2 в себе таке, що три точки x_1, x_2, x_3 лежать на одній прямій тоді і тільки тоді, коли $H(x_1), H(x_2), H(x_3)$ лежать на одній прямій.

Означення 2.3

Планарне проективне перетворення — це лінійне перетворення на однорідних векторах, представлених невиродженою матрицею H :

$$\begin{pmatrix} x'_1 \\ x'_2 \\ x'_3 \end{pmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix},$$

$$\text{або } x' = Hx.$$

H є однорідною матрицею, оскільки при її застосуванні на однорідних векторах, значення мають лише співвідношення між елементами матриці. Проективне перетворення проектує кожний об'єкт зображення в проективно еквівалентний об'єкт, залишаючи всі його проективні властивості інваріантними. Існує вісім незалежних елементів H , це означає, що проективне перетворення має вісім ступенів свободи.

Матрицю гомографії можна декомпозувати на декілька груп окремих перетворень, що мають різні геометричні властивості. Коротко опишемо кожен з груп.

Група 1. Ізометричні перетворення. До ізометричних відносять перетворення на площині R^2 , що зберігають евклідову відстань між елементами. До них відносяться обертання та зсув. Вони мають три ступені свободи, один відповідає за обертання і два за зсув. Даний клас перетворень визначається наступним чином:

$$\begin{pmatrix} x'_1 \\ x'_2 \\ x'_3 \end{pmatrix} = \begin{bmatrix} \alpha \cos(\theta) & -\sin(\theta) & t_1 \\ \alpha \sin(\theta) & \cos(\theta) & t_2 \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix},$$

де $\alpha = +- 1$ – параметер, що відповідає за орієнтацію відносно початку координат, матриця 2×2 у верхньому лівому куті – це матриця повороту, вектор (t_1, t_2) – вектор переносу.

Група 2. Перетворення подібності. Перетворення подібності — це ізометрія, що складається з ізотропного масштабування. Перетворення подібності також є відоме як рівноформне перетворення, оскільки воно зберігає форму об'єкта. Воно має чотири ступені свободи, до обертання і зміщення ще додається розширення по осям Ox та Oy . Перетворення подібності задається матрицею

$$\begin{pmatrix} x'_1 \\ x'_2 \\ x'_3 \end{pmatrix} = \begin{bmatrix} q \cos(\theta) & -q \sin(\theta) & t_1 \\ q \sin(\theta) & q \cos(\theta) & t_2 \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix},$$

де q – параметер, що відповідає за ізотропне масштабування.

Група 3. Афінне перетворення. Афінне перетворення — це несингулярне лінійне перетворення. Даному виду перетворень відповідає неізотропне масштабування, тобто нерівномірне розширення і стиск вздовж осей координат. Воно має шість ступенів свободи і задається матрицею

$$\begin{pmatrix} x'_1 \\ x'_2 \\ x'_3 \end{pmatrix} = \begin{bmatrix} a_{11} & a_{12} & t_1 \\ a_{21} & a_{22} & t_2 \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix},$$

де матриця 2×2 у верхньому лівому куті – несингулярна матриця.


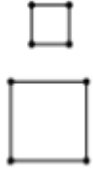

Група 4. Перспективне перетворення. Це загальне несингулярне лінійне перетворення однорідних координат. Це узагальнює афінне перетворення, яке є

композицією загального неособливого лінійного перетворення неоднорідних координат і трансляції. Проективне перетворення можна розкласти на ланцюжок перетворень, де кожна матриця в ланцюжку представляє перетворення вище за ієрархією, ніж попередній.


$$H = \begin{bmatrix} sR & t \\ 0^T & 1 \end{bmatrix} \begin{bmatrix} K & 0 \\ 0^T & 1 \end{bmatrix} \begin{bmatrix} I & 0 \\ v^T & v \end{bmatrix} = \begin{bmatrix} A & t \\ v^T & v \end{bmatrix}.$$

В таблиці 2.1 представлено основні характеристики вищеперелічених перетворень.

Таблиця 2.1 Характеристики компонент перспективного перетворення

Перетворення	Матриця	Трансформація	Інваріанти
Ізометричні перетворення	$\begin{bmatrix} \alpha \cos(\theta) & -\sin(\theta) & t_1 \\ \alpha \sin(\theta) & \cos(\theta) & t_2 \\ 0 & 0 & 1 \end{bmatrix}$		Довжина, площа, паралельність прямих
Перетворення подібності	$\begin{bmatrix} q \cos(\theta) & -q \sin(\theta) & t_1 \\ q \sin(\theta) & q \cos(\theta) & t_2 \\ 0 & 0 & 1 \end{bmatrix}$		Відношення довжин, кути, паралельність прямих
Афінне перетворення	$\begin{bmatrix} a_{11} & a_{12} & t_1 \\ a_{21} & a_{22} & t_2 \\ 0 & 0 & 1 \end{bmatrix}$		Паралельність прямих, відношення площ, відношення довжин паралельних сегментів

Продовження таблиці 2.1

Перетворення	Матриця	Трансформація	Інваріанти
Перспективне перетворення	$\begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix}$		Перехресне співвідношення чотирьох колінеарних точок

Висновок до розділу 2

У цьому розділі наведено основні математичні об'єкти та методи, необхідні для опису нашого методу. Описано такі математичні методи як детектор країв Кенні, за допомогою якого можна знаходити краї елементів на зображенні, алгоритм Хафа для знаходження регулярних геометричних об'єктів, у нашому випадку прямих ліній, описано основні ідейні положення методу RANSAC та надано спосіб його застосування для вирівнювання аркушів у клітинку або у косу лінію. Також наведено елементи проективної геометрії, які знадобляться для опису нашого алгоритму. До них належать визначення прямих і точок в проективному просторі, перетин двох прямих, задання прямої через дві точки, означення точки та прямої на нескінченності. Наведено означення перспективного перетворення (гомографії), та описано групи перетворень, що є його складовими. Ці знання будуть необхідні для побудови матриці ректифікації зображень аркушів у наступному розділі.

РОЗДІЛ 3

ОПИС АЛГОРИТМУ РЕКТИФІКАЦІЇ РОЗЛІНОВАНИХ АРКУШІВ

У даному розділі описано створений нами метод ректифікації фотографій, на яких зображено аркуші паперу з розміткою в клітинку або у косу лінію. Саме наявність ліній дозволяє коректно вирівняти зображення, незалежно від того, чи потрапляють контури аркушу на фото, чи ні. Підхід використовує чотири основні кроки: знаходження ліній на зображенні, кластеризація ліній, побудова матриці перспективного перетворення для вирівнювання кутів зображення та обрізання зображення по обмежувальній рамці.

Для визначення типу розмітки та подальшого вирівнювання аркушу нам необхідно знайти лінії на зображенні. Для цього необхідно зробити бінаризацію зображення та застосувати детектор границь Кенні [9]. На отриманому чорно-білому зображенні країв елементів застосовуємо алгоритм Хафа [14] для пошуку ліній.

Наступним кроком є кластеризація знайдених ліній та знаходження їх точок сходу. В залежності від розмітки, лінії кластеризуються на горизонтальні, вертикальні або під кутом. За допомогою методу RANSAC [7] ми знаходимо дві точки сходу, та відповідні їм групи ліній.

За допомогою знайдених точок сходу ми будуємо матрицю перспективного перетворення та застосовуємо її до зображення. Отриманий результат розтягуємо у обмежувальній рамці. Потім, якщо це має сенс, використовуючи крайні лінії, ми обрізаємо зображення і отримуємо на новому лише вирівняне зображення аркуша.

3.1 Знаходження ліній на зображенні

Для вдалого пошуку ліній нам необхідна якісна бінаризація. В цій роботі ми не автоматизуємо її пошук, а обираємо ту, що показала себе найкраще у ході проведених експериментів.

Був обраний метод адаптивної бінаризації зображення [18]. Він полягає у пошуку порогового значення поділу на чорний та білий колір не на всьому зображенні, а у локальних областях кожного пікселя. Це дає змогу більш точно бінаризувати вхідне зображення, зважаючи на незначні деталі та перепади світла у певних фрагментах фото.

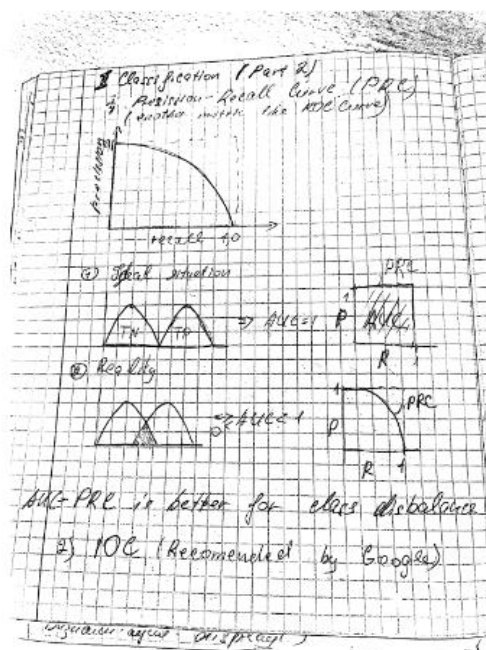


Рис 3.1 Приклад бінаризованого зображення

На бінаризованому зображенні виділяємо контури за допомогою алгоритма Кенні та запускаємо алгоритм Хафа пошуку ліній, реалізація яких використовувалась з бібліотеки OpenCV. Наразі вибір параметрів для цих методів не є автоматизованими, тому для кожного зображення слід вручну

підбирати такі параметри як мінімальний поріг довжини лінії для алгоритму Хафа і верхнє та нижнє порогові значення функції гістерезису для алгоритму Кенні.

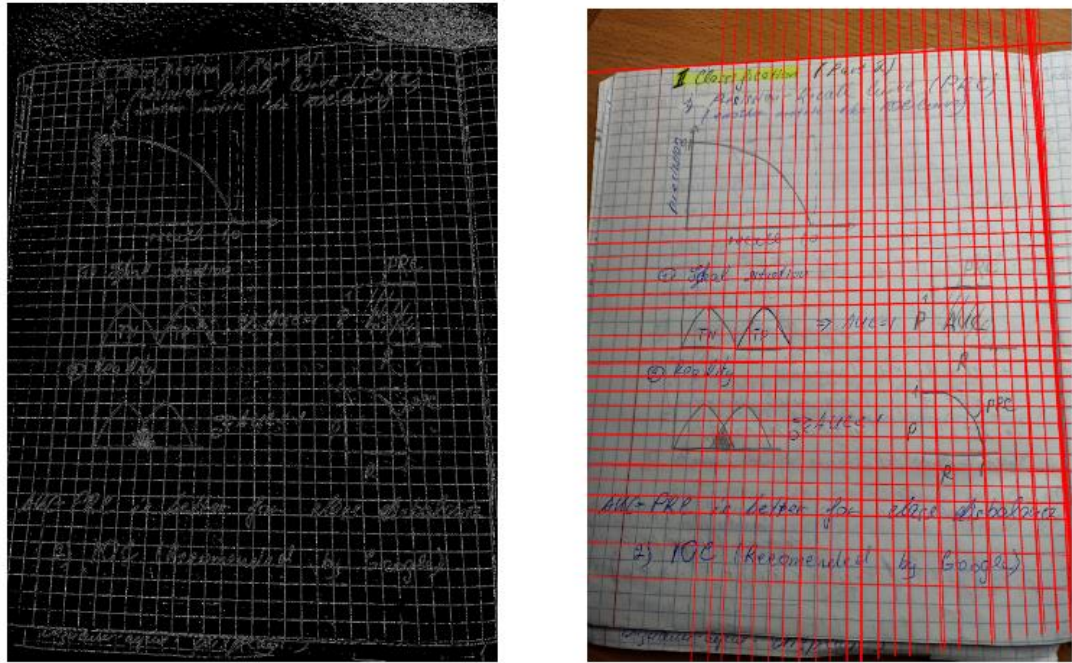


Рис 3.2 Пошук контурів та ліній на зображенні за допомогою алгоритмів Кенні та Хафа відповідно

3.2 Групування (кластеризація) ліній

Опишемо алгоритм кластеризації ліній розмітки аркушу та знаходження їх точок сходу. Для цього скористаємося математичним апаратом проективної геометрії, який наведено у другому розділі цієї роботи, а саме формулу перетину прямих, та означенням ідеальної точки (точки на нескінченності). Також наведемо означення точки сходу для розуміння подальших кроків нашого методу.

Означення 3.1

Точка сходу — це точка на площині перспективного зображення, в якій сходяться двовимірні перспективні проєкції ліній, паралельних у тривимірному просторі.

3.2.1 Оцінка відстані між точками у проєктивній площині

Зважаючи на зашумленість даних, ми хочемо перевірити, чи дві пари прямих l_1, l_2 та l_3, l_4 у перспективному просторі перетинаються в одній точці. Для цього можна порівнювати відстань між точками перетину $p_1 = l_1 \times l_2, p_2 = l_3 \times l_4$ з деяким пороговим значенням ε .

Якщо дві прямі не паралельні, їх точка перетину має ненульову третю координату. Тоді ми можемо отримати точки в евклідовому просторі R^2 як $p_i' = \frac{p_i}{p_i^z}$, і далі порівняти відстань між точками у звичайний спосіб.

$$(l_1 \# l_2) \wedge (l_3 \# l_4) \Rightarrow (p_1^z \neq 0) \wedge (p_2^z \neq 0) \Rightarrow \left\| \frac{p_1}{p_1^z} - \frac{p_2}{p_2^z} \right\| < \varepsilon$$

Якщо прямі паралельні, третя координата точки їх перетину близька до нуля (точка на нескінченності), у цьому випадку ми повинні порівнювати напрями двох точок

$$(l_1 \parallel l_2) \wedge (l_3 \parallel l_4) \Rightarrow (p_1^z \approx p_2^z \approx 0) \Rightarrow \left\| \frac{p_1}{\|p_1\|} - \frac{p_2}{\|p_2\|} \right\| < \varepsilon$$

Отже, перед нами постає питання: як порівняти відстань між двома точками в однорідних координатах, без необхідності перевірки їх третіх координат?

Пронормуємо точки $q_1 = \frac{p_1}{\|p_1\|}$, $q_2 = \frac{p_2}{\|p_2\|}$, щоб отримати пару одиничних векторів у 2-сфері тривимірного евклідового простору. Тепер ми маємо два вектори в одній півсфері S^2 і можемо знайти кут між ними:

$$\theta = \arccos(q_1 \cdot q_2) \quad [15].$$

Значення θ дає нам можливість оцінити віддаленість точок перетину пар прямих і не залежить від того, чи є точки q_1 та q_2 точками на нескінченності, чи ні.

3.2.2 Виділення груп співнапрямлених ліній та їх точок сходу

Наразі ми маємо всі необхідні геометричні об'єкти та їх властивості. Опишемо метод знаходження найкращої точки сходу та ліній, які у ній перетинаються. Для цього застосуємо метод RANSAC, який описано у розділі 2 даної роботи.

Візьмемо три довільні лінії l, l', l^* з набору V всіх знайдених алгоритмом Хафа ліній і знайдемо першу точку сходу за формулою

$$v_1 \in \underset{\substack{p = \frac{l \times l'}{\|l \times l'\|} \\ \forall l, l' \in V}}{\operatorname{argmax}} \sum_{l^* \in V} \left[\left| \cos^{-1} \left(p \cdot \frac{l \times l^*}{\|l \times l^*\|} \right) \right| < \varepsilon \right]. \quad (3.1)$$

Тоді лінії, які проходять повз v_1 на відстані менше ε , визначаються як

$$L_1 = \left\{ l : \left| \cos^{-1} \left(v_1^T \cdot \frac{l}{\|l\|} \right) \right| < \varepsilon \right\}. \quad (3.2)$$

Коли перша точка сходу оцінена, отримуємо новий набір ліній шляхом видалення групи L_1 з масиву V і повторюємо операції (3.1) та (3.2) на ньому для отримання другої точки сходу.

Тепер ми маємо дві точки сходу v_1 і v_2 , і два набори ліній, що відповідають цим точкам – L_1 і L_2 . Приклад результату роботи методу RANSAC наведено на рис. 3.3. На першому зображенні нарисовано всі лінії, знайдені алгоритмом Хафа, на другому - групи горизонтальних і вертикальних ліній.

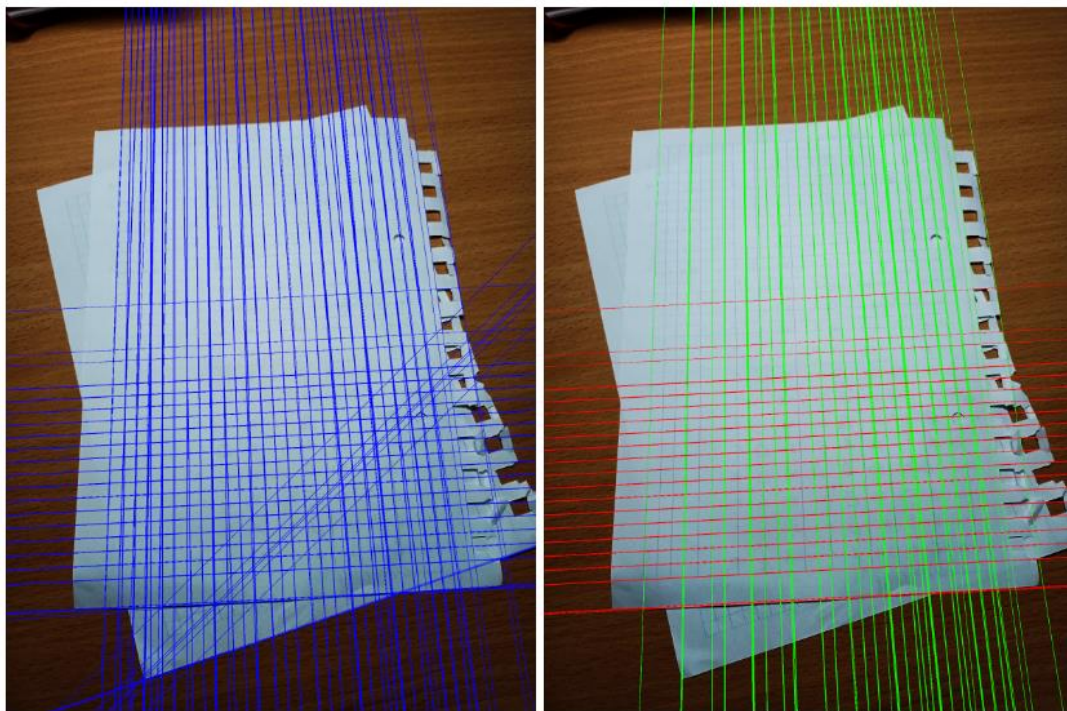


Рис. 3.3 Результат роботи методу RANSAC по виділенню груп ліній на прикладі аркуша в клітинку. Червоним і зеленим кольорами позначено горизонтальні і вертикальні групи ліній відповідно.

3.3 Побудова матриці перспективного перетворення

На цьому кроці ми використовуємо знайдені точки сходу та лінії для переведення зображення у форму, зручну для сприйняття користувачем.

Нехай ми маємо дві групи ліній і відповідні їм точки сходу v_1 і v_2 у правій системі координат. При цьому знаємо, що на площині вирівняного зображення лінії у межах кожної з груп повинні бути паралельними між собою, і перша група — це група горизонтальних ліній, а друга — вертикальних. Тобто треба викривити зображення таким чином, щоб одна з точок сходу перейшла в точку з координатами $(x, 0, 0)$ і відповідала горизонтальному напрямку, а інша — у точку з координатами $(0, y, 0)$. Для цього побудуємо таку матрицю R повороту таку, що

$$Rv_1 = (x, 0, 0)^T$$

$$Rv_2 = (0, y, 0)^T.$$

З цього випливає, що точка v_1 перпендикулярна до векторів, що відповідають другому і третьому рядкам матриці R , а v_2 — першому і третьому. Отже, R може бути задана наступним чином:

$$R(v_1, v_2) = \begin{pmatrix} v_2 \times (v_1 \times v_2) \\ (v_1 \times v_2) \times v_1 \\ (v_1 \times v_2) \end{pmatrix}.$$

Дана матриця має недолік: оскільки R описує лінійне перетворення і $\text{rang}(R) = 3$, а ядро повнорангового лінійного оператора є множина, що складається з нульового вектора, точка зображення з координатами $(0, 0)$ після застосування оператора R залишиться без викривлень, а чим далі точки

знаходиться від початку координат, тим далі вони будуть знаходитися від своїх початкових положень після перетворення R . Аби зменшити нерівномірність викривлення зображення, перенесемо початок координат у його центр перед застосуванням оператора R , а після трансформацій повернемо у початкову точку.

Повне перетворення для перенесення зображення у фронтально-паралельну площину має вигляд

$$H = T^{-1}R(v_1, v_2)T,$$

де T – це матриця зсуву центра зображення у початок координат.

Даний підхід можна природним чином розширити до випадку вирівнювання аркуша в косу лінію, де кут між групами ліній фронтально-паралельній площині є довільний кут α . Для цього візьмемо точку v_2 , яка є точкою сходу косих ліній,

повернемо її на кут $\left(\frac{\pi}{2} - \alpha\right)$ проти годинникової стрілки і позначимо точкою v'_2 :

$$v'_2 = \begin{pmatrix} \cos\left(\frac{\pi}{2} - \alpha\right), & \sin\left(\frac{\pi}{2} - \alpha\right), & 0 \\ -\sin\left(\frac{\pi}{2} - \alpha\right), & \cos\left(\frac{\pi}{2} - \alpha\right), & 0 \\ 0, & 0, & 1 \end{pmatrix} \begin{pmatrix} (v_2)_x \\ (v_2)_y \\ (v_2)_z \end{pmatrix}$$

Таким чином отримуємо прямий кут між точками v_1 та v'_2 . Після цього, застосуємо до зображення перспективне перетворення $H' = T^{-1}R(v_1, v'_2)T$.

Деформація зображення за допомогою H (H') відновлює паралельність прямих, які є паралельними фронтально-паралельній площині та відтворює

правильний кут між двома групами ліній з мінімальним викривленням зображення.

3.4 Знаходження обмежувальної рамки

Для того, щоб отримати правильний формат перетвореного зображення, знайдемо його обмежувальну рамку (bounding box). Для цього застосуємо відображення $H(H')$ до координат кутових пікселів початкового у проєктивному просторі:

$$H(0,0,1)^T = HA = A',$$

$$H(0,n,1)^T = HB = B',$$

$$H(m,n,1)^T = HC = C',$$

$$H(m,0,1)^T = HD = D',$$

де A, B, C, D — координати кутових пікселів зображення до трансформації, A', B', C', D' — координати кутових пікселів зображення після трансформації, m, n — ширина і висота початкового зображення відповідно.

Кути прямокутника, у який вписане перетворене зображення з дотиком в точках A', B', C', D' мають такі координати (порядок точок зазначений з лівого верхнього кута проти годинникової стрілки):

$$P_1 = \left(\min\left(\frac{A'x}{A'z}, \frac{B'x}{B'z}, \frac{C'x}{C'z}, \frac{D'x}{D'z}\right), \min\left(\frac{A'y}{A'z}, \frac{B'y}{B'z}, \frac{C'y}{C'z}, \frac{D'y}{D'z}\right) \right),$$

$$P_2 = \left(\min\left(\frac{A'x}{A'z}, \frac{B'x}{B'z}, \frac{C'x}{C'z}, \frac{D'x}{D'z}\right), \max\left(\frac{A'y}{A'z}, \frac{B'y}{B'z}, \frac{C'y}{C'z}, \frac{D'y}{D'z}\right) \right),$$

$$P_3 = \left(\max\left(\frac{A'x}{A'z}, \frac{B'x}{B'z}, \frac{C'x}{C'z}, \frac{D'x}{D'z}\right), \max\left(\frac{A'y}{A'z}, \frac{B'y}{B'z}, \frac{C'y}{C'z}, \frac{D'y}{D'z}\right) \right),$$

$$P_4 = \left(\max\left(\frac{A'x}{A'z}, \frac{B'x}{B'z}, \frac{C'x}{C'z}, \frac{D'x}{D'z}\right), \min\left(\frac{A'y}{A'z}, \frac{B'y}{B'z}, \frac{C'y}{C'z}, \frac{D'y}{D'z}\right) \right).$$

Ширина і довжина прямокутника дорівнюють відповідно $m' = P_4^x - P_1^x$, $n' = P_2^y - P_1^y$.

На рисунку 3.4 зображено приклад знайденого bounding box і пояснено основні його елементи.

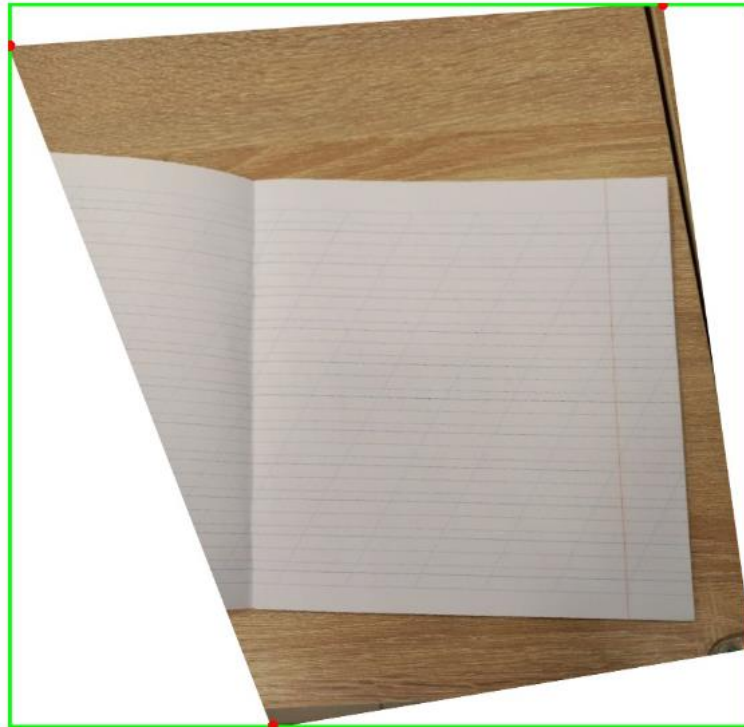


Рис. 3.4 Зображення bounding box на вирівняному зображенні. Червоні точки – точки дотику до рамки (A' , B' , C' , D'), зелений прямокутник – обмежувальна рамка.

Маючи параметри даної обмежувальної рамки, можна оцінити гомографічну матрицю Q , яка дозволяє транслювати її у початок координат. Оцінку цієї матриці було зроблено за допомогою методу `getPerspectiveTransform` з бібліотеки OpenCV [16]. Він приймає на вхід чотири пари відповідних точок вхідного і

результуючого зображення і на їх основі створює матрицю перспективного перетворення.

3.5 Виділення аркуша з фону

Після того, як зображення аркушу вирівняно, знайдено його обмежувальну рамку та перенесено її у початок координат, ми пропонуємо виділити аркуш з загального фону. Цей крок є сенс робити, якщо на фото окрім документу потрапив, наприклад, стіл, або якщо на зображенні через зміну форми після ректифікації з'явилося багато місця на зображенні, котре не зайняте аркушом, як показано на рис. 3.5.

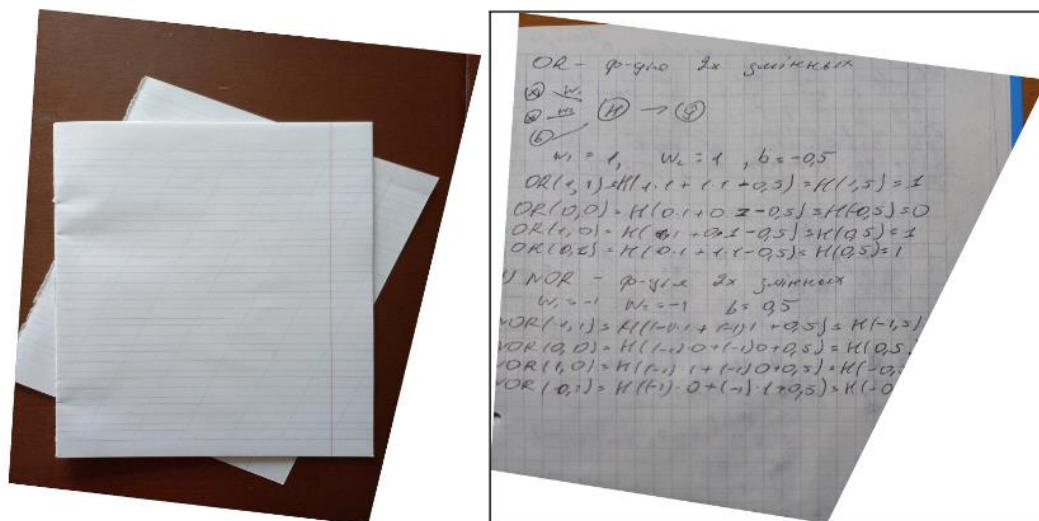


Рис. 3.5. Приклади зображень на яких слід виділяти аркуш. Фото зліва – окрім основного аркушу потрапив стіл та підкладений аркуш. Фото справа – після ректифікації зображення з'явилося багато порожнього фону.

Обрізання фото ми пропонуємо зробити наступним чином. Повернемося до кроку, на якому ми знаходили точки сходу, та скористаємося знайденими відповідними ним групами ліній L_1, L_2 . Серед груп горизонтальних і вертикальних (косих) ліній знайдемо ті, які займають в них крайні положення

відповідно за формулами, беручи до уваги те, що початок координат знаходиться у лівому верхньому куті зображення:

$$l_{top} = \operatorname{argmin}_{l \in L_1} \left(-\frac{l_z}{l_y} \right),$$

$$l_{left} = \operatorname{argmin}_{l \in L_2} \left(-\frac{l_z}{l_x} \right),$$

$$l_{bottom} = \operatorname{argmax}_{l \in L_1} \left(-\frac{l_z}{l_y} \right),$$

$$l_{right} = \operatorname{argmax}_{l \in L_2} \left(-\frac{l_z}{l_x} \right),$$

де l_{top} , l_{bottom} відповідно крайні верхня і нижня горизонтальна лінії, l_{left} , l_{right} — крайні ліва і права вертикальні (косі). Далі знайдемо точки перетину крайніх ліній для виявлення кутових точок фрагменту зображення, який буде виділено. Оскільки точка перетину ліній у проєктивному просторі — це їх векторний добуток, то кутові точки можна знайти за формулами

$$A = l_{top} \times l_{left}, \quad B = l_{left} \times l_{bottom},$$

$$C = l_{bottom} \times l_{right}, \quad D = l_{right} \times l_{top},$$

де точки A , B , C , D перелічені від лівого верхнього кута фрагменту проти годинникової стрілки.

Оскільки лінії з наборів L_1 , L_2 були виявлені на викривленому зображенні, треба перенести точки A , B , C , D у відповідні точки ректифікованого зображення. Щоб зробити це, слід до кожної з них застосувати послідовність

проективних перетворень QH , якщо L_1, L_2 — це групи горизонтальних і вертикальних ліній, або QH' , якщо групи горизонтальних і косих.

Нехай маємо координати точок A, B, C, D після застосування відповідних перетворень, позначимо їх A', B', C', D' . І нехай, L_1, L_2 — це групи горизонтальних і вертикальних ліній. Тоді, аналогічно перенесенню обмежувальної рамки, що було зроблено на попередньому кроці, можемо оцінити матрицю перспективного перетворення для перенесення фрагменту зображення, обмеженого кутівими точками A', B', C', D' у початок координат. На рис. 3.6. можна побачити результати виділення фрагменту фото, де знаходиться аркуш у клітинку.

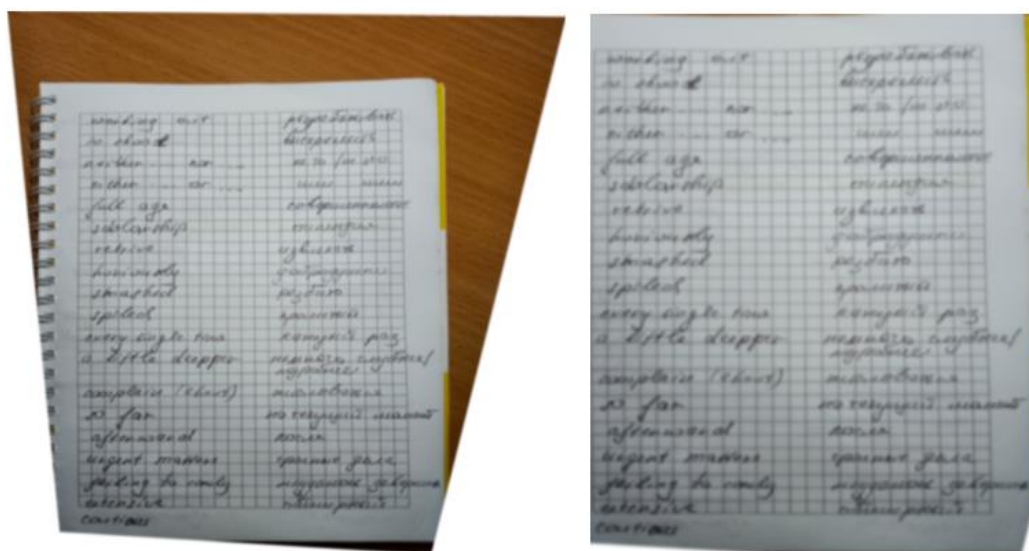


Рис. 3.6. Результат кроку відділення аркуша від фону зображення

Для випадку, коли знайдено L_1, L_2 — групи горизонтальних і косих ліній відповідно, необхідно додати ще один крок. Щоб пояснити його, скористаємося рис. 3.7.

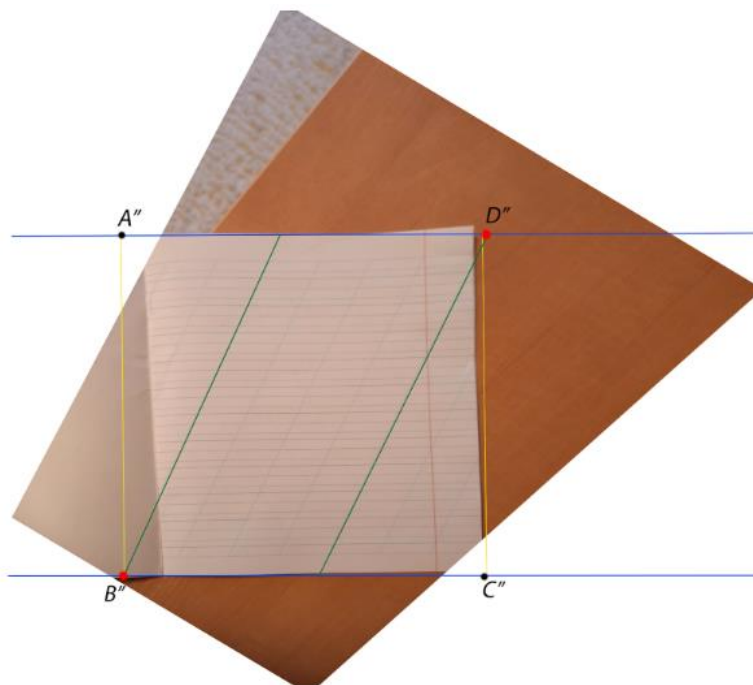


Рис. 3.7. Візуалізація кроку виділення фрагменту зображення у випадку, коли L_1 , L_2 – групи горизонтальних і косих ліній відповідно. Зелені лінії – крайні лінії групи L_2 , сині лінії – крайні лінії групи L_1

Оскільки коса лінія розмітки аркуша перетинається під певним кутом до горизонтальних, її точки перетину з крайніми нижньою і верхньою лініями мають різні координати x . Тоді, серед цих точок перетину знайдемо ті, координати x яких є крайніми у виділеному паралелограмі. На рис 3.7. Ці точки позначені червоним кольором. Їх перші координати у евклідовому просторі можуть бути знайденими за формулами

$$x_{left} = \min_{i \in \{A', B', C', D'\}} \left(\frac{i_z}{i_x} \right), \quad x_{right} = \max_{i \in \{A', B', C', D'\}} \left(\frac{i_z}{i_x} \right).$$

Далі, маючи необхідні крайні ліве і праве положення точок фрагменту, можемо знайти кутові точки у проєктивній площині, по яким даний фрагмент буде виділено і позначимо їх A'' , B'' , C'' , D'' , де

$$\begin{aligned} A'' &= (x_{left} \cdot A'_z, A'_y, A'_z), \\ B'' &= (x_{left} \cdot B'_z, B'_y, B'_z), \\ C'' &= (x_{right} \cdot C'_z, C'_y, C'_z), \\ D'' &= (x_{right} \cdot D'_z, D'_y, D'_z). \end{aligned}$$

Результат даного кроку можна побачити на рис. 3.8.

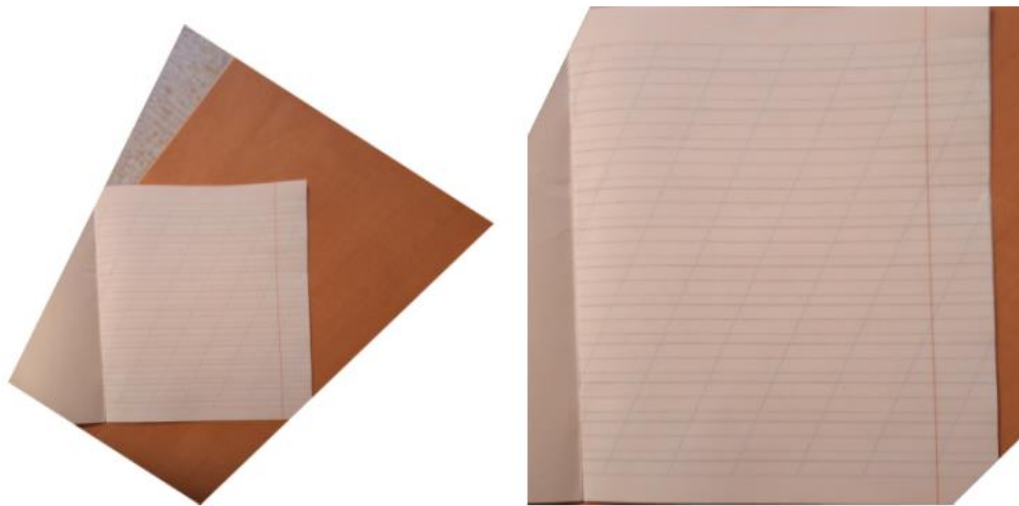


Рис. 3.8. Результат обрізання аркушу на основі горизонтальних та косих ліній

Висновок до розділу 3

У даному розділі описано метод вирівнювання зображень розлінованих аркушів, що мають розмітку у клітинку та косу лінію. Він складається з чотирьох

основних кроків: знаходження ліній на зображенні за допомогою методів Кенні та Хафа, кластеризація ліній на основі методу RANSAC, побудова матриці перспективного перетворення для вирівнювання кутів зображення, параметром якої є реальний між групами ліній у площині, перпендикулярній оптичній осі камери, обрізання зображення по обмежувальній рамці. Також було описано додатковий крок, який дозволяє відділити фрагмент фото, на якому зображено аркуш, від загального фону. За допомогою цього підходу можливо ректифікувати аркуші, розмітка яких не задовольняє основним державним стандартам розліновки, але параметри ліній якої є відомими.

Даний метод реалізований на практиці, проте він ще не є автоматизованим і потребує підбору параметрів кожного кроку алгоритму для кожного нового зображення. Ми побачили, що результати експериментів підтверджують гіпотези про те, що за допомогою ліній на аркуші можна виконувати ректифікацію зображення. Автоматизація та подальше удосконалення цього методу планується у ході наступних досліджень.

РОЗДІЛ 4

ОЦІНКА РЕЗУЛЬТАТІВ ТА ОБМЕЖЕННЯ АЛГОРИТМУ

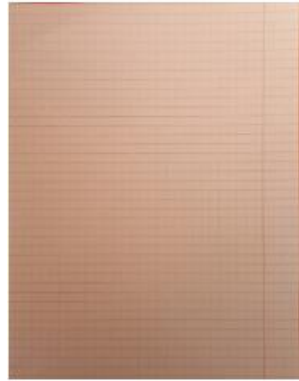
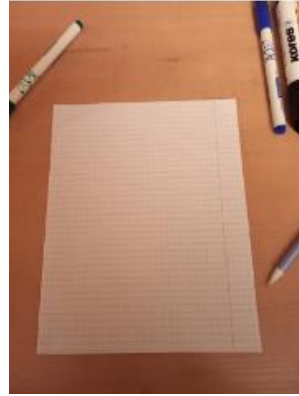

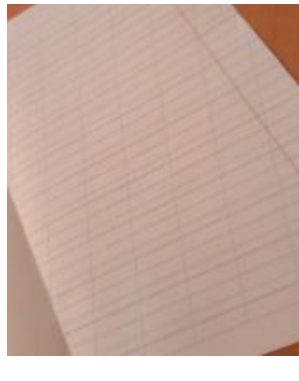





Реалізація запропонованого нами методу дозволяє вирівнювати фотографії розлінованих зошитів в клітинку та косу лінію. Саме ці два види розліновок дають змогу виявити дві необхідні точки сходу, за допомогою яких будується матриця перспективного перетворення, що вирівнює зображення. Наразі можлива якісна ректифікація наступних зображень: зображення містить не цілий документ, документ не контрастує з фоном, основний документ лежить на іншому і повернутий під певним кутом. Наш алгоритм, за наявності розмітки на аркушах, дозволяє вирівнювати такі зображення, що є якісним доповненням вже існуючих методів.

У цьому розділі наведені результати роботи алгоритму, в порівнянні з такими відомими комерційними програмами як Dropbox, PDF Scanner, Adobe Scan, Microsoft Lens. Приклади деяких з них розділені по категоріям відповідно до особливостей зображення аркушів на фото. Також зазначені “слабкі” місця даного методу та плани на його подальше удосконалення.

4.1 Оцінка результатів експериментів

У таблиці 4.1 наведено порівняльні результати роботи методів ректифікації зображень для базового випадку, коли аркуш знаходиться на контрастному фоні. Очікується, що всі програми будуть давати схожі результати.










Таблиця 4.1 Порівняння результатів обробки аркушів на контрастному фоні

Оригінал	Dropbox	PDF Scanner	Наше рішення
			
			
			

З порівняльної таблиці видно, що за найменгошо відхилення від необхідної умови для зображення аркуша: потрапляння інших предметів у кадр, або вихід деяких кутів аркуша за межі фотографії, деякі комерційні програми вже не можуть коректно відшукати документ. Але для нашого методу це не є проблемою.


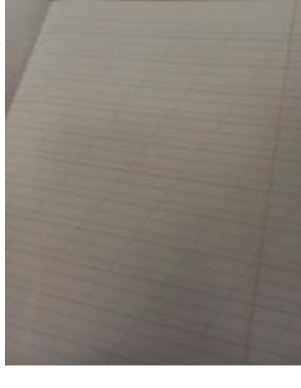


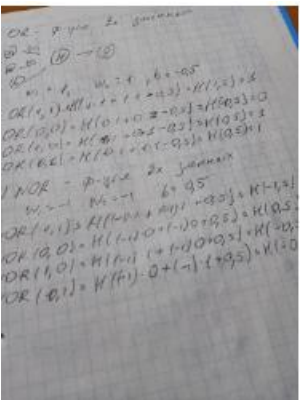

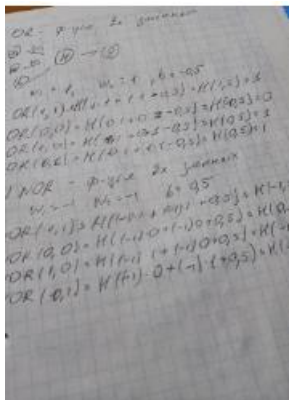
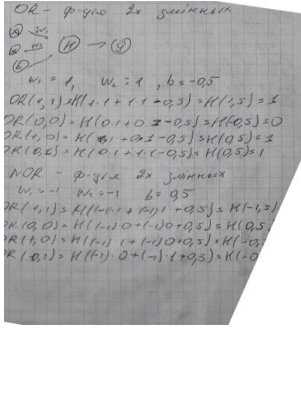
У таблиці 4.2 наведено порівняльні результати роботи методів з ректифікації зображень для випадку, коли на фоні основного аркушу знаходяться інші аркуші, і таким чином візуально викривляється форма документа. Тобто аркуш замасковано. Очікується, що наша програма буде давати кращі результати.

Таблиця 4.2 Порівняння результатів обробки зображень із замаскованим фоном

Оригінал	Dropbox	PDF Scanner	Наше рішення
			
			
			

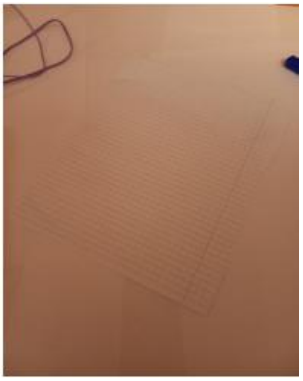







У таблиці 4.3 наведено порівняльні результати роботи методів з ректифікації зображень для випадку, коли на фото потрапляє не весь аркуш, таким чином не проглядається прямокутник як форма документа. Комерційні програми можуть навіть не знаходити аркушів, а наш алгоритм не тільки знайде, але й вирівняє.

Таблиця 4.3 Порівняння результатів обробки зображень, де зображено не повний документ

Оригінал	Dropbox	PDF Scanner	Наше рішення
			
			

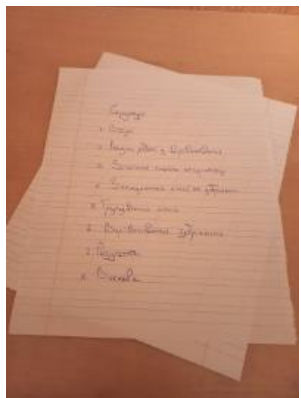

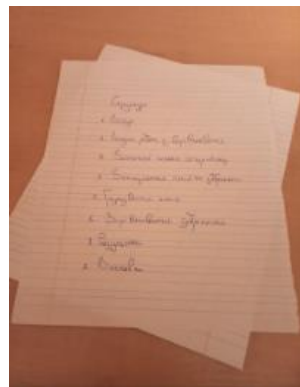





У таблиці 4.4 наведено порівняльні результати роботи методів з ректифікації зображень для випадку, коли аркуш знаходиться на неконтрастному фоні. Очікується, що наш алгоритм буде працювати значно краще.

Таблиця 4.4 Порівняння результатів обробки зображень, де аркуш знаходиться на неконтрастному фоні

Оригінал	Dropbox	PDF Scanner	Наше рішення
			
			

При накладанні одного аркушу на інший можуть виникати ситуації, в котрих розглянуті застосунки використовують кути різних аркушів, які в результаті неправильно окреслюють форму документа. Наш метод уникає таких випадків, це показано у таблиці 4.5.

Таблиця 4.5 Порівняння результатів обробки зображень, де аркуш перекриває інший аркуш

Оригінал	Dropbox	PDF Scanner	Наше рішення
			
			

4.2 Обмеження розробленого алгоритму

У цьому розділі ми висвітлюємо обмеження нашого алгоритму та плани на його подальше удосконалення.

Як видно з попереднього розділу, на поточній стадії, наш метод досить добре опрацьовує велику кількість випадків зображень розлінованих аркушів, і дає на них кращі результати за відомі комерційні програми. Однак, у деяких випадках фотографій він не вірно відтворює правильну геометрію аркушу у фронтально-

паралельній площині. Алгоритм погано справляється з зображеннями, які мають наступні особливості:

а) велике перспективне спотворення: коли кут зйомки дуже малий, зображення не буде коректно перенесено в його фронтально-паралельну площину і стане непридатним до подальшого використання (рис. 4.1а);

б) аркуші, де користувач нарисував паралельні лінії, які не є паралельними хоч одній з істинних груп: під час пошуку ліній алгоритм може захопитися за нарисовані лінії, якщо їх кількість є більшою за знайдену кількість паралельних ліній будь-якої з груп, що призведе до невірному відновленню кута між групами ліній (рис. 4.1б);

в) вигнуті аркуші: вирівнювання працює погано, адже після викривлення ліній, алгоритм Хафа може не знаходити розмітку аркуша (рис. 4.1в);

г) перевернуте зображення: алгоритм не контролює орієнтацію аркуша на зображенні (рис. 4.1г).

Для вирішення проблеми вигнутих аркушів можна застосувати техніку TPS (Thin plate spline) [20] для інтерполяції та згладжування даних. Також у роботі [21] запропоновано метод вирівнювання складених документів, що може бути корисним, якщо аркуш був попередньо зігнутий і після його розгинання залишилися лінії згину, що порушують паралельність прямих. У випадку перевернутих аркушів, можна знаходити початок і кінець рядків тексту і таким чином коректувати орієнтацію аркуша, а у випадку нарисованих ліній намагатися їх розпізнавати споміж ліній розмітки.

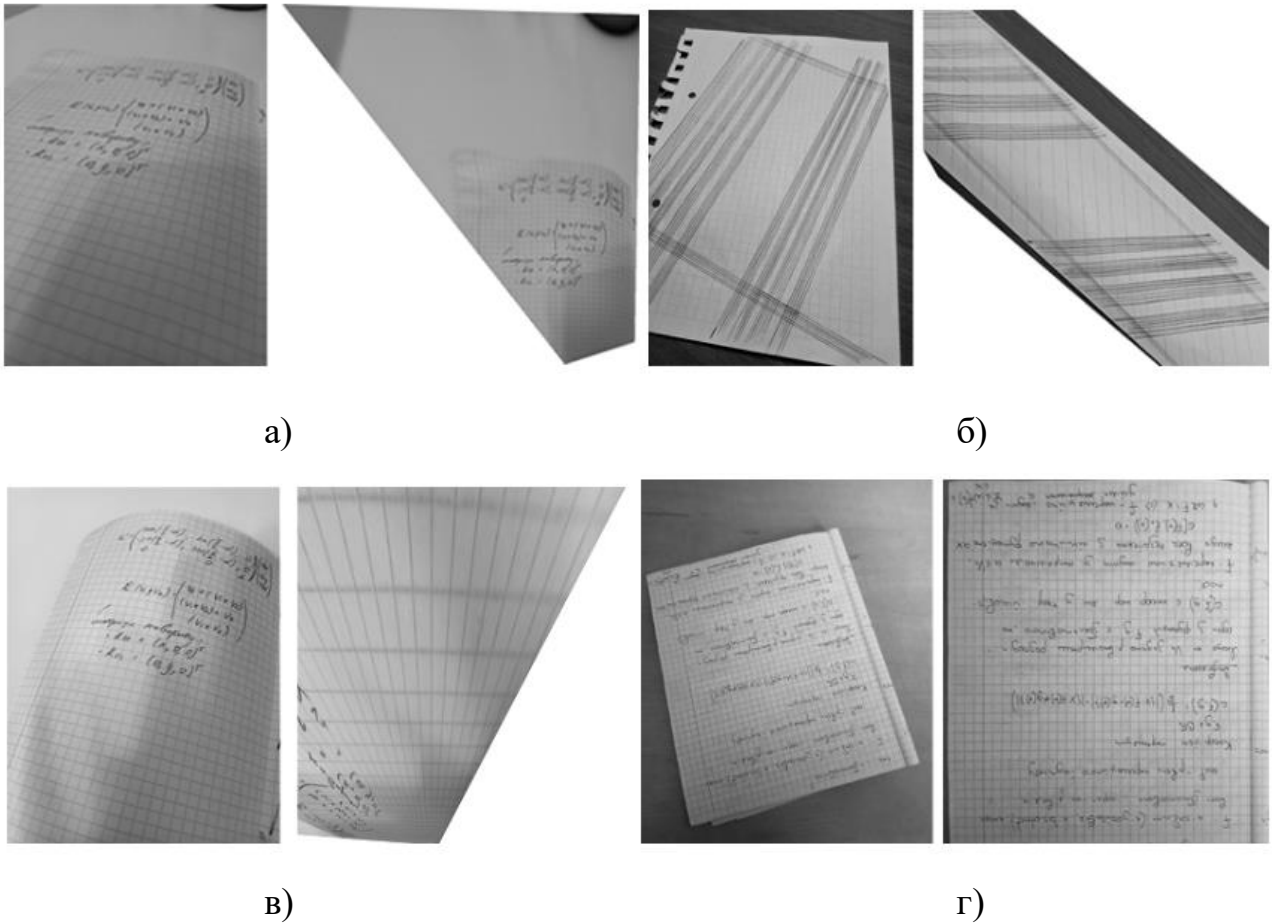


Рис. 4.1 а) – результат роботи алгоритму на зображенні, знятому під малим кутом до площини зображення, б) – результат роботи алгоритму на зображенні з нарисованими паралельними лініями, в) – результат роботи алгоритму на зображенні вигнутого аркуша

Крім цього, удосконалення потребує наведений нами спосіб пошуку паралельних ліній. При обчисленні формули (3.1) виникають ситуації, коли точки перетину ліній l^* з лінією l розташовані майже на одному промені з точкою перетину опорних ліній l та l' , але при цьому далекі від неї. Якщо таких ліній l^* багато, вони можуть бути віднесені до однієї групи паралельних ліній, що призводить до неправильного результату алгоритму (рис. 4.2).

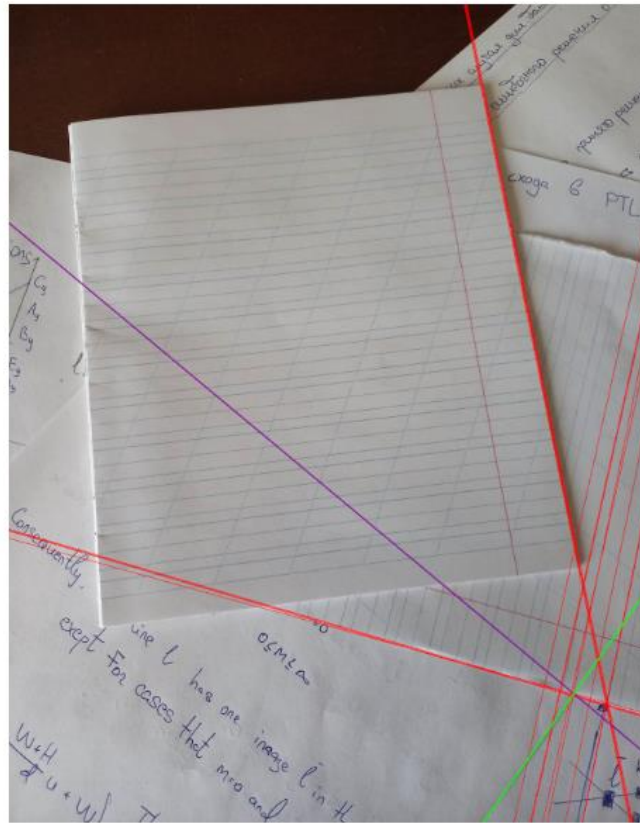


Рис 4.2 – Візуалізація некоретності методу RANSAC. Фіолетова та зелена лінії – це твірні лінії точки сходу даної групи, червоні лінії – всі інші лінії, які входять у дану групу

Ця некоректність може вимагати додаткової фільтрації ліній та/або покращення алгоритму методу RANSAC, що буде забезпечувати перевірку ліній на співнапрямленість перед тим, як виконати крок, що відповідає за знаходження точки сходу.

Висновок до розділу 4

У цьому розділі наведено результати роботи нашого методу вирівнювання зображень аркушів паперу, що мають розмітку у клітинку чи косу лінію у порівнянні з результатами інших відомих комерційних програм. Він показав

кращі результати за такі застосунки як Dropbox, PDF Scanner, Adobe Scan та Microsoft Lens для всіх наступних випадків зображень: зображення містить не цілий документ, документ не контрастує з фоном, основний документ лежить на іншому і повернутий під певним кутом. Проте наш метод має певні обмеження і на окремих випадках зображень дає неправильний результат. Ці випадки включають в себе велике перспективне спотворення, велика кількість паралельних ліній нарисованих користувачем, напрям яких не підпадає під жоден з напрямів ліній розмітки, фото вигнутих та перевернутих аркушів. Також описано проблему, що спонукає до покращення кроку кластеризації ліній.

У подальшому планується удосконалення даного алгоритму, що дозволить автоматично і більш точно знаходити лінії та їх точки сходу, а також доповнення його функціоналу для обробки аркушів в лінію, які потребують визначення параметрів камери або використання стереопари.

ВИСНОВКИ

У результаті роботи вдалося розробити метод вирівнювання зображень аркушів, які мають розмітку в клітинку або у косу лінію. Він розширює множину допустимих для обробки зображень паперових документів та може стати якісним доповненням вже існуючих методів, які орієнтовані на вирівнювання та створення цифрових скан-копій звичайних аркушів, фотографії яких задовольняють певним критеріям, а саме: зображення містить документ цілком, фон зображення констаує з кольором документа, ніякий інший документ не перекриває основний. Наше рішення спрямоване на уникнення залежності від даних критеріїв, тому ми використовуємо розмітку на аркушах, що дозволяє опрацьовувати випадки, які не підпадають під описані вище обмеження. Проектування та реалізація методу були проведені сумісно з Кузнєцовим Павлом Сергійовичем, студентом Фізико-технічного інституту НТУУ “КПІ ім. І. Сікорського”, групи ФІ-12мн.

З аналізу існуючих наукових робіт та комерційних програм, предметом яких є ректифікація зображень, ми зробили висновок, що їх автори керувались припущенням, що основними напрямками паралельних ліній на цих зображеннях є горизонтальний і вертикальний. Тому більшість рішень орієнтована на вирівнювання зображень, на яких можна виділити горизонтальні та вертикальні прямі. Наш метод дозволяє вирівнювати зображення, напрями точок сходу паралельних ліній на яких можуть бути під довільним наперед заданим кутом. Ця особливість і дозволяє нам якісно вирівнювати зображення аркушів у косу лінію.

Застосування нашого алгоритму на різних прикладах зображень показало, що для ряду випадків він краще вирівнює фотографії аркушів, ніж комерційні програми, з якими ми порівнювали його роботу: Dropbox, PDF Scanner, Adobe Scan, Microsoft Lens. Ключовими для нас випадками є коли аркуш не повністю

вміщений на зображенні, тобто його країв не видно, а також коли аркуш, що є об'єктом зйомки, лежить на інших подібних аркушах чи на неконтрастному фоні.

Наразі планується удосконалення існуючого алгоритму, що дозволить автоматично і більш точно знаходити лінії та їх точки сходу, а також доповнення його функціональності методами відтворення правильного формату аркуша та обробки аркушів в лінію, які потребують визначення параметрів камери або використання стереопари.

ПЕРЕЛІК ПОСИЛАНЬ

1. Turn your phone into a document scanner [Electronic resources]: Dropbox official documentation. — Access mode: <https://www.dropbox.com/features/productivity/document-scanner>
2. PDF Scanner - Document Scanner [Electronic resources]. — Access mode: <https://play.google.com/store/apps/details?id=com.camscanner.documentscanner.pdfscanner.textscanner.photos.scanner>
3. Scan documents with Google Drive [Electronic resources]. — Access mode: <https://support.google.com/drive/answer/3145835?hl=en&co=GENIE.Platform%3DAndroid>
4. Adobe Scan for Android [Electronic resources]. — Access mode: <https://www.adobe.com/devnet-docs/adobescan/android/en/scan.html>
5. D. Liebowitz, Metric Rectification for Perspective Images of Planes / D. Liebowitz, A. Zisserman. IEEE Transactions on Computers, 2014
6. K. Chaudhury, Auto-Rectification of User Photos / K. Chaudhury, S. DiVerdi, S. Ioffe – IEEE Conference Publication, 2015.
7. M. A. Fischler, Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography / M. A. Fischler, R. C. Bolles. Communications of the ACM, 1981 pp. 381–395
8. M. Fawzi, Rectification of Camera Captured Document Images for Camera-Based OCR Technology / M. Fawzi, M. Rashwan, H. Ahmed. International Conference on Document Analysis and Recognition (ICDAR), 2015
9. John Canny. A computational approach to edge detection. Pattern Analysis and Machine Intelligence, IEEE Transactions, (6):679–698, 1986.
10. P.V.C. Hough, Method and means for recognizing complex patterns, U.S. Patent 3,069,654, Dec. 18, 1962

11. J. Baek, Fast Document Rectification and Enhancement – Dropbox.tech Publication, 2016.
12. Ying Xiong, Fast and Accurate Document Detection for Scanning – Dropbox.tech Publication, 2016.
13. R. Hartley, Multiple View Geometry in Computer Vision Second Edition / R. Hartley, A. Zisserman – Cambridge University Press, Cambridge, United Kingdom, 2004 – pp. 25-44
14. R. Duda, Use of the Hough transformation to detect lines and curves in pictures / R. Duda, P. Hart, Communications of the ACM, 1972
15. How to compare points in homogeneous coordinates universally? Access mode: <https://math.stackexchange.com/a/3905967/303340> – 04.12.2020
16. GetPerspectiveTransform, OpenCV documentation [Electronic resources]. — Access mode: https://docs.opencv.org/4.x/da/d54/group__imgproc__transform.html#ga20f62aa3235d869c9956436c870893ae
17. R. Ebker, Random Sample Consensus Explained. Access mode: <https://www.baeldung.com/cs/ransac>
18. AdaptiveThreshold, OpenCV documentation [Electronic resources]. — Access mode: https://docs.opencv.org/4.x/d7/d4d/tutorial_py_thresholding.html
19. Application of the projection matrix: Removing perspective distortion [Electronic resources]. — Access mode: <https://mathematica.stackexchange.com/questions/121822/application-of-the-projection-matrix-removing-perspective-distortion>, 2016
20. Bookstein, F. L. Principal Warps: Thin Plate Splines and the Decomposition of Deformations. IEEE Trans. Pattern Anal. Mach. Intell. 11, 1989, pp. 567-585
21. S. You, Multiview Rectification of Folded Documents / S. You, Y. Matsushita, S. Sinha, Y. Bou, IEEE Transactions on Pattern Analysis and Machine Intelligence, 2018