

Вінницький національний технічний університет
Факультет інформаційних технологій та комп'ютерної інженерії
Кафедра комп'ютерних наук

Пояснювальна записка

до магістерської кваліфікаційної роботи

на тему «Інформаційна технологія підрахунку голосів на виборах»

Виконав: студент 2 курсу,
групи 2КН-19м
спеціальності 122 «Комп'ютерні науки»
Спажев О. М.

Керівник: к.т.н., ст. викладач Озеранський В. С.

Рецензент: д.т.н., професор Романюк О. Н.

Вінниця
2020

ЗАТВЕРДЖУЮ
Завідувач кафедри КН
д.т.н., проф. Яровий А.А.
(наук. ст., вч. зв., ініц. та прізви.)

_____ (підпис)
“ ____ ” _____ 2020 року

ЗАВДАННЯ

на магістерську кваліфікаційну роботу на здобуття кваліфікації магістра наук зі спеціальності: 122 «Комп'ютерні науки»
(шифр – назва спеціальності)

08-22.МКР.023.19.000.ПЗ

Магістранта групи 2КН-19м Спажева Олександра Миколайовича
(назва групи) (прізвище, ім'я і по батькові)

Тема магістерської кваліфікаційної роботи: «Інформаційна технологія підрахунку голосів на виборах»

Вхідні дані: набір сканованих зображень бюлетенів; обробка зображень формату png, jpeg; обробка зображень з кутом повороту від 0 до 360 градусів; час на обробку одного зображення – не більше 1 секунди; мова програмування та середовище розробки, що забезпечує обробку великого об'єму графічних даних.

Короткий зміст частин магістерської кваліфікаційної роботи

1. Графічна: Спрощена діаграма прецедентів виборчого процесу, діаграма прецедентів виборчого процесу з OCR, алгоритм обробки бюлетенів, діаграма діяльності обробки бюлетенів

2. Текстова (пояснювальна записка): вступ, аналіз предметної області Підрахунку голосів на виборах, електронне голосування та його ризику, обґрунтування вибору види процедур електронного голосування, обґрунтування вибору типу системи фізичного голосування з підрахунком за допомогою електронних засобів, аналіз існуючих аналогів систем голосування з оптичним скануванням, розробка інформаційної технології підрахунку голосів на виборах, розробка підходу для впровадження автоматизованого підрахунку голосів за допомогою системи оптичного розпізнавання, розробка алгоритму обробки бюлетенів, обґрунтування вибору технології отримання даних із зображення, визначення підходу до попередньої обробки зображення бюлетеня, програмна реалізація інформаційної технології підрахунку голосів на виборах, обґрунтування вибору мови програмування, обґрунтування вибору середовища розробки, обґрунтування вибору модуля OCR, опис складових програмної реалізації, тестування, економічна частина, висновки, перелік використаних джерел.

КАЛЕНДАРНИЙ ПЛАН ВИКОНАННЯ МКР

№ етапу	Назва етапу	Термін виконання		Очікувані результати
		початок	кінець	
1	Аналіз сучасного рівня розвитку інформаційних технологій підрахунку голосів на виборах. Постановка задач дослідження			Аналітичний огляд літературних джерел, задачі досліджень, розділ 1 ПЗ
2	Розробка методу та інформаційної технології підрахунку голосів на виборах			Метод, інформаційна технологія, розділ 2
3	Програмна реалізація розробленої інформаційної технології, тестування та оцінка параметрів			Програмне забезпечення, розділ 3
4	Підготовка економічної частини			розділ 4
5	Апробація та/або впровадження результатів дослідження			тези доповідей
6	Оформлення пояснювальної записки, графічного матеріалу та презентації			Пояснювальна записка, графічний матеріал, презентація

Консультанти з окремих розділів магістерської кваліфікаційної роботи

1. Науковий керівник _____ канд. техн. наук, ст. викладач кафедри КН
(підпис) наук. ступінь, вчене звання (посада)
В. С. Озеранський
ініціали та прізвище

“ _____ ” _____ 20__ р.

2. Економічна частина _____ канд. екон. наук, доц. кафедри ЕПВМ
(підпис) наук. ступінь, вчене звання (посада)
М. В. Бальзан
ініціали та прізвище

“ _____ ” _____ 20__ р.

Дата попереднього захисту роботи “ _____ ” _____ 20__ р.

Рецензент _____ докт. техн. наук, зав. каф. ПЗ, професор
(підпис) наук. ступінь, вчене звання (посада)
О. Н. Романюк
ініціали та прізвище

Завдання видав
 науковий керівник _____ канд. техн. наук, ст. викладач кафедри КН
(підпис) наук. ступінь, вчене звання (посада)
В. С. Озеранський
ініціали та прізвище

“ _____ ” _____ 20__ р.

Завдання отримав магістрант _____ О. М. Спажев
(підпис) ініціали та прізвище

“ _____ ” _____ 20__ р.

АНОТАЦІЯ

Дана магістерська кваліфікаційна робота присвячена створенню інформаційної технології підрахунку голосів на виборах. Під час виконання дослідження було запропоновано використати алгоритм попередньої обробки, що значно покращило можливості розпізнавання зображення в різних умовах. Результати тестування підтвердили ефективність обраного методу на практиці. Результат роботи може бути використаний будь-якою державою або установою, яка використовує паперове голосування для волевиявлення з того чи іншого питання.

ABSTRACT

This master's qualification work is devoted to the creation of information technology of counting votes in elections. The author propose to use a pre-processing algorithm that significantly improved the accuracy of image recognition in different conditions. Further confirmed the effectiveness of the chosen method in practice. The result of the work can be used by any state or institution that uses paper voting to express a will on a particular issue.

ЗМІСТ

ВСТУП	8
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ПІДРАХУНКУ ГОЛОСІВ НА ВИБОРАХ	12
1.1 Електронне голосування та його ризики	12
1.2 Обґрунтування вибору виду процедур електронного голосування.....	13
1.3 Обґрунтування вибору типу системи фізичного голосування з підрахунком за допомогою електронних засобів.	15
1.4 Аналіз існуючих аналогів систем голосування з оптичним скануванням	16
1.5 Висновок	18
2 РОЗРОБКА ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ПІДРАХУНКУ ГОЛОСІВ НА ВИБОРАХ	19
2.1 Розробка підходу для впровадження автоматизованого підрахунку голосів за допомогою системи оптичного розпізнавання.....	19
2.1.1 Аналіз виборчого процесу для впровадження системи оптичного розпізнавання бюлетенів	20
2.1.2 Обґрунтування необхідності запропонованих змін до процесу підрахунку голосів	20
2.1.3 Розробка схеми виборчого процесу із запропонованими змінами .	21
2.2 Розробка алгоритму обробки бюлетенів.....	22
2.3 Обґрунтування вибору технології отримання даних із зображення.....	24
2.4 Визначення підходу до попередньої обробки зображення бюлетеня ...	25
2.4.1 Алгоритм BRISK	27
2.4.2 Алгоритм A-KAZE	30
2.4.3 Алгоритм ORB.....	32

2.4.4 Обґрунтування вибору алгоритму пошуку «особливих» точок.....	34
2.5 Структура інформаційної технології підрахунку голосів на виборах...	37
2.6 Висновок	38
3 ПРОГРАМНА РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ПІДРАХУНКУ ГОЛОСІВ НА ВИБОРАХ.....	38
3.1 Обґрунтування вибору мови програмування	39
3.2 Обґрунтування вибору середовища розробки.....	41
3.3 Обґрунтування вибору модуля OCR.....	42
3.4 Опис складових програмної реалізації	46
3.5 Тестування інформаційної системи підрахунку голосів на виборах та аналіз результатів	48
3.6 Висновок	55
4 ЕКОНОМІЧНА ЧАСТИНА	56
4.1 Оцінювання комерційного потенціалу розробки.....	56
4.2 Прогнозування витрат на виконання науково-дослідної роботи та конструкторсько–технологічної роботи	57
4.3 Прогнозування комерційних ефектів від реалізації результатів розробки	61
4.4 Розрахунок ефективності вкладених інвестицій та період їх окупності	63
4.5 Висновок	66
ВИСНОВКИ.....	68
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	69
Додаток А Інструкція користувача	73
Додаток Б Лістинг програми.....	76
Додаток В Графічна частина.....	78

ВСТУП

Актуальність теми дослідження. Основною формою правління в наш час є – демократія. Застосовується вона як серед невеличкої групи людей, так і в багатомільйонних державах. Основним інструментом прийняття рішень для даної форми є – голосування або ж волевиявлення. Складність використання цього інструмента збільшується залежно від кількості учасників, які можуть брати участь. Хоча демократія існує з часів Древньої Греції, проведення чесних виборів (які також є голосуванням) досі є складною задачею, а неправдиві результати приводять до протестів, революцій або, навіть, до громадянської війни та військових конфліктів.

Вибори, як і процедура підрахунку голосів, дуже сильно відрізняється в різних країнах та визначається законодавством.

Деякі країни, такі як: США, Канада, Індія, Бразилія, Бельгія, Естонія, Австралія та Південна Корея вже запровадили у себе електронне голосування. А Великобританія, Німеччина, Франція, Іспанія, Португалія, Норвегія, Швейцарія, Росія, Казахстан, Японія та Китай проводять експерименти з його використання [1].

Україна відсутня в списках країн, які впроваджують або знаходяться на шляху впровадження даних нововведень, але наша країна ратифікувала Статут Ради Європи відповідно до якого взяла на себе зобов'язання прислуховуватися до порад Венеціанської комісії [2, 3]. Єдине що було зроблено – прийнято ряд законів: "Про інформацію", "Про основні засади розвитку інформаційного суспільства в Україні на 2007 - 2015 роки", "Про Концепцію національної програми інформатизації", "Про Національну програму інформатизації" та найголовніший – «Про Концепцію "Запровадження системи електронного голосування в Україні"», який так і не був розглянутий, а лише надійшов до Верховної Ради України.

Але все це не призвело до жодних практичних результатів, які б стосувалися рекомендацій наданих Венеціанською Комісією у ряді доповідей Європейської комісії за Демократію через Право, присвячених складнощам та проблемам відповідності віддаленого голосування (одним з видів якого є електронне голосування).

Враховуючи все вищесказане, дана тема є дійсно актуальною з точки зору підвищення довіри до результатів голосування та реалізації цілей державної політики. Основним ж інструментом, що задовольняє обидві цілі є використання штучного інтелекту, а саме певної інформаційної технології, результати роботи якої будуть точнішими, прозорішими та такими, що зменшують можливість маніпуляцій.

Зв'язок роботи з науковими програмами, планами, темами. Магістерська робота виконана згідно з напрямком наукових досліджень кафедри комп'ютерних наук Вінницького національного технічного університету 22 К1 «Моделі, методи, технології та пристрої інтелектуальних інформаційних систем управління, економіки, навчання та комунікацій» та плану наукової та навчально-методичної роботи кафедри. Також, вона реалізує статтю 3 «Державна інформаційна політика» Закону України «Про інформацію» від 02.10.1992 № 2657-ХІІ, де вказано, що одним з основних напрямів державної інформаційної політики є «створення інформаційних систем і мереж інформації, розвиток електронного урядування» [4].

Мета та завдання дослідження. Метою дослідження магістерської кваліфікаційної роботи є розширення функціональних можливостей системи підрахунку голосів на виборах шляхом усунення основних недоліків аналогів, зменшення можливості маніпуляцій під час підрахунку голосів на виборах за допомогою методів та програмних засобів для автоматизованого підрахунку голосів на виборах.

Для досягнення поставленої мети необхідно розв'язати наступні завдання:

- провести аналіз предметної області електронного голосування та підрахунку голосів на виборах;

- розглянути існуючі методи, які можуть бути використані для автоматизованого підрахунку та обґрунтувати вибір того методу, який дозволяє досягти мети даної магістерської кваліфікаційної роботи;

- розробити структуру та алгоритм роботи інформаційної технології підрахунку голосів на виборах;

- здійснити програмну реалізацію запропонованої інформаційної технології;

- провести тестування розробленого програмного забезпечення та виконати аналіз отриманих результатів.

Об'єкт дослідження – це процес підрахунку голосів на виборах.

Предмет дослідження – програмні засоби для автоматизації підрахунку голосів на виборах.

Методи дослідження. У роботі використані такі методи наукових досліджень: проектування інформаційних систем з ціллю створення структури інформаційної технології, теорія нейронних мереж для реалізації інформаційної технології підрахунку голосів на виборах, об'єктно-орієнтоване програмування для реалізації вищевказаної інформаційної технології.

Наукова новизна одержаних результатів полягає в наступному:

- Удосконалено інформаційну технологію підрахунку голосів на виборах, що відрізняється від відомих аналогів попередньою обробкою зображень бюлетенів, яка збільшує можливості коректного визначення результату при нетипових ситуаціях;

Практичне значення одержаних результатів полягає у наступному:

- Розроблено алгоритм попередньої обробки зображень бюлетеня різної якості, який дозволяє розширити функціонал системи підрахунку голосів, що в свою чергу збільшує довіру до результатів голосування на виборах.

Достовірність теоретичних положень магістерської кваліфікаційної роботи підтверджується строгістю постановки задач, строгим виведенням аналітичних співвідношень, правильним застосуванням математичних методів під час доведення наукових положень, порівняння результатів роботи з відомими та збіжністю результатів математичного моделювання з отриманими результатами під час тестування програмного засобу.

Особистий внесок магістранта. Усі результати наведені у магістерській кваліфікаційній роботі отримані самостійно.

Апробація результатів роботи. Результати роботи були апробовані на XLIX Науково-технічній конференції факультету інформаційних технологій та комп'ютерної інженерії (2020).

Публікації. За результатами магістерської кваліфікаційної роботи опубліковано 1 тезу доповідей конференцій [5].

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ПІДРАХУНКУ ГОЛОСІВ НА ВИБОРАХ

Найнадійнішим способом підрахувати голоси на виборах – довірити це тим, хто є максимально незалежним. Звісно, нічого повністю незалежного в наш час не існує. Все залежить щонайменше від живлення. Але загалом, комп'ютери, при певних умовах, здається, підходять краще ніж люди. Особливо що стосується швидкості обчислень та їх точності. Звісно, при правильній організації процесу. Однак, ступінь залучення технологій може бути різним.

1.1 Електронне голосування та його ризики

Практично будь-яка електронна система побудована так, що дозволяє перевіряти правильність функціонування за допомогою спостереження за даними, які надходять на всіх програми.

Системи е-голосування принципово відрізняються. Основним чином, за рахунок вимоги зберігати таємницю голосування. А саме: забороняє відстеження зв'язку між виборцем і тим як він проголосував, що в своє чергу унеможлиблює стандартний спосіб перевірки транзакцій для встановлення достовірності, на якому побудовані електронні системи.

В зв'язку з цим надзвичайно важливими стають такі непрямі докази правильності роботи та визначення результатів як паперові сліди або сертифікація системи в об'єднанні з суворими заходами безпеки та контролем якості. При їх відсутності, способи маніпуляції або неправильні результати, отримані внаслідок роботи системи електронного голосування залишитися непоміченими протягом тривалого часу, що призведе до визнання результатів недійсними і понесе за собою, в найкращому випадку, значні фінансові витрати [6].

1.2 Обґрунтування вибору виду процедур електронного голосування

Зазвичай виділяють три види процедур електронного голосування. А саме:

1. Віддалене голосування за допомогою Passport ID та інтернету.

Процес голосування: виборець приєднує свою ID картку до кардрідера, який показано на рисунку 1.1, переходить на сайт ЦВК, здійснює вхід за допомогою пін-коду та переходить на сторінку електронного бюлетеня, голосує та виконує підтвердження свого вибору повторивши введення пін-коду. Голос виборця опрацьовується майже моментально, підраховується та оприлюднюється онлайн.

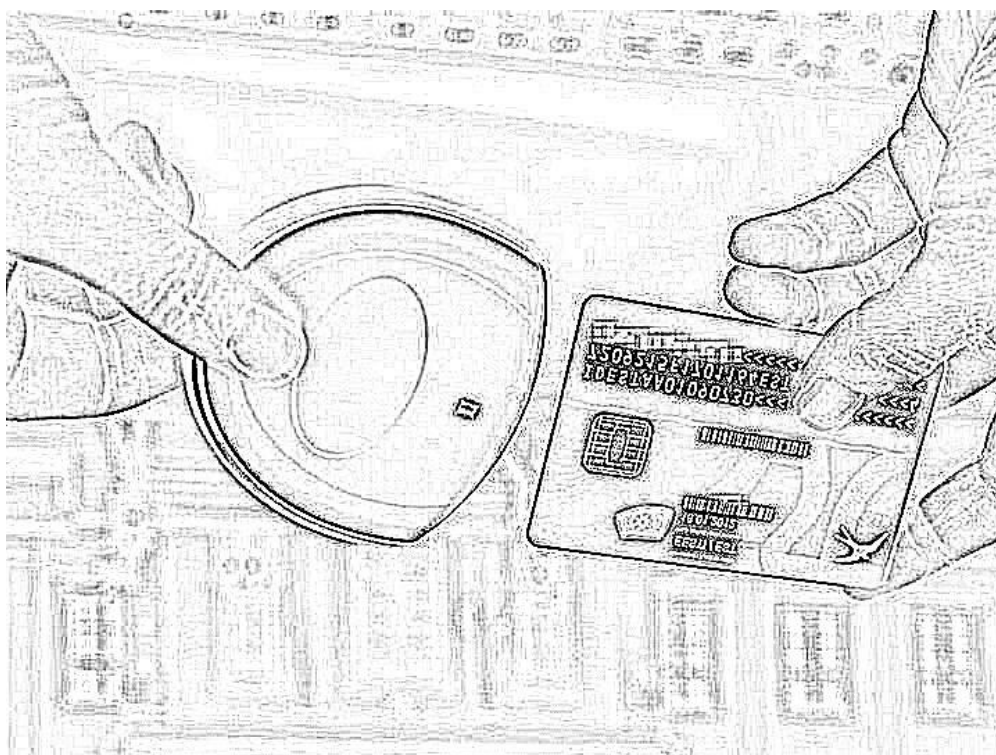


Рисунок 1.1 – ID картка та кардрідер для ідентифікації особи при голосуванні

2. Фізичне голосування на дільницях з підрахунком за допомогою електронних засобів.

Процес голосування схожий на традиційний, але підрахунок проводиться не людьми, а автоматизованою системою, яка після зчитування бюлетеня самостійно визначить результат та збереже результати для подальшої обробки або одразу зрахує та відправить на відображення на офіційному сайті.

3. Голосування за допомогою спеціальних цифрових терміналів.

Спеціальні цифрові термінали – це електронні обчислювальні машини або комп'ютери з сенсорним екранами на зразок тих, де можна проводити електронні операції поповнення рахунку, оплати послуг та інші. Такі термінали можуть бути багатофункціональні з можливістю авторизації через засіб ідентифікації особи (Passport ID-картку або звичайного паспорту з використанням спеціального номеру для голосування від членів виборчої комісії) [7].

Аналізуючи наведені вище види процедур електронного голосування та врахувавши вплив наступних факторів:

- складність реалізації та забезпечення достовірності голосування через інтернет;
- неможливість забезпечення усіх жителів України інтернетом нормальної якості або в цілому (мала площа покриття території країни мобільним інтернетом, обмеженість у доступі до інтернету у селах та невеликих містечках);
- неможливість забезпечення безпеки, дотримання прав і свобод кожного учасника голосування при віддаленому голосуванні;
- необхідність додаткових витрати для навчання населення користуванню терміналами або голосуванню через інтернет;

очевидним стає те, що єдиним прийнятним варіантом на момент написання даної роботи є 2-гий варіант, а саме фізичне голосування на дільницях з підрахунком за допомогою електронних засобів.

1.3 Обґрунтування вибору типу системи фізичного голосування з підрахунком за допомогою електронних засобів.

Фізичне голосування з використанням електронних засобів може здійснюватися різними способами. Нижче наведено основні з них:

1. Електронні апарати для голосування з прямим записом.

Такі апарати можуть поставлятися разом з паперовим слідом або без нього (контрольний паперовий слід, що перевіряється виборцем). Паперові сліди призначені для створення фізичних доказів поданих голосів.

2. Системи оптичного розпізнавання міток.

Такі системи використовуються сканери, що можуть визначити вибір громадянина на спеціальних виборчих бюлетенях, що автоматично зчитуються спеціальними засобами. Системи оптичного розпізнавання міток можуть бути централізованими системами підрахунку (якщо виборчі бюлетені збирають, а потім сканують і підраховують у спеціальних центрах підрахунку голосів) або дільничними (розподіленими) системами оптичного сканування й підрахунку (коли підрахунок та сканування відбуваються на виборчій дільниці, одразу коли виборці вкидають свій виборчий бюлетень до автомату для голосування).

3. Електронні принтери бюлетенів – пристрої, подібні до електронних апаратів для голосування з прямим записом, які друкують папірці для зчитування автоматизованими записами або електронні жетони, що містять інформацію про вибір громадянина. Жетони кидають до окремого сканеру виборчих бюлетенів, де піддаються автоматичному зчитуванню та підрахунку голосів [8].

Порівнюючи дані методи, перевагу має другий метод, оскільки перший як і третій варіант має декілька суттєвих ризиків, а саме:

- a. Небезпека співставлення особи яка проголосувала з її вибором;
- b. Залежність роботи системи від інтернет з'єднання;
- c. Необхідно додаткове фінансування для навчання людей користуватися даним апаратом;

а також необхідність значної кількості додаткового фінансування для створення таких апаратів а також забезпечення стабільного інтернет з'єднання з потенційно дуже великими витратами.

Враховуючи все вищевказане, варіант з фізичним голосуванням та використанням систем оптичного розпізнавання міток має наступні переваги (варто зазначити, що мова йде лише про централізовану обробку, оскільки децентралізована призводить до тих самих складнощів, що й інші розглянуті альтернативи):

- простота впровадження – кількість людей, яких необхідно навчити для користування даною системою значно менше порівняно з іншими альтернативами;
- менша вартість впровадження – необхідно лише розробити систему для обробки та кілька автоматів для обробки на яких буде ця система через використання саме централізованого опрацювання.

1.4 Аналіз існуючих аналогів систем голосування з оптичним скануванням

Провівши інформаційний пошук в системі Google за запитом «Системи голосування з оптичним скануванням» було отримано результат зображений на рисунку 1.2.

В результаті було отримано 23 тисячі 300 результатів.

Провівши глибший аналіз перших 10 сторінок отриманих результатів (найвідповідніші запити відображаються першими), було встановлено, що

лише одне з наданих посилань, а саме «Система распознавания с оптическим сканированием» [9] містить в собі інформацію про аналоги.

Стаття містить згадки про виробників, таких як:

- Premier Election Solutions;
- Избирательные системы и программное обеспечение (ES&S);
- Smartmatic;

та інші, але не містить детального опису таких систем або посилань на них.

The image shows a Google search results page. The search bar contains the text "системи голосування з оптичним скануванням". Below the search bar, there are navigation options: "Усі", "Зображення", "Новини", "Відео", "Карти", "Більше", "Налаштування", and "Інструменти". The search results are as follows:

- Approximate number of results: 23 300 (0,70 сек.)
- uk.wikipedia.org › wiki › Електронне_голосування
 - Електронне голосування — Вікіпедія**
 - Під електронними технологіями **голосування** можуть розумітися: перфокарти та **системи оптичного сканування** бюлетенів, спеціальні «кіоски ...
- rada.oporaua.org › analityka › a-iak-u-nykh › 23280-a...
 - А як у них? Електронне голосування — Рада - все про ...**
 - 8 серп. 2018 р. - **Оптичне сканування** (Optical Scan Marksense) – виборець обирає ... Усі автоматизовані **системи** електронного **голосування** мають бути ...
- euinfocenter.rada.gov.ua › uploads › documents
 - Системи електронних виборів - Європейський ...**
 - система** електронного **голосування** повинна бути прозорою, тобто надавати ... **Оптичне сканування** (Optical Scan Marksense) передбачає здійснення.
 - Ви вже відвідували цю сторінку стільки разів: 2 Останнє відвідування: 17.09.20
- ru.qwe.wiki › wiki › Optical_s...
 - Система голосования с оптическим сканированием ...**
 - Система** голосования с оптическим сканированием - Optical scan voting system. Из Википедии, свободной энциклопедии. Технология выборов.
 - Ви вже відвідували цю сторінку стільки разів: 3 Останнє відвідування: 23.09.20
- ips.ligazakon.net › document › view
 - Про внесення змін до деяких законодавчих актів України ...**
 - ... засобів, зокрема **систем оптичного сканування** бюлетенів, електронних ... На виборчому бюлетені для **голосування** в територіальному виборчому ...

Рисунок 1.2 – Результати інформаційного пошуку за запитом «Системи голосування з оптичним сканування»

Також вказано, що більшість з цих систем мали значні недоліки, одні з яких були:

- подача бюлетеня «вверх ногами»;
- одночасне сканування декількох бюлетенів при центральному підрахунку;
- зім'яття паперу;
- помилки в самих системах;

та інші, більшість з яких була виявлена вже після підрахунку голосів (в деяких випадках навіть через кілька років).

1.5 Висновок

В даному розділі було проведено аналіз предметної області електронного голосування.

В результаті, враховуючи поставлені вимоги, а саме:

- простоту інтеграції в діючу систему;
- мінімізація витрат;
- максимізація універсальності системи;

було обрано реалізацію системи при фізичному голосування на дільницях, але з централізованим підрахунком за допомогою оптичної системи сканування.

Також було здійснено спробу проаналізувати існуючі аналоги систем голосування з оптичним скануванням. В результаті можна визначити додаткові вимоги або уточнення до них для врахування недоліків аналогів.

- працювати не лише зі скан-копіями, але й з фотографіями;
- обробляти не лише строго вертикальний текст, але й з певними відхиленнями;
- перевіряти відповідність кількості порохованих бюлетенів до кількості контрольних талонів.

2 РОЗРОБКА ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ПІДРАХУНКУ ГОЛОСІВ НА ВИБОРАХ

Процес електронного голосування або виборів складається з великої кількості етапів та подій, що відбуваються всередині кожного з них.

Базуючись на аналізі, результати якого наведено в першому розділі, основним елементом технології в даній роботі буде автоматизований підрахунок голосів за допомогою системи оптичного розпізнавання.

2.1 Розробка підходу для впровадження автоматизованого підрахунку голосів за допомогою системи оптичного розпізнавання

Для кращого розуміння можливостей розробки інформаційної технології підрахунку голосів на виборах слід навести процеси існуючої виборчої системи [10].

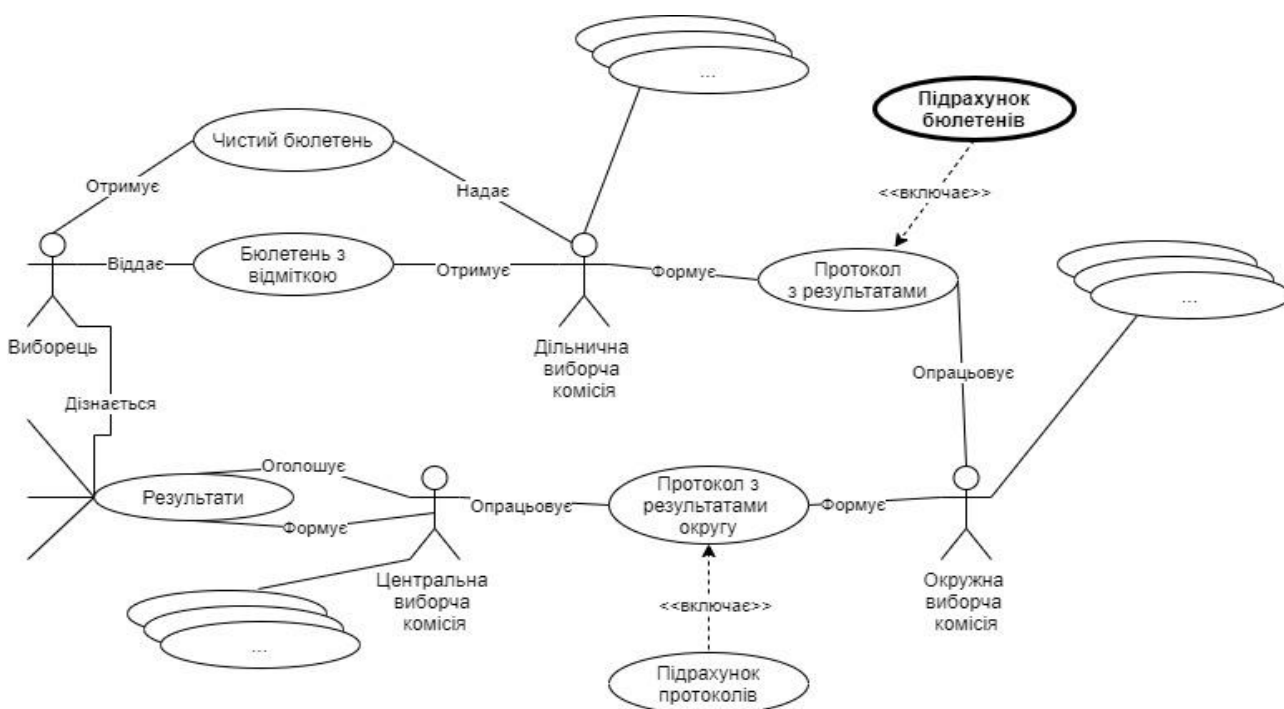


Рисунок 2.1 – Спрощена діаграма прецедентів виборчого процесу

Для цього краще використати UML діаграми [11], а саме спрощену діаграму [12] прецедентів виборчого процесу, зображену на рисунку 2.1, та проаналізувати її стосовно процесу підрахунку голосів.

2.1.1 Аналіз виборчого процесу для впровадження системи оптичного розпізнавання бюлетенів

Аналізуючи наведену на рисунку 2.1 діаграму прецедентів виборчого процесу стосовно підрахунку голосів, можна побачити, що фактичний підрахунок голосів відбувається на дільницях (виділено жирним) «випадковими» людьми, які втрачаються свої повноваження через 2 дні після виборів, а їх найми відбувається за кілька тижнів до початку виборів. А результати формуються Центральною виборчою комісією (ЦВК). Однак, ЦВК працює навіть не з вторинними, а з третинними результатами, оскільки після формування дільничними виборчими комісіями протоколів, на їх основі протоколи формують окружні виборчі комісії, які вже опрацьовуються ЦВК.

Звісно, делегація повноважень спрощує та пришвидшує процес підрахунку голосів, але це також збільшує ризики махінацій та зменшує довіру до результатів.

2.1.2 Обґрунтування необхідності запропонованих змін до процесу підрахунку голосів

Для збільшення рівня довіри до результатів за зменшення ризику махінацій слід ввести контрольну перевірку результатів, а саме автоматизований підрахунок голосів.

Однак, потрібно не просто її запровадити, а зробити це оптимально.

Відповідно до рисунку 2.1, перевірку можна впровадити на етапі:

- Голосування.

Хоч це і буде найшвидшим способом обробки, але це потребує

значних витрат на створення необхідної кількості систем для підрахунку (замінити всі «урни» для бюлетенів);

- В дільничних або окружних виборчих комісіях. Виникає та ж сама проблема, що і під час голосування, однак затрати вже менші. Окрім того знову ж таки потрібно делегувати відповідальність за цілісність системи на сторонніх осіб, що не вирішує поставлену задачу;
- В центральній виборчій комісії. Потребує також значну кількість автоматів для підрахунку за прийнятний час, але делегування відповідальності не відбувається, оскільки відповідальна людина може бути членом ЦВК, яке призначається на 5 років. Окрім цього необхідно створити певний підрозділ, який буде слідкувати за коректною роботою автоматів та гарантувати їх цілісність. Також необхідно зробити можливість допуску спостерігачів на процес підрахунку.

2.1.3 Розробка схеми виборчого процесу із запропонованими змінами

Врахувавши дослідження проведені в розділі 1 та необхідні зміни вказані вище – оновимо діаграму прецедентів. Результат зображено на рисунку 2.2.

Основною зміною стало впровадження системи, частиною якої є інформаційна технологія електронного голосування з якою взаємодіє ЦВК (або певний її підрозділ). Основними складовими системи є інформаційна технологія підрахунку голосів та модуль оптичного розпізнавання.

Також додатковою властивістю даної системи планується можливість сигналізувати про можливі порушення або підозрілі ситуації, які формуються на основі оброблюваних даних. Наприклад, поява бюлетенів у яких дублюються номери.

Порівнюючи діаграми, можна побачити, що формування результатів було делеговано частині інформаційної системи, а за обробку бюлетенів відповідає модуль оптичного розпізнання. Далі необхідно розробити алгоритм обробки бюлетенів інформаційною технологією для формування попередніх результатів.

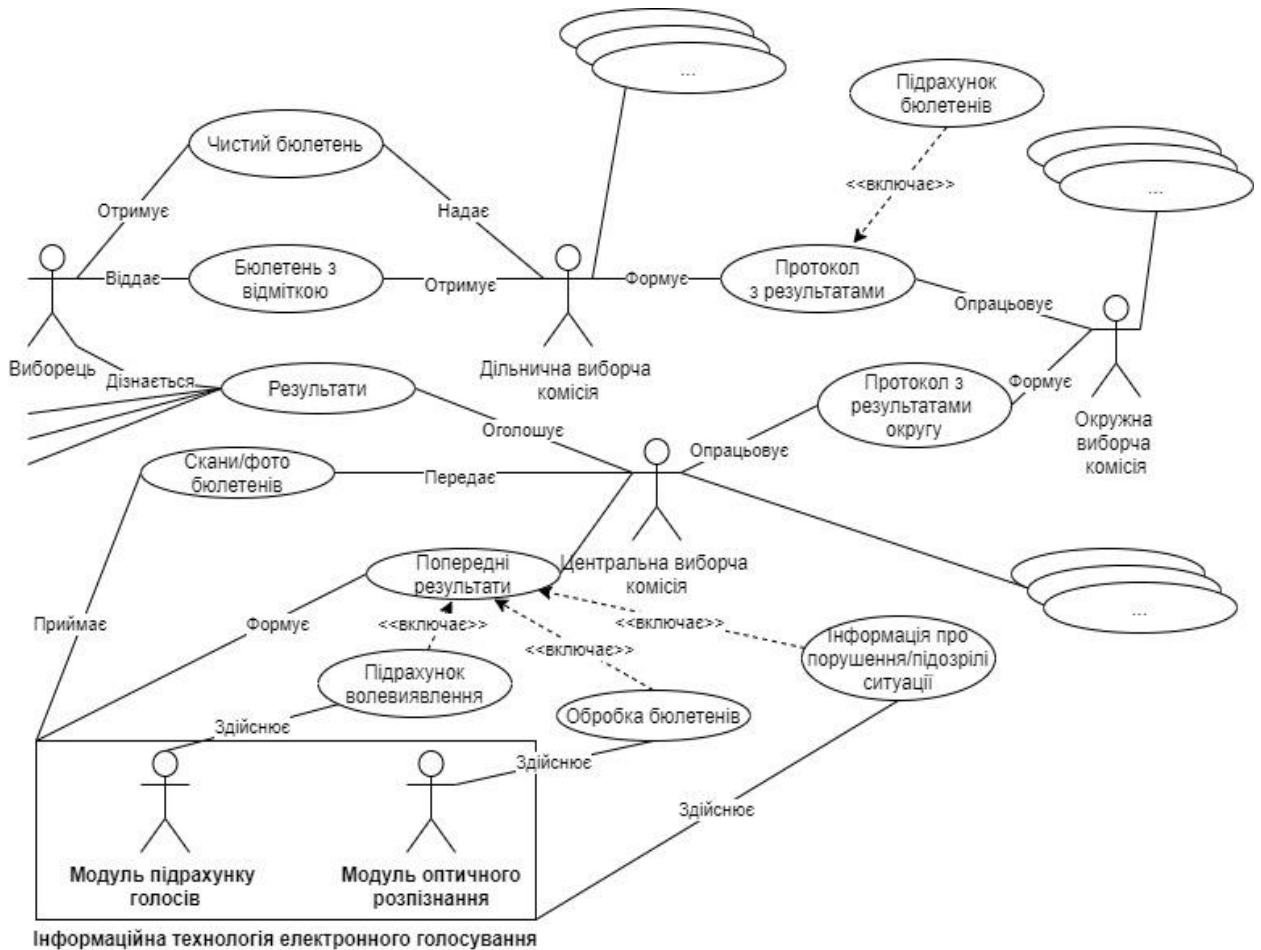


Рисунок 2.2 – Діаграма прецедентів виборчого процесу з OCR

2.2 Розробка алгоритму обробки бюлетенів

Бюлетень – офіційний документ для голосування на виборах, в якому виборець зазначає своє волевиявлення на підтримку окремих кандидатів та виборчих списків кандидатів, висунутих політичними партіями [13].

В ідеальному випадку його обробка полягає у визначенні клітинки, в якій виборець зробив позначку обравши свого кандидата.

Однак, в реальному житті все складніше. Можливі наступні випадки:

- виборець зробив більше однієї позначки;
- фальсифікований бюлетень;
- неправильно оформлений бюлетень;
- спроба двічі використати один і той самий бюлетень;
- інше.

З усіма такими випадками система повинна справлятися, оскільки вони можуть становити значний відсоток і бути ціленаправленими на дискредитацію волевиявлення.

Для того, щоб задовольнити всі вищезазначені умови і визначити подальші кроки розробимо алгоритм обробки бюлетенів. Результати роботи зображено на рисунку 2.3.

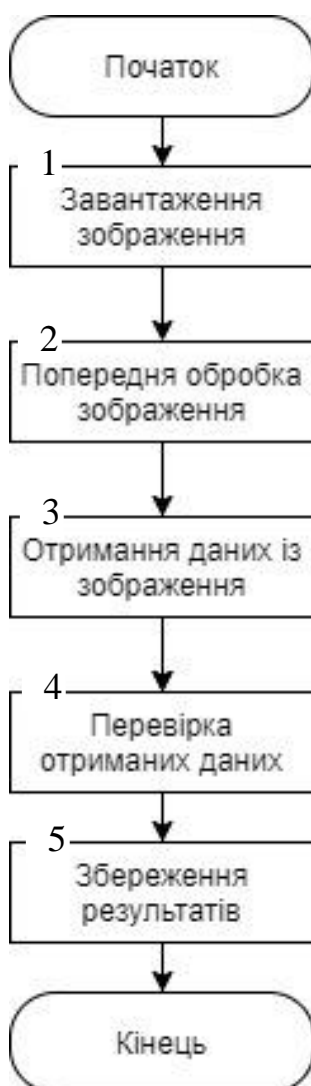


Рисунок 2.3 – Алгоритм обробки бюлетенів

Для більш детального аналізу, наведемо діаграму діяльності, зображену на рисунку 2.4.

Аналізуючи наведену діаграму діяльності, можна побачити, що найважливішими елементами системи є попередня обробка зображення та отримання даних з нього за допомогою OCR. Не менш важливими є організація отримання та передачі зображень між етапами, а також, збереження результатів роботи програми для формування результатів голосування. Однак це лише технічна частина.

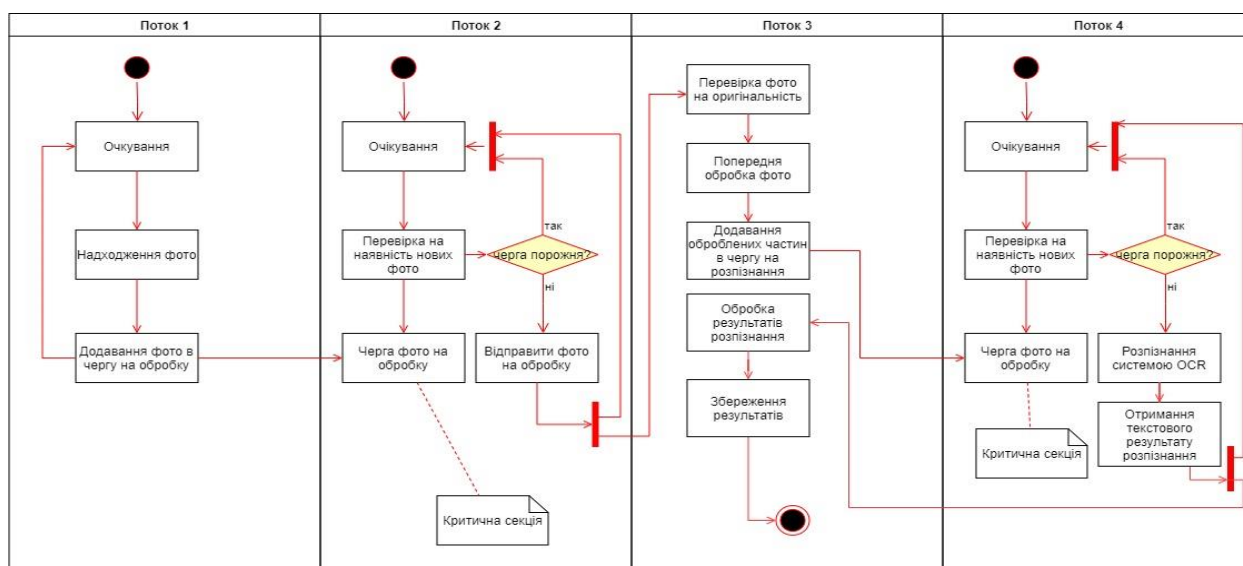


Рисунок 2.4 – Діаграма діяльності обробки бюлетенів

Для подальшої розробки технології визначимо метод попередньої обробки зображення та отримання даних.

2.3 Обґрунтування вибору технології отримання даних із зображення

Отримання даних з бюлетеня на сьогоднішній день зводиться до задачі OCR (optical character recognition).

OCR або оптичне розпізнавання символів – це механічне або електронне трансформування зображень будь якого тексту (рукописного, машинописного

або друкованого) в послідовність символів, що використовуються для представлення певної інформації в текстовому редакторі [14].

На сьогоднішній день точність розпізнання символів перевищує 99% [15], а існуюче різноманіття програмних засобів, таких як: ABBYY FineReader [16], Tesseract OCR [17], Amazon Textract [18], Google Cloud Vision [19] та багато інших, дозволяють отримати точність розпізнавання, яка задовольнить поставлену задачу.

Враховуючи вищесказане, сервіс для оптичного розпізнавання тексту буде обрано в розділі програмної реалізації даної інформаційної технології.

Однак, одного лише розпізнавання тексту замало, адже навіть невеличкий поворот фото на кілька градусів може призвести до значної похибки розпізнавання, що в свою чергу може серйозно вплинути не лише на достовірність, але і загалом на результат виборів. Саме для вирішення цієї ситуації необхідна попередня обробка зображення.

2.4 Визначення підходу до попередньої обробки зображення бюлетеня

В області науки даних та машинного навчання, 90% роботи полягає у підготовці даних, і лише 10% в написанні алгоритму та, власне, його навчанні.

Але навчання не є фінальним етапом. Дуже мало систем, здатні обробляти будь-яку інформацію одразу після навчання. Так чи інакше інформація (фото, тексти тощо) потребує додаткової обробки для одержання точніших результатів.

Попередня обробка зображення – дуже важливий, якщо не найважливіший етап розпізнавання тексту (використання готових систем) в наш час.

Пов'язано це, перш за все, з тим, що з одного і того ж зображення можна отримати абсолютно різну кількість інформації з різною точністю шляхом невеликих маніпуляцій.

Найпростішою з таких маніпуляцій є зміна кольору, а саме перетворення кольорового зображення в чорно-біле. Це дозволить більшості алгоритмів швидше та точніше визначати межі літер, знаків, символів тощо.

Однак, це покращення мало чим допоможе, якщо символ повернутий на кілька градусів (або більше). Звісно, можна спробувати збільшити навчальну вибірку, включивши в неї повернуті літери, але тоді її об'єм зросте в рази, що потребує не лише додаткових зусиль, але й часу. Але найбільше постраждає точність. Чудовою ілюстрацією є літери “Е” та “Ш”. Повернувши одну з них на 90 градусів, можна отримати літеру дуже схожу на іншу.

Враховуючи вищесказане, найкращим варіантом вирішення даної проблеми є вирівнювання фото з якого необхідно дістати інформацію.

Для вирівнювання зображень часто використовуються «особливі» точки.

«Особливі» точки – це точка зображення, що відповідає ряду властивостей:

1. Визначеність – полягає у необхідності виділення серед інших сусідніх точок.
2. Стійкість – зміна яскравості, контрастності і кольорової гами не повинна впливати на те, чи є ця точка особливою.
3. Інваріантність – стійкість до повороту, зміни масштабу зображення і ракурсу зйомки.
4. Стабільність – точка залишається особливою не залежно від шумів, які не перевищують певний поріг.
5. Інтерпретабельність – формат представлення повинен бути таким, що дає можливість використовувати точку для подальшої роботи
6. Кількість – знайдені точки повинні забезпечувати можливість знаходження об'єктів шляхом наявності.

Розробкою алгоритмів, що знаходять «особливі» точки зображено і описують їх певні властивості займаються вже досить довго. Постійно

відбуваються покращення існуючих або створення нових підходів для вирішення проблем швидкості і якості роботи.

Найпопулярнішими є алгоритми SIFT, SURF, KAZE, BRISK, A-KAZE, ORB однак перші два з них (SIFT та SURF) не можуть бути використані в даній роботі в зв'язку з їх патентними обмеженнями, а третій (KAZE) має значне покращення у вигляді A-KAZE Тому далі будемо розглядати BRISK та A-KAZE, ORB.

2.4.1 Алгоритм BRISK

BRISK – Binary Robust Invariant Scalable Keypoints. Цей метод представлений в 2011 [20]. Детектування «особливих» точок здійснюється за допомогою FAST [21] (Feature from Accelerated Segment Test), а дескриптором є BRIEF [22], але в їх роботу внесені певні зміни.

Для досягнення інваріантності до масштабу, пропонується вибрати найкращу «особливу» точку з максимальним значенням інтенсивності в «піраміді».

Пошук особливих точок

Приклад пошуку особливої точки зображено на рисунку 2.5.

“Піраміда” – певне перетворення, що складається з 4 “октав” c_i і 4 внутрішніх “октав” $d_i, i = 0 \dots 3$.

“Октава” – формується шляхом стиснення оригінального зображення c_0 в 2^i рази.

Внутрішні октави розташовані між c_i та c_{i+1} , представлені у вигляді стисненої c_0 в $\frac{2}{3} 2^i$ рази.

Дескриптор.

Область навколо «особливої» точки розбивається на N ділянок p (формула 2.1):

$$\mathcal{A} = \{(p_i, p_j) \in R^2 \times R^2 | i < N \wedge j < i \wedge i, j \in \mathbb{N}\} \quad (2.1)$$

Приклад таких ділянок показано на рисунку 2.6 (кількість «особливих» точок $N=60$).

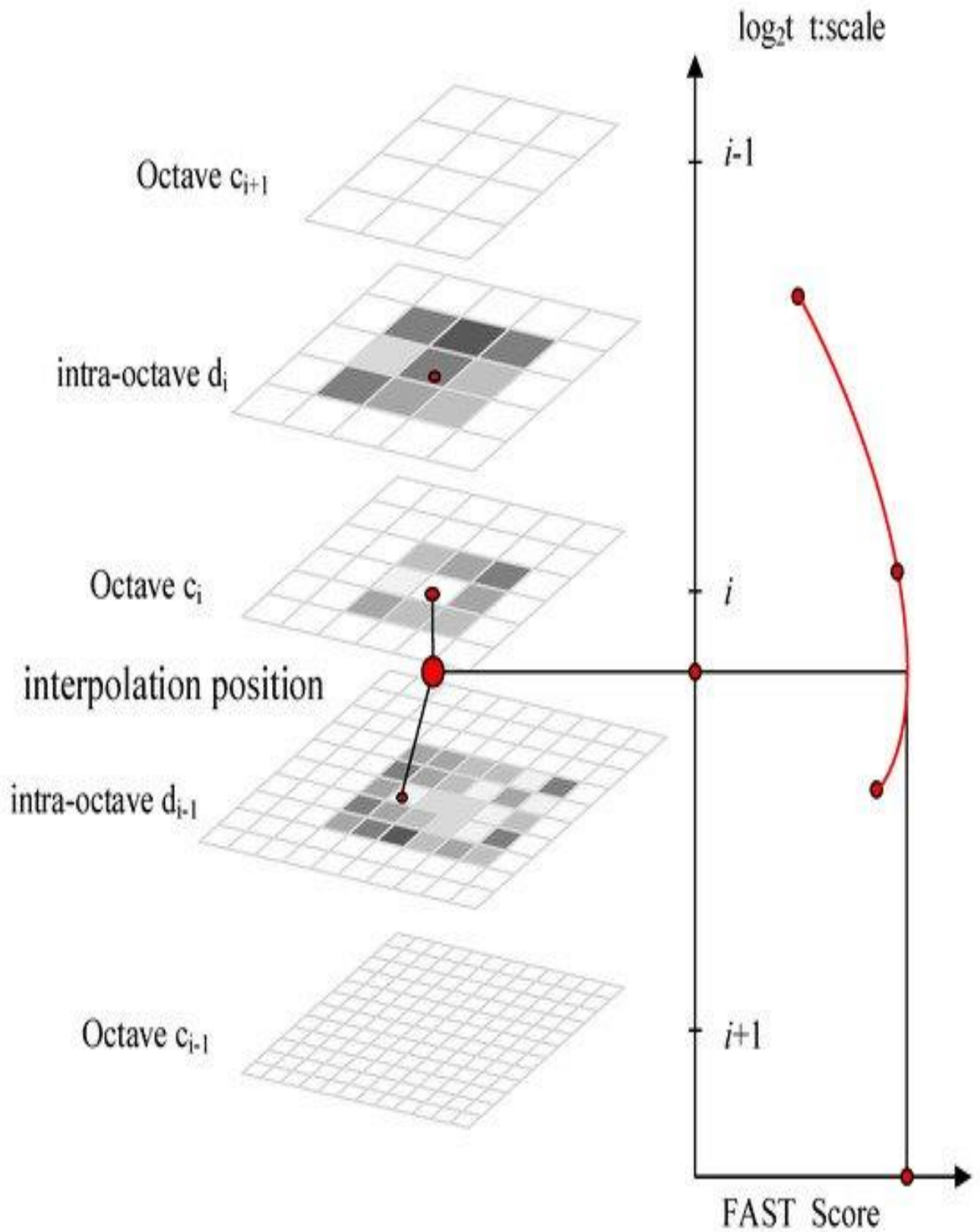


Рисунок 2.5 – Просторове зображення визначення «особливої» точки

Множина \mathcal{A} розбивається на 2-ві підмножини (формули 2.2 та 2.3):

$$\mathcal{S} = \{(p_i, p_j) \in \mathcal{A} \mid \|p_j - p_i\| < \delta_{\max}\} \subseteq \mathcal{A} \quad (2.2)$$

$$\mathcal{L} = \{(p_i, p_j) \in \mathcal{A} \mid \|p_j - p_i\| > \delta_{\min}\} \subseteq \mathcal{A} \quad (2.3)$$

, де $\delta_{\min} = 13,67t$, $\delta_{\max} = 9,75t$, t – розмір особливої точки.

Вираховується середня значення градієнта множини \mathcal{L} (формула 2.4):

$$g = \begin{pmatrix} g_x \\ g_y \end{pmatrix} = \frac{1}{|\mathcal{L}|} * \sum_{(p_i, p_j) \in \mathcal{L}} \left[(p_j - p_i) \frac{I(p_j, \sigma_j) - I(p_i, \sigma_i)}{\|p_j - p_i\|^2} \right] \quad (2.4)$$

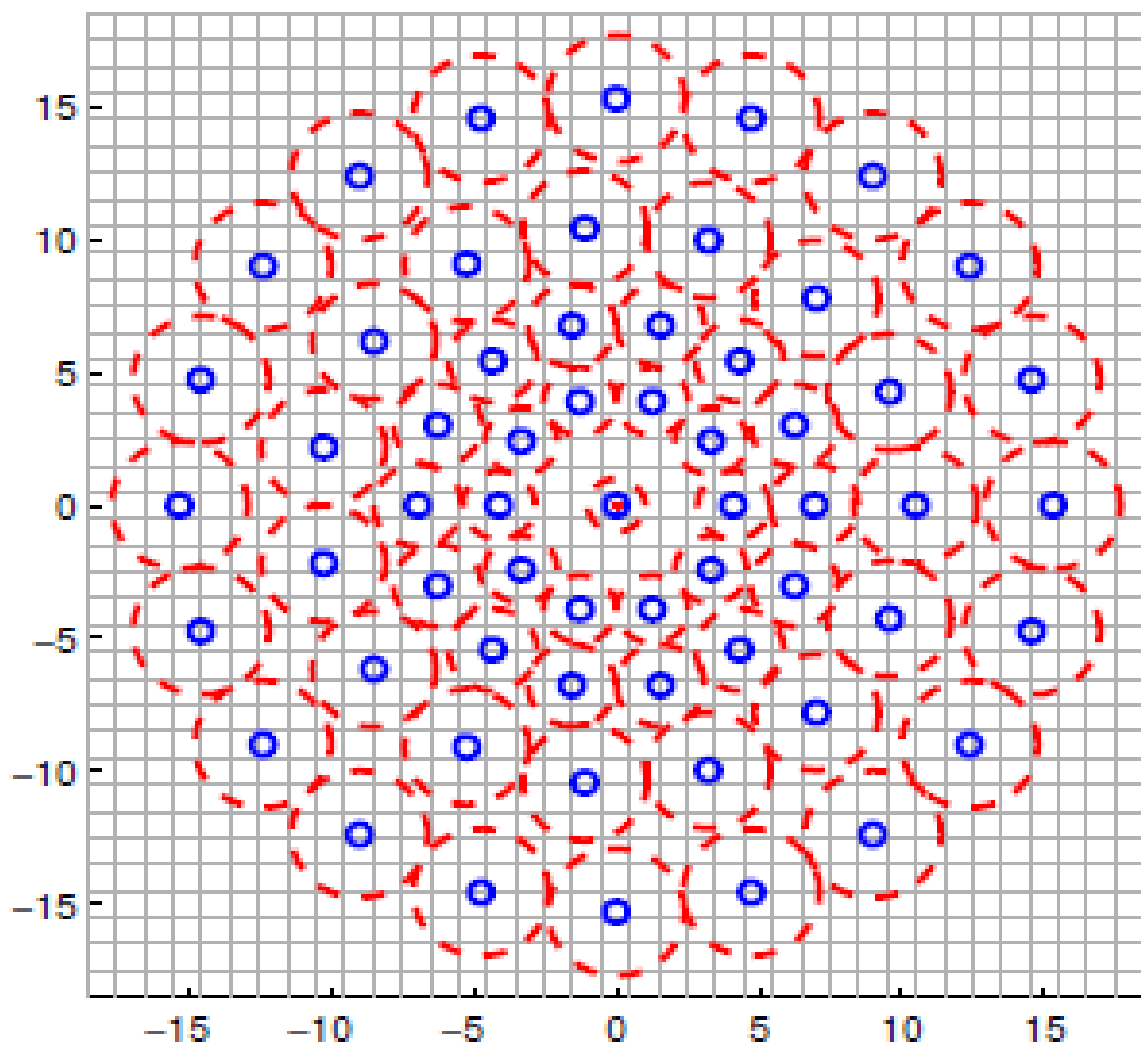


Рисунок 2.6 – Приклад вибору точок в дескрипторі BRISK

Дескриптор складається із бінарної стрічки довжиною 512, заповненої результатами проведених тестів в множині \mathcal{S} (формула 2.5):

$$b = \begin{cases} 1, I(p_j^\alpha, \sigma_j) > I(p_i^\alpha, \sigma_i) \\ 0, I(p_j^\alpha, \sigma_j) \leq I(p_i^\alpha, \sigma_i) \end{cases}; \forall (p_i^\alpha, p_j^\alpha) \in \mathcal{S}, \quad (2.5)$$

де $I(p_i^\alpha, \sigma_i)$ – інтенсивність оточення радіусом σ_i точки p_i ,
 $\alpha = \arctan2(g_y, g_x)$ – кут напрямку градієнту g .

2.4.2 Алгоритм A-KAZE

При розробці даного методу, що був представлений в 2013 році, намагалися досягти високої швидкості роботи як детектора, так і дескриптора. При цього знайдені «особливі» точки і їх дескриптори повинні були задовольняти високим показникам точності при порівнянні зображень [23].

Застосування алгоритму FED – Fast Explicit Diffusion на пірамідальній схемі дозволяє побудувати нелінійну багатомасштабну піраміду [24]. Застосування нелінійного коефіцієнта масштабування дозволяє збільшити швидкість знаходження необхідної особливої точки у порівнянні з Гаусівською пірамідою.

Обрахунок даного коефіцієнта базується на зміні яскравості зображення при масштабуванні.

Пошук особливих точок.

Для кожної октави L^i в піраміді обраховується визначник Гессе (формула 2.6).

$$L_{Hessian}^i = \sigma_{i,norm}^2 (L_{xx}^i L_{yy}^i - L_{xy}^i L_{xy}^i), \quad (2.6)$$

де $\sigma_{i,norm} = \frac{\sigma_i}{2^i}$ нормалізований відносно масштабу коефіцієнта для $L_{Hessian}^i$ з врахуванням розміру октави σ_i .

Похідна другого порядку вираховується за допомогою фільтра Шарра з кроком $\sigma_{i,norm}$. Даний фільтр дозволяє враховувати орієнтацію особливих точок. За допомогою такого підходу шукаємо такі точки в октаві, значення фільтру яких вище заданого порогу і є найбільшим із оточення точки 3×3 пікселів.

Далі для кожної точки з потенційних максимумів порівнюється її значення відносно результатів в сусідніх октавах $i + 1$ і $i - 1$ у вікні розміром $\sigma^i \times \sigma^i$ відповідно. В результаті розташування особливої точки оцінюється з субпіксельною точністю, що відповідає квадратичній функції до визначника Гессе в 3×3 сусідніх пікселів для пошуку максимуму.

Дескриптор M-LDB.

Початковий дескриптор LDB базувався на тих же принципах, що і BRIEF (буде розглянуто далі), але до порівняння показників яскравості областей додали порівняння значень градієнтів яскравості по осі x і y , внаслідок цього результат одного тесту складається із трьох бітів замість одного. Проведення тестів проводилося у вікні розміром 20×20 пікселів, поділеному на 4, 9 та 16 областей.

Але LDB має недоліки такі як не інваріантність до повертання та масштабування. І в якості рішення цих проблем в AKAZE використовується його покращена версія M-LDB:

1. Вікно дескриптора орієнтується по орієнтації «особливої» точки.
2. Інваріантність до масштабу отримана за допомогою вибору розміру вікна дескриптора в залежності від розміру октави σ_i в якій була знайдена його «особлива» точка.

На відміну від LDB, в M-LDB тести проводяться між середніми значеннями всіх пікселів в області, а між заданою їх кількістю в залежності від розміру σ_i . Це дозволяє пришвидшити обрахунок дескриптора.

Результуючий бінарний дескриптор має довжину 486 і має три складових.

2.4.3 Алгоритм ORB

ORB представлений в 2011 році [25]. В його основі лежить комбінація алгоритмів детектора FAST [21] (Features from Accelerated Segment Test) і дескриптор BRIEF [22] (Binary Robust Independent Elementary Features) з деякими покращеннями.

Пошук «особливих» точок.

Для пошуку «особливих» точок по черзі розглядаються окіл по 16 пікселів навколо кожного пікселя p .

Точка p вважається «підозрілою» на «особливу», якщо існує N пікселів в її околі довжиною 16 пікселів, якщо всі N яскравіше $I_p + t$ або темніше $I_p - t$, де I_p – яскравість точки p , t – порогова величина. При виконанні цієї умови далі досліджується значення яскравості на околі під номерами 1, 5, 9, 13. Приклад показано на рисунку 2.7.

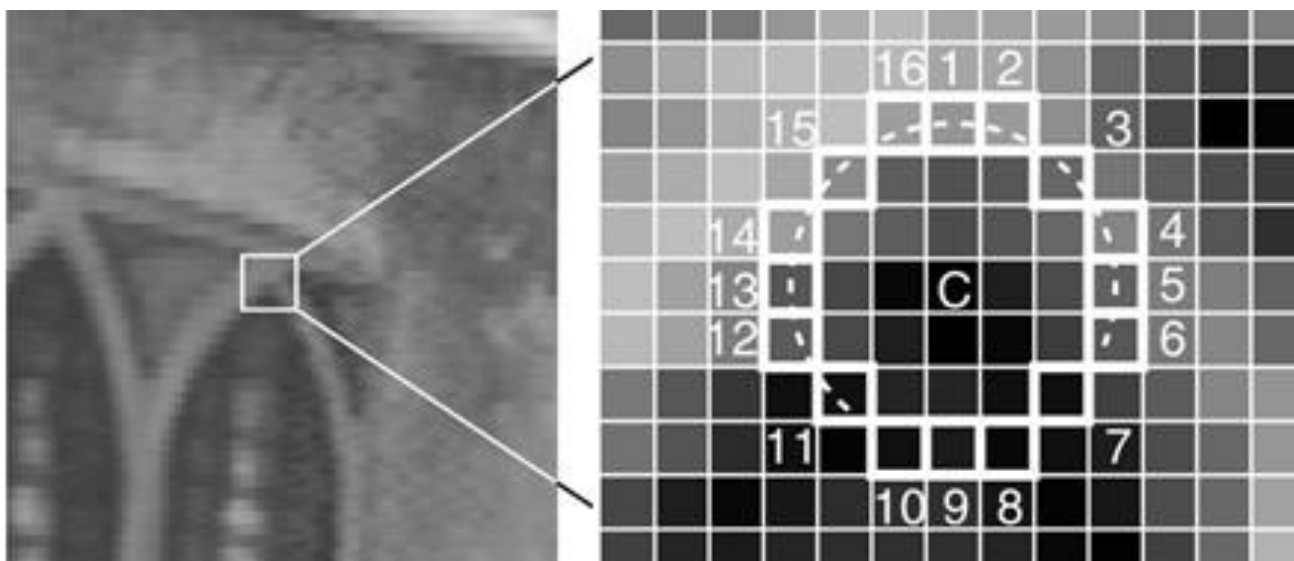


Рисунок 2.7 – Приклад околу точки для детектора

Якщо для трьох пікселів із чотирьох виконується умова $I_p + t < I_i$ або $I_i < I_p - t$, $i = 1 \dots 4$, тоді p вважається «особливою» точкою.

Вибір лише 4-ох пікселів на околі дозволяє швидко відсіяти точки, що не підходять, але в деяких випадках можливе визначення певних різних особливостей в одному околі. В алгоритмі ORB максимальна кількість «особливих» точок по замовчуванню не більше 500, якщо їх більше, то до них застосовується детектор кутів Харіса, для виключення найменш значимих.

Для інваріантності до масштабування застосовується вищеописаний алгоритм на піраміді Гауса. Октавами c_j є початкове зображення c_0 стиснуте з лінійним кроком.

Введення параметра кутової орієнтації дозволяє добитися стійкості детектування при повороті об'єкта. Він заснований на напрямках градієнта яскравості відносно центра точки, напрямлення з найбільшою інтенсивністю назначається орієнтацією «особливої» точки θ .

Дескриптор напрямків BRIEF.

Даний дескриптор представляється у вигляді вектору довжиною 256, що складається із результатів бінарних тестів навколо особливої точки. В околі 31×31 піксель порівнюються середнє значення яскравості між x та y , де x, y – області 5×5 пікселів (через розмиття) (формула 2.7):

$$\tau(I, x, y) = \begin{cases} 1: I_x < I_y \\ 0: I_x \geq I_y \end{cases}, \quad (2.7)$$

де I – середня яскравість обраної області.

Для досягнення інваріантності до обертання область обчислень дескриптора орієнтується по відношенню до особливої точки θ .

Всі $n = 256$ наборів x_i та y_i формують матрицю S розмірністю $2 \times n$. Далі S за допомогою матриці повороту R_θ орієнтується у відповідності з кутом θ (формули 2.8-2.10):

$$S = \begin{pmatrix} x_i \\ y_i \end{pmatrix}, \quad (2.8)$$

$$S_\theta = \begin{pmatrix} x'_i \\ y'_i \end{pmatrix}, \quad (2.9)$$

$$S_\theta = R_\theta S, \quad (2.10)$$

А сам вектор дескриптора записується як (формула 2.11):

$$g_n(I, \theta) = f_n(I) | (x_i, y_i) \in S_\theta$$

$$\text{де } f_n(I) = \sum_{1 \leq i \leq n} 2^{i-1} \tau(I, x_i, y_i) \quad (2.11)$$

2.4.4 Обґрунтування вибору алгоритму пошуку «особливих» точок

Враховуючи особливості розроблюваної системи, а саме:

- система здійснює розпізнавання шаблонних документів;
- переважна більшість зображень не буде мати значних шумів та буде сформована за допомогою сканера (однак, відповідно до вимог, система повинна мати можливість розпізнавати і фото зроблені на інші засоби з необхідною якістю);
- кількість зображень буде значна, що потребує швидкої їх обробки з високою точністю;

необхідно обрати такий алгоритм, який дозволить:

- визначити «особливі» точки так, щоб в подальшому їх можна було використовувати для співставлення зображення із шаблонним, вилучення необхідних частин і розпізнавання;
- стабільність визначення особливих точок при повороті;
- матиме високу швидкість роботи при прийнятній точності визначення «особливих» точок.

Для визначення такого алгоритму, що задовольнить поставлені вимоги, проаналізуємо опис обраних для порівняння алгоритмів наведений вище, їх особливості, а також деякі проведені дослідження [26].

Розглядаючи перший пункт вимог, можемо стверджувати, що будь-який з цих алгоритмів може бути використаний для співставлення з шаблонним зображенням з подальшим вилученням необхідних частин та їх розпізнавання.

Стосовно другого пункту, розглянемо ряд графіків з проведених досліджень.

Лінії графік будуть відповідати легенді наведеній на рисунку 2.8. Також варто зазначити, що ORB(1000) та BRISK(1000) означає обмеження в кількості «особливих» точок, щоб відсіяти ті, які найменше відрізняються за яскравістю.

Аналізуючи графік, що показує залежність між кутом повороту та повторюваністю знайдених «особливих» точок (рис. 2.9) можна побачити, що ORB(1000) має значно вищу повторюваність при визначені «особливих» при будь-якому куті повороту, що говорить про його стабільність.

Legend	
—	SIFT
—	SURF(128D)
⋯	SURF(64D)
—	KAZE
—	AKAZE
—	ORB
⋯	ORB(1000)
—	BRISK
⋯	BRISK(1000)

Рисунок 2.8 – Легенда графіків

При розгляді третьої вимоги слід звернути увагу на таблицю 2.1 та таблицю 2.2, які були сформовані внаслідок експериментів проведених на різних наборах зображень.

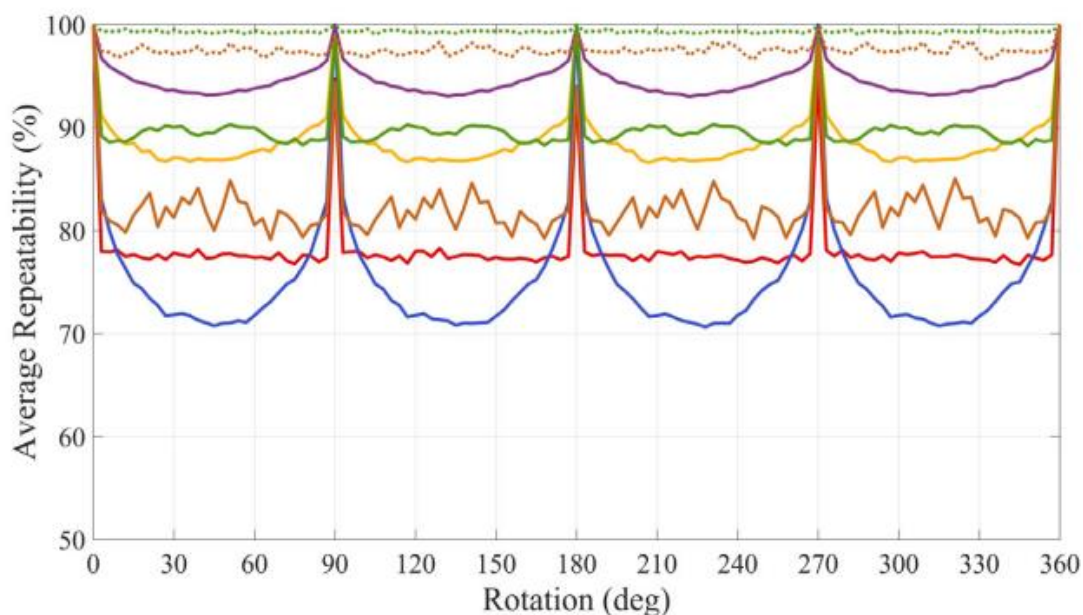


Рисунок 2.9 – Графік залежності середньої повторюваності «особливих» точок від кута повороту

В таблиці 2.2 загальний час дещо більше ніж сума інших стовпців, оскільки наведено, оскільки наведено найбільш затратні етапи роботи алгоритму.

Таблиця 2.1 – Кількісне порівняння алгоритмів (в середньому)

Назва	Кількість «особливих» точок		Співставленні «особливі» точки	Відкинуті «особливі» точки
	Зображення 1	Зображення 2		
AKAZE	1633.0	1776.0	231.5	33.0 14,3%
ORB	9782.7	10828.0	742.2	162.7 21,9%
ORB(1000)	955.2	955.7	122.8	21.2 17.3%
BRISK	6612.3	7476.8	544.7	113.7 20.9%
BRISK(1000)	966.0	958.7	127.5	20.0 15.7%

Таблиця 2.2. – Порівняння часу затраченого на роботу алгоритмів (в середньому)

Назва	Пошук «особливих» точок		Співставлення «особливих» точок	Загальний час співставлення
	Зображення 1	Зображення 2		
AKAZE	0.0995	0.1013	0.0437	0.2502
ORB	0.0385	0.0427	1.0530	1.1410
ORB(1000)	0.0129	0.0133	0.0113	0.0426
BRISK	0.1253	0.1253	0.9319	1.1735
BRISK(1000)	0.0206	0.0206	0.0149	0.0609

Аналізуючи дані наведених вище таблиць, можна побачити, що за точністю лідирує AKAZE, але його перевагу над ORB(1000) та BRISK(1000), які показують майже однакові результати, не можна вважати значною. Але, якщо врахувати швидкість, то AKAZE значно поступається конкурентам (в кілька разів!!!), а ORB(1000) має перевагу більше ніж в 1,5 рази над BRISK(1000), що дає підстави вважати його найкращим варіантом для поставлених вимог.

2.5 Структура інформаційної технології підрахунку голосів на виборах

Враховуючи все вищезазначене, інформаційна технологія підрахунку голосів на виборах буде містити наступні елементи, зображені на рисунку 2.10:

- інтерфейс користувача;
- система підрахунку голосів (включаючи модуль обробки зображень, модуль обробки інформації та модуль авторизації);
- база даних;
- черга повідомлень.

Також для повноцінного функціонування технології необхідна система розпізнавання інформації із зображень.

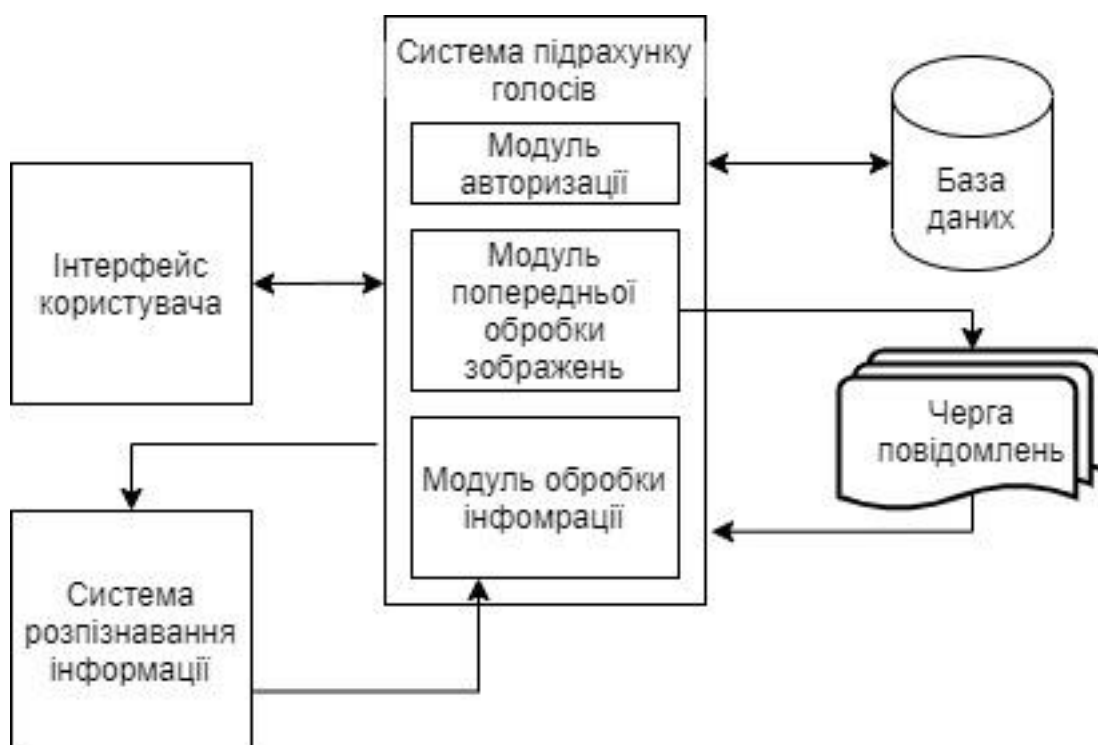


Рисунок 2.10 – Узагальнена структурна схема інформаційної технології

2.6 Висновок

В цьому розділі було: визначено місце у виборчому процесі для застосування системи; визначено основні елементи шляхом створення діаграми діяльності інформаційної системи та обґрунтовано вибір ORB алгоритму для попередньої обробки фото. Розроблено структуру інформаційної технології підрахунку голосів на виборах.

3 ПРОГРАМНА РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ПІДРАХУНКУ ГОЛОСІВ НА ВИБОРАХ

3.1 Обґрунтування вибору мови програмування

Прийнято вважати, що програми написані на Python, в цілому працюють повільніше, ніж програми на Java, але при цьому їх розробка займає набагато менше часу.

Програми на Python зазвичай в 3-5 разів коротше, ніж їх аналоги на Java. Ця різниця може бути приписана вбудованим в Python на високому рівні типів даних і їх динамічній типізації.

Наприклад, програміст, що пише програми на Python, не витрачає час на оголошення типів аргументів чи змінних, а багаточисельні види поліморфних списків і словників, синтаксична підтримка яких вбудована в саму мову, знаходять застосування в кожній програмі на Python.

Із-за динамічної типізації виконання програми на Python іде важче ніж з Java. Наприклад, рахуючи вираз $a+b$, програма повинна спочатку визначити тип об'єктів a та b , адже він невідомий при компіляції. Потім вона шукає операцію додавання, яка може бути створена користувачем.

Java, в свою чергу, може виконувати ефективні складання цілих або чисел з плаваючою комою, але потребує декларування змінних a і b та не дозволяє перевизначити оператор $+$, наприклад, окремим користувацьким класам.

Виходячи з цього, Python значно краще підходить в якості мови, що інтегрує, в той час як Java, порівняно з Python, скоріше в якості низькорівневої мови програмування. Фактично, разом вони утворюють собою прекрасну комбінацію. Компоненти можуть розроблятися на Java і комбінуватися для формування додатків в Python. Python також може бути використаний для прототипування компонентів та хорошої дослідницького інструменту, який може використовувати складні обчислення.

Майже все сказане про Java відноситься і до C++, але з обмовкою. Якщо код на Python в 3-5 разів коротше ніж його еквівалент на Java, то у випадку з C++ він коротше в 5-10 разів! Не пов'язані між собою випадкові приклади

свідчать, що один програміст на Python може зробити за 2 місяці те, що два програмісти на C++ не можуть завершити за рік.

Python чудовий у якості мови інтегрування, коли використовується для зборки компонентів, написаних на C++.

Підсумовуючи все вищесказане, виділимо основні переваги та недоліки кожної з розглянутих мов.

Python – потужна мова програмування, яка підтримує багато парадигм, оптимізована під забезпечення високої продуктивності програмістів, читабельності коду і якості програмного забезпечення [27].

Переваги:

- Простий у написанні;
- Низький поріг входження;
- Високорівнева;
- Лаконічний (потрібно менше кількість рядків коду для написання більшості речей ніж на інших мовах програмування);
- Велика кількість бібліотек та фреймворків для роботи з ШІ, МН та ГН;
- Може бути використаний для розробки майже будь-якого продукту (web, desktop, etc.).

Недоліки:

- Низька швидкодія;

C++ -- мова програмування високого рівня з підтримкою декількох парадигм програмування: об'єктно-орієнтованої, узагальненої та процедурної. Створена Б'ярном Страуструпом [28].

Переваги:

- Висока швидкодія;
- Може бути використаний для розробки майже будь-якого продукту (web, desktop, etc.).

Недоліки:

- Складний у написанні;

- Високий поріг входження;
- Низькорівнева;
- Багатослівна у порівнянні з Python (через низькорівневність);
- Невелика кількість бібліотек для ШІ, МН;

Java – об'єктно-орієнтована мова програмування, випущена компанією Sun Microsystems у 1995 році як основний компонент платформи Java.[29].

Переваги:

- Високорівнева;
- Висока швидкодія;
- Може бути використаний для розробки майже будь-якого продукту (web, desktop, etc.).

Недоліки:

- Складний у написанні;
- Високий поріг входження;
- Багатослівна у порівнянні з Python;
- Невелика кількість бібліотек для ШІ, МН.

Зважаючи на недоліки та переваги кожної з мов, для розробки програмного засобу було обрано мову Python.

3.2 Обґрунтування вибору середовища розробки

Для розробки програмного продукту було обрано інтегроване середовище розробки PyCharm.

PyCharm — інтегроване середовище розробки для мови програмування Python. Надає засоби для аналізу коду, інструмент для запуску юніт-тестів і підтримує веб-розробку на Django. PyCharm розроблена чеською компанією JetBrains на основі IntelliJ IDEA.

PyCharm працює під операційними системами Windows, Mac OS X і Linux.

PyCharm був випущений на ринок інтегрованих середовищ розробки для Python щоб створити конкуренцію з PyDev і поширенішим середовищем розробки Komodo IDE. Бета-версія була випущена в липні 2010, версія 1.0 була випущена на три місяцями пізніше. Версія 2.0 вийшла 13 грудня 2011. Версія 3.0 була випущена 24 вересня 2013. PyCharm Community Edition, безкоштовна версія з відкритим сирцевим кодом була опублікована 22 жовтня 2013 [30].

- Головними перевагами PyCharm вважається:
- Статичний аналіз коду, підсвічування синтаксису і помилок.
- Навігація серед проектів і сирцевого коду: відображення файлової структури проекту, швидкий перехід між файлами, класами, методами і використанням методів.
- Рефакторинг: перейменування, витяг методу, введення змінної, введення константи, підняття і опускання методу тощо.
- Інструменти для веб-розробки з використанням фреймворку Django.
- Вбудований зневаджувач для Python
- Вбудовані інструменти для юніт-тестування
- Розробка з використанням Google App Engine
- Підтримка систем контролю версій: загальний користувацький інтерфейс для Mercurial, Git, Subversion, Perforce і CVS з підтримкою списків змін та злиття.

3.3 Обґрунтування вибору модуля OCR

Оптичне розпізнавання символів (коротко: OCR) – завдання автоматичного вилучення тексту з зображень. На сьогоднішній день доступні різноманітні інструменти та сервіси, які прості у використанні та роблять це завдання просто вирішеним та таким, що не потрібно «винаходити велосипед заново». В цій роботі буде порівняно чотири найпопулярніші інструменти:

1. Tesseract OCR

2. ABBYY FineReader
3. Google Cloud Vision
4. Amazon Textract

Зважаючи на вимоги до розроблюваної технології, зображення документів бувають різної форми та якості. Це можуть бути як скановані, так і фотографії зі смартфонів, планшетів або інших пристроїв, що мають вбудовану камеру. Крім друкованого тексту, вони також можуть містити структурні елементи, такі як поля та таблиці. Таким чином, ідеальний інструмент OCR повинен:

- надійно розпізнати добре відсканований текст,
- бути стійким до поганої якості зображення,
- вихідна інформація про форматування та структуру документа.

Для порівняння візьмемо зображення на рисунках 3.1-3.3:

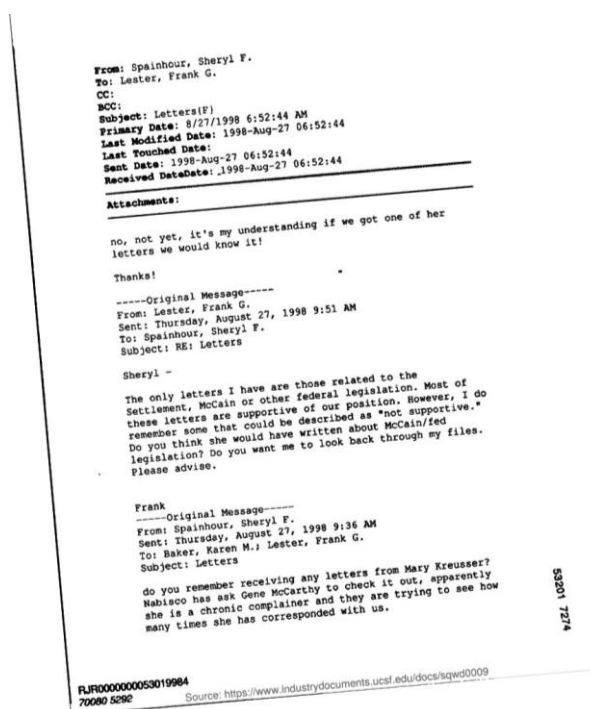


Рисунок 3.1 – Перше фото для порівняння

From: Spainhour, Sheryl F.
 To: Lester, Frank G.
 CC:
 Subject: Letters(F)
 Primary Date: 8/27/1998 6:52:44 AM
 Last Modified Date: 1998-Aug-27 06:52:44
 Last Touched Date:
 Sent Date: 1998-Aug-27 06:52:44
 Received Date: 1998-Aug-27 06:52:44

Attachments:

no, not yet, it's my understanding if we got one of her letters we would know it!

Thanks!

-----Original Message-----
 From: Lester, Frank G.
 Sent: Thursday, August 27, 1998 9:51 AM
 To: Spainhour, Sheryl F.
 Subject: RE: Letters

Sheryl -

The only letters I have are those related to the Settlement, McCain or other federal legislation. Most of these letters are supportive of our position. However, I do remember some that could be described as "not supportive." Do you think she would have written about McCain/fed legislation? Do you want me to look back through my files. Please advise.

Frank
 -----Original Message-----
 From: Spainhour, Sheryl F.
 Sent: Thursday, August 27, 1998 9:36 AM
 To: Baker, Karen M.; Lester, Frank G.
 Subject: Letters

do you remember receiving any letters from Mary Kreusser? Nabisco has ask Gene McCarthy to check it out, apparently she is a chronic complainer and they are trying to see how many times she has corresponded with us.

RJR000000053019984
 70080 5292

Source: <https://www.industrydocuments.ucsf.edu/docs/sqwd0009>

53201 7274

Рисунок 3.2 – Друге фото для порівняння

Результат порівнянь наведено в таблиці 3.1 [30]:

Таблиця 3.1 – Результат порівняння OCR інструментів

OCR інструмент	Відсканований документ	Фото смартфона	Витяг таблиці
Tesseract OCR	Прийнятне	Погано	Погано
ABBYY FineReader	Добре	Добре	Добре
Google Cloud Vision	Добре	Добре	Погано
Amazon Textract	Добре	Прийнятне	Прийнятне

Після проведеного порівняння найкращим варіантом виявився ABBYY FineReader.

Він має ряд переваг:

- Наявність API SDK – інструмент для розробки за допомогою даного пакету;
- Висока якість розпізнавання;
- Можливість використання без прив'язки до інтернету;
- Має багато форматів для виводу результату;
- Низький поріг входження;
- Хороша робота з таблицями, адже саме вигляд таблиці мають бюлетені (або значна їх частина).

Але має і недолік: платний план використання. Але для значних обсягів, середня вартість буде прийнятною.

3.4 Опис складових програмної реалізації

Система електронного голосування є дуже складною і такою, що потребує роботи багатьох спеціалістів, в тому числі спеціалістів з безпеки. Тому в даній роботі буде розроблена дещо спрощена система, але така, що так само буде виконувати поставлені задачі, окрім гарантування безпеки та

витримки високих навантажень, але розроблені алгоритми дозволять допрацювати систему та виконувати всі поставлені задачі з тим же успіхом.

Для реалізації даної системи було використано ряд бібліотек, фреймворків та технологій, які будуть описані нижче.

Ключовою бібліотекою для реалізації даної системи стала бібліотека OpenCV.

OpenCV – бібліотека алгоритмів та функцій комп’ютерного зору, чисельних алгоритмів загального призначення, обробки зображень з відкритим кодом. Бібліотека надає можливість обробляти та аналізувати вміст зображень, у тому числі відстежування руху об’єктів, аналіз вмісту та перетворення зображень, застосування алгоритмів машинного навчання і знаходження спільного на різних зображеннях. Головною особливістю даної бібліотеки є те, що вона вільно може бути використана в академічних цілях (а також комерційних) через те, що поширюється в умовах ліцензії BSD, що мають не так вже і багато хороших бібліотек пов’язаних з комп’ютерним зором [31].

Для створення сайту та системи підрахунку голосів як такої, було використано фреймворк Django.

Django – Python-фреймворк з відкритим кодом, що використовується для розробки веб-систем. Сайти на Django складаються з однієї або кількох частин, які зачасту робляться модульними, що в свою чергу дозволяє робити системи досить гнучкими і покращувати частини не змінюючи при цьому загальної архітектури [32].

На початку, ціллю створення фреймворку Django була робота з новинними ресурсами, що сильно позначилося на його архітектурі, а саме те, що він дає можливості для швидкої розробки веб-сайтів інформаційного характеру, чим в певній мірі є розроблювана система.

Для зберігання даних було обрано PostgreSQL.

PostgreSQL – вільна об’єктно-орієнтована система керування базами даних.

Сильними сторонами PostgreSQL вважаються:

- висока продуктивність і надійні механізми транзакцій і реплікації;
- система вбудованих мов програмування, що здатна до розширення. Однак стандартна поставка вже містить PL/Python;
- розширюваність (можливість створювати нові типи даних, типи індексів, мови програмування, модулі розширення, підключати будь-які зовнішні ресурси даних [33]).

Всі вищевказані переваги мають пряме відношення до розроблюваної системи і є чудовим поясненням використання саме даної системи керування базами даних.

Для забезпечення швидкодії та зв'язку між різними модулями системи було використано RabbitMQ.

RabbitMQ – платформа, що реалізує систему, основною задачею якої є обмін повідомленнями між модулями програмної розробки [34]. Для передачі повідомлень використовується стандарт AMQP (Advanced Message Queuing Protocol).

До переваг системи слід віднести наступне:

- використання вищевказаного протоколу;
- підтримка масштабування (горизонтального) для побудови кластерної або модульної архітектури;
- можливість збереження даних на диск;
- наявність клієнта мовою Python.

Окрім того, завдяки універсальності Django, всі ці компоненти дуже просто об'єднуються в ньому і не потребують багато часу для налаштування.

3.5 Тестування інформаційної системи підрахунку голосів на виборах та аналіз результатів

Тестування даної системи буде полягати у перевірці виправлення недоліків системи, опис яких було наведено в першому розділі.

Для тестування скористаємося бюлетенем зображеним на рисунку 3.4

Місцеві вибори 25 жовтня 2020 року

ВИБОРЧИЙ БЮЛЕТЕНЬ

<p>Багатомандатний виборчий округ з виборів депутатів Київської міської ради</p>	<p>Територіальний виборчий округ №6 Виборча дільниця № 856 395</p>
<p>ПЕРШИЙ КРОК Зробіть одну позначку "плюс" або іншу, яка засвідчує ваше волевиявлення у лівому квадраті проти назви місцевої організації політичної партії</p>	<p>ДРУГИЙ КРОК Занесіть в трафарет номер обраного вами кандидата зі списку політичної партії, яку ви обрали (за бажанням)</p>
<p>1 Політична партія «УКРАЇНСЬКЕ ОБ'ЄДНАННЯ ПАТРІОТІВ – УКРОП» Бабенко Володимир Сергійович</p>	<div style="display: flex; align-items: center; gap: 20px;"> <div style="border: 1px solid black; width: 60px; height: 60px; display: flex; align-items: center; justify-content: center;"> <div style="width: 30px; height: 30px; border: 1px solid black;"></div> <div style="width: 30px; height: 30px; border: 1px solid black; display: flex; align-items: center; justify-content: center;"> <div style="width: 15px; height: 15px; border: 1px dashed black;"></div> <div style="width: 15px; height: 15px; border: 1px dashed black;"></div> </div> </div> <div style="display: flex; flex-direction: column; gap: 5px;"> <p>☐1 Дашкевич Г.І.</p> <p>☐2 Остапенко К.Я.</p> <p>☐3 Марченко Л.І.</p> <p>☐4 Василів К.Я.</p> </div> <div style="display: flex; flex-direction: column; gap: 5px;"> <p>☐5 Овчинникова Ю.Ю.</p> <p>☐6 Константинов К.К.</p> <p>☐7 Новосад Г.І.</p> <p>☐8 Марченко Л.І.</p> </div> </div>
<p>2 Політична партія «Слуга Народу» Переможцева Євгенія Петрівна</p>	<div style="display: flex; align-items: center; gap: 20px;"> <div style="border: 1px solid black; width: 60px; height: 60px; display: flex; align-items: center; justify-content: center;"> <div style="width: 30px; height: 30px; border: 1px solid black;"></div> <div style="width: 30px; height: 30px; border: 1px solid black; display: flex; align-items: center; justify-content: center;"> <div style="width: 15px; height: 15px; border: 1px dashed black;"></div> <div style="width: 15px; height: 15px; border: 1px dashed black;"></div> </div> </div> <div style="display: flex; flex-direction: column; gap: 5px;"> <p>☐1 Гончаренко Г.І.</p> <p>☐2 Шевченко К.Я.</p> <p>☐3 Димов К.Я.</p> <p>☐4 Наконечний К.Я.</p> </div> <div style="display: flex; flex-direction: column; gap: 5px;"> <p>☐5 Леонов Ю.Ю.</p> <p>☐6 Поліщук К.К.</p> <p>☐7 Хмельницький Г.І.</p> <p>☐8 Підпригора Л.І.</p> </div> </div>
<p>3 Політична партія «Опозиційна платформа – За життя» Волноваха Сергій Петрович</p>	<div style="display: flex; align-items: center; gap: 20px;"> <div style="border: 1px solid black; width: 60px; height: 60px; display: flex; align-items: center; justify-content: center;"> <div style="width: 30px; height: 30px; border: 1px solid black;"></div> <div style="width: 30px; height: 30px; border: 1px solid black; display: flex; align-items: center; justify-content: center;"> <div style="width: 15px; height: 15px; border: 1px dashed black;"></div> <div style="width: 15px; height: 15px; border: 1px dashed black;"></div> </div> </div> <div style="display: flex; flex-direction: column; gap: 5px;"> <p>☐1 Олійник Г.І.</p> <p>☐2 Панасов К.Я.</p> <p>☐3 Джегур Л.І.</p> <p>☐4 Тищенко К.Я.</p> </div> <div style="display: flex; flex-direction: column; gap: 5px;"> <p>☐5 Даліба Ю.Ю.</p> <p>☐6 Кошляк К.К.</p> <p>☐7 Балас Г.І.</p> <p>☐8 Музальов Л.І.</p> </div> </div>
<p>4 Політична партія Всеукраїнське об'єднання «Батьківщина» Кулеба Олег Миколайович</p>	<div style="display: flex; align-items: center; gap: 20px;"> <div style="border: 1px solid black; width: 60px; height: 60px; display: flex; align-items: center; justify-content: center;"> <div style="width: 30px; height: 30px; border: 1px solid black;"></div> <div style="width: 30px; height: 30px; border: 1px solid black; display: flex; align-items: center; justify-content: center;"> <div style="width: 15px; height: 15px; border: 1px dashed black;"></div> <div style="width: 15px; height: 15px; border: 1px dashed black;"></div> </div> </div> <div style="display: flex; flex-direction: column; gap: 5px;"> <p>☐1 Беркут Г.І.</p> <p>☐2 Петрик К.Я.</p> <p>☐3 Денисенко Л.І.</p> <p>☐4 Коробчук К.Я.</p> </div> <div style="display: flex; flex-direction: column; gap: 5px;"> <p>☐5 Колосенко Ю.Ю.</p> <p>☐6 Савчук К.К.</p> <p>☐7 Ореховський Г.І.</p> <p>☐8 Іщенко Л.І.</p> </div> </div>
<p>5 Політична партія «Партія економічного відродження України» Чук Василь Григорович</p>	<div style="display: flex; align-items: center; gap: 20px;"> <div style="border: 1px solid black; width: 60px; height: 60px; display: flex; align-items: center; justify-content: center;"> <div style="width: 30px; height: 30px; border: 1px solid black;"></div> <div style="width: 30px; height: 30px; border: 1px solid black; display: flex; align-items: center; justify-content: center;"> <div style="width: 15px; height: 15px; border: 1px dashed black;"></div> <div style="width: 15px; height: 15px; border: 1px dashed black;"></div> </div> </div> <div style="display: flex; flex-direction: column; gap: 5px;"> <p>☐1 Сербин Г.І.</p> <p>☐2 Гейло К.Я.</p> <p>☐3 Поляруш Л.І.</p> <p>☐4 Мазуревич К.Я.</p> </div> <div style="display: flex; flex-direction: column; gap: 5px;"> <p>☐5 Підпалій Ю.Ю.</p> <p>☐6 Гламазда К.К.</p> <p>☐7 Король Г.І.</p> <p>☐8 Грисяк Л.І.</p> </div> </div>
<p>6 Політична партія Всеукраїнське об'єднання «Свобода» Сівак Світлана Теодівна</p>	<div style="display: flex; align-items: center; gap: 20px;"> <div style="border: 1px solid black; width: 60px; height: 60px; display: flex; align-items: center; justify-content: center;"> <div style="width: 30px; height: 30px; border: 1px solid black;"></div> <div style="width: 30px; height: 30px; border: 1px solid black; display: flex; align-items: center; justify-content: center;"> <div style="width: 15px; height: 15px; border: 1px dashed black;"></div> <div style="width: 15px; height: 15px; border: 1px dashed black;"></div> </div> </div> <div style="display: flex; flex-direction: column; gap: 5px;"> <p>☐1 Ланович Г.І.</p> <p>☐2 Бабій К.Я.</p> <p>☐3 Станько Л.І.</p> <p>☐4 Вікарчук К.Я.</p> </div> <div style="display: flex; flex-direction: column; gap: 5px;"> <p>☐5 Тігаренко Ю.Ю.</p> <p>☐6 Слоньовський К.К.</p> <p>☐7 Дубина Г.І.</p> <p>☐8 Бандусяк І.Л.І.</p> </div> </div>
<p>7 Політична партія «Сила Людей» Солонтай Олександр Гейзович</p>	<div style="display: flex; align-items: center; gap: 20px;"> <div style="border: 1px solid black; width: 60px; height: 60px; display: flex; align-items: center; justify-content: center;"> <div style="width: 30px; height: 30px; border: 1px solid black;"></div> <div style="width: 30px; height: 30px; border: 1px solid black; display: flex; align-items: center; justify-content: center;"> <div style="width: 15px; height: 15px; border: 1px dashed black;"></div> <div style="width: 15px; height: 15px; border: 1px dashed black;"></div> </div> </div> <div style="display: flex; flex-direction: column; gap: 5px;"> <p>☐1 Новосад Г.І.</p> <p>☐2 Димов К.Я.</p> <p>☐3 Марченко Л.І.</p> <p>☐4 Василів К.Я.</p> </div> <div style="display: flex; flex-direction: column; gap: 5px;"> <p>☐5 Овчинникова Ю.Ю.</p> <p>☐6 Константинов К.К.</p> <p>☐7 Новосад Г.І.</p> <p>☐8 Марченко Л.І.</p> </div> </div>

Рисунок 3.4 – Шаблон бюлетеня

Оскільки, для розпізнавання зображення буде обрано один з лідерів систем розпізнавання, то перевірити слід розроблювану частину попередньої обробки бюлетенів.

Для цього базовим тестом буде перевірка правильності розміщення сканованого, вертикально розміщеного бюлетеня з голосом виборця.

Застосувавши розроблений модуль, на сканованому зображенні і після цього наклавши його на шаблон з прозорістю 50% було отримано зображення показане на рисунку 3.5

Місцеві вибори 25 жовтня 2020 року

ВИБОРЧИЙ БЮЛЕТЕНЬ

Багатомандатний виборчий округ
з виборів депутатів Київської міської ради

Територіальний виборчий округ №6
Виборча дільниця № 856 395

ПЕРШИЙ КРОК
Зробіть одну позначку "плюс" або іншу, яка засвідчує ваше волевиявлення у лівому квадраті проти назви місцевої організації політичної партії

ДРУГИЙ КРОК
Занесіть в трафарет номер обраного вами кандидата зі списку політичної партії, яку ви обрали (за бажанням)

1	<p>Політична партія «УКРАЇНСЬКЕ ОБ'ЄДНАННЯ ПАТРІОТІВ – УКРОП» Бабенко Володимир Сергійович</p>		<p><input type="checkbox"/>1 Дашкевич Г.І. <input type="checkbox"/>5 Овчинникова Ю.Ю. <input type="checkbox"/>2 Остапенко К.Я. <input type="checkbox"/>6 Константинов К.К. <input type="checkbox"/>3 Марченко Л.І. <input type="checkbox"/>7 Новосад Г.І. <input type="checkbox"/>4 Василів К.Я. <input type="checkbox"/>8 Марченко Л.І.</p>
2	<p>Політична партія «Слуга Народу» Переможенцева Євгенія Петрівна</p>		<p><input type="checkbox"/>1 Гончаренко Г.І. <input type="checkbox"/>5 Леонов Ю.Ю. <input type="checkbox"/>2 Шевченко К.Я. <input type="checkbox"/>6 Поліщук К.К. <input type="checkbox"/>3 Димов К.Я. <input type="checkbox"/>7 Хмельницький Г.І. <input type="checkbox"/>4 Наконечний К.Я. <input type="checkbox"/>8 Підпригора Л.І.</p>
3	<p>Політична партія «Опозиційна платформа – За життя» Волноваха Сергій Петрович</p>		<p><input type="checkbox"/>1 Олійник Г.І. <input type="checkbox"/>5 Даліба Ю.Ю. <input type="checkbox"/>2 Панасов К.Я. <input type="checkbox"/>6 Кошляк К.К. <input type="checkbox"/>3 Джегур Л.І. <input type="checkbox"/>7 Балас Г.І. <input type="checkbox"/>4 Тищенко К.Я. <input type="checkbox"/>8 Музальов Л.І.</p>
4	<p>Політична партія Всеукраїнське об'єднання «Батьківщина» Кулеба Олег Миколайович</p>		<p><input type="checkbox"/>1 Беркут Г.І. <input type="checkbox"/>5 Колосенко Ю.Ю. <input type="checkbox"/>2 Петрик К.Я. <input type="checkbox"/>6 Савчук К.К. <input type="checkbox"/>3 Денисенко Л.І. <input type="checkbox"/>7 Ореховський Г.І. <input type="checkbox"/>4 Коробчук К.Я. <input type="checkbox"/>8 Іщенко Л.І.</p>
5	<p>Політична партія «Партія економічного відродження України» Чук Василь Григорович</p>		<p><input type="checkbox"/>1 Сербин Г.І. <input type="checkbox"/>5 Підпалій Ю.Ю. <input type="checkbox"/>2 Гейло К.Я. <input type="checkbox"/>6 Гламазда К.К. <input type="checkbox"/>3 Поляруш Л.І. <input type="checkbox"/>7 Король Г.І. <input type="checkbox"/>4 Мазуревич К.Я. <input type="checkbox"/>8 Гришок Л.І.</p>
6	<p>Політична партія Всеукраїнське об'єднання «Свобода» Сівак Світлана Теодівна</p>		<p><input type="checkbox"/>1 Ланович Г.І. <input type="checkbox"/>5 Титаренко Ю.Ю. <input type="checkbox"/>2 Бабій К.Я. <input type="checkbox"/>6 Слоцьовський К.К. <input type="checkbox"/>3 Станько Л.І. <input type="checkbox"/>7 Дубина Г.І. <input type="checkbox"/>4 Вікарчук К.Я. <input type="checkbox"/>8 Бандусяк І.І.</p>
7	<p>Політична партія «Сила Людей» Солонтай Олександр Гейзович</p>		<p><input type="checkbox"/>1 Новосад Г.І. <input type="checkbox"/>5 Овчинникова Ю.Ю. <input type="checkbox"/>2 Димов К.Я. <input type="checkbox"/>6 Константинов К.К. <input type="checkbox"/>3 Марченко Л.І. <input type="checkbox"/>7 Новосад Г.І. <input type="checkbox"/>4 Василів К.Я. <input type="checkbox"/>8 Марченко Л.І.</p>

Рисунок 3.5 – Результат накладання обробленого вертикально сканованого бюлетеня на шаблон

Аналізуючи рисунок 3.5, видно що позначка виборця, номер округу та виборча дільниця вийшли стали бліді, а інший текст залишився чітким і не

розмитим. Це свідчить про те, що алгоритм відпрацював дуже добре, і всі елементи залишилися на своїх місцях і для подальшої обробки потрібно лише виділити завідомо відомі частини та відправити на розпізнавання, що значно збільшить можливості розпізнавання.

Одним з основних недоліків було «зажовування» бюлетеня. Звісно, якщо в частину, яка «зажувалася» потрапляє важлива інформація – неможливо обробити бюлетень повноцінно та правильно. А основна проблема полягає скоріше у пристроях, що фізично приймають бюлетні. Оптимізувати це на програмному рівні майже неможливо.

Однак, певним видом такої помилки можна вважати повернення документа на кілька градусів (одну зі сторін вставили раніше ніж іншу). Приклад такого бюлетеня зображено на рисунку 3.6.

Місцеві вибори 25 жовтня 2020 року

ВИБОРЧИЙ БЮЛЕТЕНЬ

Багатомандатний виборчий округ з виборів депутатів Київської міської ради

Територіальний виборчий округ №6
Виборча дільниця № 856 395

ПЕРШИЙ КРОК
Зробіть одну позначку "плюс" або іншу, яка засвідчує ваше волевиявлення у лівому квадраті проти назви місцевої організації політичної партії

ДРУГИЙ КРОК
Занесіть в трафарет номер обраного вами кандидата зі списку політичної партії, яку ви обрали (за бажанням)

1	Політична партія «УКРАЇНСЬКЕ ОБ'ЄДНАННЯ ПАТРІОТІВ – УКРОП» Бабенко Володимир Сергійович		<input type="checkbox"/> 1 Дашкевич Г.І. <input type="checkbox"/> 2 Остапенко К.Я. <input type="checkbox"/> 3 Марченко Л.І. <input type="checkbox"/> 4 Василюк К.Я.	<input type="checkbox"/> 5 Овчинников Ю.Ю. <input type="checkbox"/> 6 Константинов К.К. <input type="checkbox"/> 7 Новосад Г.І. <input type="checkbox"/> 8 Марченко Л.І.
2	Політична партія «Слуга Народу» Переможцева Євгенія Петрівна		<input type="checkbox"/> 1 Гончаренко Г.І. <input type="checkbox"/> 2 Шевченко К.Я. <input type="checkbox"/> 3 Димов К.Я. <input type="checkbox"/> 4 Наконечний К.Я.	<input type="checkbox"/> 5 Леонов Ю.Ю. <input type="checkbox"/> 6 Поліщук К.К. <input type="checkbox"/> 7 Хмельницький Г.І. <input type="checkbox"/> 8 Підпригора Л.І.
3	Політична партія «Опозиційна платформа – За життя» Волноваха Сергій Петрович		<input type="checkbox"/> 1 Олійник Г.І. <input type="checkbox"/> 2 Панасов К.Я. <input type="checkbox"/> 3 Джегур Л.І. <input type="checkbox"/> 4 Тищенко К.Я.	<input type="checkbox"/> 5 Далиба Ю.Ю. <input type="checkbox"/> 6 Кошляк К.К. <input type="checkbox"/> 7 Балас Г.І. <input type="checkbox"/> 8 Музальов Л.І.
4	Політична партія Всеукраїнське об'єднання «Батьківщина» Кулеба Олег Миколайович		<input type="checkbox"/> 1 Беркут Г.І. <input type="checkbox"/> 2 Петрик К.Я. <input type="checkbox"/> 3 Денисенко Л.І. <input type="checkbox"/> 4 Коробчук К.Я.	<input type="checkbox"/> 5 Колосенко Ю.Ю. <input type="checkbox"/> 6 Савчук К.К. <input type="checkbox"/> 7 Ореховський Г.І. <input type="checkbox"/> 8 Іщенко Л.І.
5	Політична партія «Партія економічного відродження України» Чук Василь Григорович		<input type="checkbox"/> 1 Сербин Г.І. <input type="checkbox"/> 2 Гейло К.Я. <input type="checkbox"/> 3 Поляруш Л.І. <input type="checkbox"/> 4 Мазуревич К.Я.	<input type="checkbox"/> 5 Підпалій Ю.Ю. <input type="checkbox"/> 6 Глямазда К.К. <input type="checkbox"/> 7 Король Г.І. <input type="checkbox"/> 8 Грисюк Л.І.
6	Політична партія Всеукраїнське об'єднання «Свобода» Сівак Світлана Теодівна		<input type="checkbox"/> 1 Лановий Г.І. <input type="checkbox"/> 2 Бабій К.Я. <input type="checkbox"/> 3 Станько Л.І. <input type="checkbox"/> 4 Вікарчук К.Я.	<input type="checkbox"/> 5 Тітаренко Ю.Ю. <input type="checkbox"/> 6 Слоньовський К.К. <input type="checkbox"/> 7 Дубина Г.І. <input type="checkbox"/> 8 Бандусяк Л.І.
7	Політична партія «Сила Людей» Солонтай Олександр Гейзович		<input type="checkbox"/> 1 Новосад Г.І. <input type="checkbox"/> 2 Димов К.Я. <input type="checkbox"/> 3 Марченко Л.І. <input type="checkbox"/> 4 Василюк К.Я.	<input type="checkbox"/> 5 Овчинников Ю.Ю. <input type="checkbox"/> 6 Константинов К.К. <input type="checkbox"/> 7 Новосад Г.І. <input type="checkbox"/> 8 Марченко Л.І.

Рисунок 3.6 – Зісканований під кутом бюлетень

Застосувавши до даного бюлетеня запропонований алгоритм попередньої обробки і наклавши його на шаблонне зображення бюлетеня, отримаємо зображення показане на рисунку 3.7.

Місцеві вибори 25 жовтня 2020 року

ВИБОРЧИЙ БЮЛЕТЕНЬ

Багатомандатний виборчий округ
з виборів депутатів Київської міської ради

Територіальний виборчий округ №6
Виборча дільниця № 856 395

ПЕРШИЙ КРОК
Зробіть одну позначку "плюс" або іншу, яка засвідчує ваше волевиявлення у лівому квадраті проти назви місцевої організації політичної партії

ДРУГИЙ КРОК
Занесіть в трафарет номер обраного вами кандидата зі списку політичної партії, яку ви обрали (за бажанням)


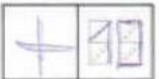





1	<p><small>Політична партія</small> «УКРАЇНСЬКЕ ОБ'ЄДНАННЯ ПАТРІОТІВ – УКРОП» Бабенко Володимир Сергійович</p>		<p><input type="checkbox"/>1 Дашкевич Г.І. <input type="checkbox"/>2 Остапенко К.Я. <input type="checkbox"/>3 Марченко Л.І. <input type="checkbox"/>4 Василів К.Я.</p>	<p><input type="checkbox"/>5 Свчинникова Ю.Ю. <input type="checkbox"/>6 Константинов К.К. <input type="checkbox"/>7 Новосад Г.І. <input type="checkbox"/>8 Марченко Л.І.</p>
2	<p><small>Політична партія</small> «Слуга Народу» Переможцева Євгенія Петрівна</p>		<p><input type="checkbox"/>1 Гончаренко Г.І. <input type="checkbox"/>2 Шевченко К.Я. <input type="checkbox"/>3 Димов К.Я. <input type="checkbox"/>4 Наконечний К.Я.</p>	<p><input type="checkbox"/>5 Леснов Ю.Ю. <input type="checkbox"/>6 Поліщук К.К. <input type="checkbox"/>7 Хмельницький Г.І. <input type="checkbox"/>8 Підопригара Л.І.</p>
3	<p><small>Політична партія</small> «Опозиційна платформа – За життя» Волноваха Сергій Петрович</p>		<p><input type="checkbox"/>1 Олійник Г.І. <input type="checkbox"/>2 Панасов К.Я. <input type="checkbox"/>3 Джигур Л.І. <input type="checkbox"/>4 Тищенко К.Я.</p>	<p><input type="checkbox"/>5 Далиба Ю.Ю. <input type="checkbox"/>6 Кошляк К.К. <input type="checkbox"/>7 Балас Г.І. <input type="checkbox"/>8 Музалов Л.І.</p>
4	<p><small>Політична партія</small> Всеукраїнське об'єднання «Батьківщина» Кулєба Олег Миколайович</p>		<p><input type="checkbox"/>1 Беркут Г.І. <input type="checkbox"/>2 Петрик К.Я. <input type="checkbox"/>3 Денисенко Л.І. <input type="checkbox"/>4 Коробчук К.Я.</p>	<p><input type="checkbox"/>5 Колошенко Ю.Ю. <input type="checkbox"/>6 Савчук К.К. <input type="checkbox"/>7 Орезовський Г.І. <input type="checkbox"/>8 Іщенко Л.І.</p>
5	<p><small>Політична партія</small> «Партія економічного відродження України» Чук Василь Григорович</p>		<p><input type="checkbox"/>1 Сербик Г.І. <input type="checkbox"/>2 Рейле К.Я. <input type="checkbox"/>3 Поляруш Л.І. <input type="checkbox"/>4 Мазуревич К.Я.</p>	<p><input type="checkbox"/>5 Підпалый Ю.Ю. <input type="checkbox"/>6 Станізде К.К. <input type="checkbox"/>7 Король Г.І. <input type="checkbox"/>8 Фрисюк Л.І.</p>
6	<p><small>Політична партія</small> Всеукраїнське об'єднання «Свобода» Сівак Світлана Теодівна</p>		<p><input type="checkbox"/>1 Ланович Г.І. <input type="checkbox"/>2 Бабій К.Я. <input type="checkbox"/>3 Станько Л.І. <input type="checkbox"/>4 Віварчук К.Я.</p>	<p><input type="checkbox"/>5 Титаренко Ю.Ю. <input type="checkbox"/>6 Слоновський К.К. <input type="checkbox"/>7 Дубина Г.І. <input type="checkbox"/>8 Бандусяк Л.І.</p>
7	<p><small>Політична партія</small> «Сила Людей» Солонтей Олександр Гейзович</p>		<p><input type="checkbox"/>1 Новосад Г.І. <input type="checkbox"/>2 Димов К.Я. <input type="checkbox"/>3 Марченко Л.І. <input type="checkbox"/>4 Василів К.Я.</p>	<p><input type="checkbox"/>5 Свчинникова Ю.Ю. <input type="checkbox"/>6 Константинов К.К. <input type="checkbox"/>7 Новосад Г.І. <input type="checkbox"/>8 Марченко Л.І.</p>

Рисунок 3.7 – Результат накладання обробленого сканованого під кутом бюлетеня на шаблон

Аналізуючи результат накладання, бачимо, що результат вийшов дещо розмитий в нижній частині, що свідчить про погіршення якості співставлення,

однак суттєвої різниці не спостерігається, а отже частини для подальшого розпізнавання розширяти не доведеться.

Дуже схожий результат система дає на бюлетенях поданих «вверх ногами». А саме: при поверненні бюлетеня на 90, 180, 270 градусів з відхиленнями в кілька градусів система дає дуже схожий результат до наведеного вище. Що також свідчить про виправлення ще однієї проблеми аналогів.

Останньою частиною тесту буде також перевірка додаткових можливостей системи. А саме обробки не сканованого зображення, а зробленого на телефон. Приклад такого зображення показано на рисунку 3.8.



Рисунок 3.8 – Зображення бюлетеня зроблене за допомогою телефона

Результат накладання обробленого зображення на шаблон показано на рисунку 3.9.

Місцеві вибори 25 жовтня 2020 року

ВИБОРЧИЙ БЮЛЕТЕНЬ

Багатомандатний виборчий округ
з виборів депутатів Київської міської ради

Територіальний виборчий округ №6
Виборча дільниця № 856.395

ПЕРШИЙ КРОК

Зробіть одну позначку "плюс" або іншу, яка засвідчує ваше волевиявлення у лівому квадраті проти назви місцевої організації політичної партії

ДРУГИЙ КРОК

Занесіть в трафарет номер обраного вами кандидата зі списку політичної партії, яку ви обрали (за бажанням)

1	<p><small>Політична партія</small> «УКРАЇНСЬКЕ ОБ'ЄДНАННЯ ПАТРІОТІВ – УКРОП» Бабенко Володимир Сергійович</p>		<p>01 Дашкевич Г.І. 02 Остапенко К.Я. 03 Марченко Л.І. 04 Василюк К.Я.</p>	<p>05 Овчинникова Ю.Ю. 06 Константинов К.К. 07 Новосад Г.І. 08 Марченко Л.І.</p>
2	<p><small>Політична партія</small> «Слуга Народу» Переможцева Євгенія Петрівна</p>		<p>01 Гончаренко Г.І. 02 Шевченко К.Я. 03 Димов К.Я. 04 Наконечний К.Я.</p>	<p>05 Леонов Ю.Ю. 06 Поліщук К.К. 07 Хмельницький Г.І. 08 Підпригора Л.І.</p>
3	<p><small>Політична партія</small> «Опозиційна платформа – За життя» Волноваха Сергій Петрович</p>		<p>01 Олійник Г.І. 02 Панасов К.Я. 03 Джедур Л.І. 04 Тищенко К.Я.</p>	<p>05 Далиба Ю.Ю. 06 Кошляк К.К. 07 Балаєв Г.І. 08 Музальов Л.І.</p>
4	<p><small>Політична партія</small> Всеукраїнське об'єднання «Батьківщина» Кулеба Олег Миколайович</p>		<p>01 Беркут Г.І. 02 Петрик К.Я. 03 Денисенко Л.І. 04 Коробчук К.Я.</p>	<p>05 Колошенко Ю.Ю. 06 Савчук К.К. 07 Ореховський Г.І. 08 Іщенко Л.І.</p>
5	<p><small>Політична партія</small> «Партія економічного відродження України» Чук Василь Григорович</p>		<p>01 Сербин Г.І. 02 Гейло К.Я. 03 Поляруш Л.І. 04 Мазуревич К.Я.</p>	<p>05 Підплатий Ю.Ю. 06 Глазнавада К.К. 07 Король Г.І. 08 Грисюк Л.І.</p>
6	<p><small>Політична партія</small> Всеукраїнське об'єднання «Свобода» Сівак Світлана Теодівна</p>		<p>01 Ланович Г.І. 02 Бабій К.Я. 03 Станько Л.І. 04 Вікарчук К.Я.</p>	<p>05 Титаренко Ю.Ю. 06 Слоновський К.К. 07 Дубина Г.І. 08 Бандушок Л.І.</p>
7	<p><small>Політична партія</small> «Сила Людей» Солонтей Олександр Гейзович</p>		<p>01 Новосад Г.І. 02 Димов К.Я. 03 Марченко Л.І. 04 Василюк К.Я.</p>	<p>05 Овчинникова Ю.Ю. 06 Константинов К.К. 07 Новосад Г.І. 08 Марченко Л.І.</p>

Рисунок 3.9 – Результат накладання обробленого зображення бюлетеня зробленого за допомогою телефона на шаблон

Аналізуючи отриманий результат можна так само зробити висновок про високу якість обробки так як розмиття виникли лише на невеликій частині фото, але враховуючи складність освітлення та не найкращу якість самого фото результат можна вважати дуже хорошим.

3.6 Висновок

В цьому розділі було обґрунтовано вибір Python як основної мови програмування для реалізації даної технології. Також було обґрунтовано вибір PyChart як основного середовища для розробки системи. Окрім цього, було здійснено порівняння найкращих систем OCR та визначення тієї, що найкраще задовольняє поставлені цілі. На базі обраних елементів було розроблено інформаційну технологію підрахунку голосів на виборах, шляхом тестування перевірено її можливості та відповідність поставленим вимогам.

4 ЕКОНОМІЧНА ЧАСТИНА

4.1 Оцінювання комерційного потенціалу розробки

Метою проведення технологічного аудиту є оцінювання комерційного потенціалу розробки [35]. Для проведення технологічного аудиту було залучено 2-х незалежних експертів. Такими експертами будуть Яровий А.А. та Бальзан М.В.

Здійснюємо оцінювання комерційного потенціалу даної розробки за 12-ма критеріями за 5-ти бальною шкалою.

Оцінювання комерційного потенціалу розробки та його результати наведено в таблиці 4.1.

Таблиця 4.1 – Результати оцінювання комерційного потенціалу розробки

Критерії	Прізвище, ініціали, посада експерта	
	1. Яровий А.А.	2. Бальзан М.В.
	Бали, виставлені експертами:	
1	3	4
2	4	3
3	5	5
4	4	4
5	3	3
6	3	3
7	4	4
8	5	4
9	4	4
10	3	4
11	3	3
12	4	4
Сума балів	СБ ₁ = 45	СБ ₂ = 45
Середньоарифметична сума балів $\overline{СБ}$	$\overline{СБ} = \frac{\sum_1^2 СБ_i}{2} = 45$	

Отже, з отриманих даних таблиці 4.1 видно, що нова розробка має високий рівень комерційного потенціалу.

4.2 Прогнозування витрат на виконання науково-дослідної роботи та конструкторсько–технологічної роботи

Для розробки нового програмного продукту необхідні такі витрати.

Основна заробітна плата для розробників визначається за формулою (4.1):

$$Z_o = \frac{M}{T_p} \cdot t \quad (4.1)$$

де M - місячний посадовий оклад конкретного розробника;

T_p - кількість робочих днів у місяці, $T_p = 21$ дні;

t - число днів роботи розробника, $t = 60$ днів.

Розрахунки заробітних плат для керівника і програміста наведені в таблиці 4.2.

Таблиця 4.2 – Розрахунки основної заробітної плати

Працівник	Оклад M , грн.	Оплата за робочий день, грн.	Число днів роботи, t	Витрати на оплату праці, грн.
Науковий керівник	42000	2000	5	10000
Інженер-програміст	31500	1500	55	82500
Всього:				92500

Розрахуємо додаткову заробітну плату:

$$Z_{\text{дод}} = 0,1 \cdot 92500 = 9250 \text{ (грн.)}$$

Нарахування на заробітну плату операторів НЗП розраховується як 22% від суми їхньої основної та додаткової заробітної плати:

$$H_{зп} = (Z_o + Z_p) \cdot \frac{\beta}{100}, \quad (4.2)$$

$$H_{зп} = (92500 + 9250) \cdot \frac{22}{100} = 22385 \text{ (грн.)}.$$

Розрахунок амортизаційних витрат для програмного забезпечення виконується за такою формулою:

$$A = \frac{Ц \cdot H_a}{100} \cdot \frac{T}{12}, \quad (4.3)$$

де Ц – балансова вартість обладнання, грн;

H_a – річна норма амортизаційних відрахувань % (для програмного забезпечення 25%);

T – Термін використання (T=3 міс.).

Таблиця 4.3 – Розрахунок амортизаційних відрахувань

Найменування програмного забезпечення	Балансова вартість, грн.	Норма амортизації, %	Термін використання, міс.	Величина амортизаційних відрахувань, грн
Персональний комп'ютер	27000	25	3	1687,5
Всього:				1687,5

Розрахуємо витрати на комплектуючі. Витрати на комплектуючі розрахуємо за формулою:

$$K = \sum_1^n H_i \cdot C_i \cdot K_i, \quad (4.4)$$

де n – кількість комплектуючих;

H_i – кількість комплектуючих i -го виду;

C_i – покупна ціна комплектуючих i -го виду, грн;

K_i – коефіцієнт транспортних витрат (прийmemo $K_i = 1,1$).

Таблиця 4.4 - Витрати на комплектуючі, що були використані для розробки ПЗ.

Найменування матеріалу	Одиниці виміру	Ціна, грн.	Витрачено	Вартість витрачених матеріалів, грн.
Флешка	шт.	180	1	180
Пачка паперу	уп.	75	1	75
Ручка	шт.	6,82	1	6,82
Всього з урахуванням транспортних витрат				288

Витрати на силову електроенергію розраховуються за формулою:

$$V_e = V \cdot P \cdot \Phi \cdot K_{\Pi} \quad (4.5)$$

де V – вартість 1кВт-години електроенергії ($V=2$ грн/кВт);

P – установлена потужність комп'ютера ($P=0,6$ кВт);

Φ – фактична кількість годин роботи комп'ютера ($\Phi=200$ год.);

K_{Π} – коефіцієнт використання потужності ($K_{\Pi} < 1$, $K_{\Pi} = 0,7$).

$$B_e = 2 \cdot 0,6 \cdot 200 \cdot 0,7 = 168 \text{ (грн.)}$$

Розрахуємо інші витрати $B_{ін}$.

Інші витрати I_B можна прийняти як (100...300)% від суми основної заробітної плати розробників та робітників, які виконували дану роботу, тобто:

$$B_{ін} = (1..3) \cdot (Z_o + Z_p). \quad (4.6)$$

$$B_{ін} = 1 \cdot (92500 + 9250) = 101750 \text{ (грн.)}$$

Сума всіх попередніх статей витрат дає витрати на виконання даної частини роботи:

$$B = Z_o + Z_{дод} + H_{зп} + A + K + B_e + I_B \quad (4.7)$$

$$\begin{aligned} B &= 92500 + 9250 + 22385 + 1687,5 + 288 + 168 + 101750 \\ &= 228028,5 \text{ (грн)} \end{aligned}$$

Розрахуємо загальну вартість наукової роботи $B_{заг}$ за формулою:

$$B_{заг} = \frac{B}{\alpha}, \quad (4.8)$$

де α – частка витрат, які безпосередньо здійснює виконавець даного етапу роботи, у відн. одиницях = 1.

$$B_{заг} = \frac{228028,5}{1} = 228028,5 \text{ (грн)}$$

Прогнозування загальних витрат ЗВ на виконання та впровадження результатів виконаної наукової роботи здійснюється за формулою:

$$ЗВ = \frac{В_{заг}}{\beta}, \quad (4.9)$$

де β – коефіцієнт, який характеризує етап (стадію) виконання даної роботи.

Отже, розрахуємо загальні витрати:

$$ЗВ = \frac{228028,5}{0,9} = 253365 \text{ (грн).}$$

4.3 Прогнозування комерційних ефектів від реалізації результатів розробки

Спрогнозуємо отримання прибутку від реалізації результатів нашої розробки. Зростання чистого прибутку можна оцінити у теперішній вартості грошей. Це забезпечить підприємству (організації) надходження додаткових коштів, які дозволять покращити фінансові результати діяльності .

Оцінка зростання чистого прибутку підприємства від впровадження результатів наукової розробки. У цьому випадку збільшення чистого прибутку підприємства $\Delta \Pi_i$ для кожного із років, протягом яких очікується отримання позитивних результатів від впровадження розробки, розраховується за формулою:

$$\Delta \Pi_i = \sum_1^n (\Delta \Pi_{я} \cdot N + \Pi_{я} \Delta N)_i \quad (4.10)$$

де $\Delta \Pi_{\text{я}}$ – покращення основного якісного показника від впровадження результатів розробки у даному році;

N – основний кількісний показник, який визначає діяльність підприємства у даному році до впровадження результатів наукової розробки;

ΔN – покращення основного кількісного показника діяльності підприємства від впровадження результатів розробки;

$\Pi_{\text{я}}$ – основний якісний показник, який визначає діяльність підприємства у даному році після впровадження результатів наукової розробки;

n – кількість років, протягом яких очікується отримання позитивних результатів від впровадження розробки.

В результаті впровадження результатів наукової розробки витрати на виготовлення інформаційної технології зменшаться на 5000 грн (що автоматично спричинить збільшення чистого прибутку підприємства на 5000 грн), а кількість користувачів, які будуть користуватись збільшиться: протягом першого року – на 50 користувачів, протягом другого року – на 25 користувачів, протягом третього року – 10 користувачів. Реалізація інформаційної технології до впровадження результатів наукової розробки складала 1 користувачів, а прибуток, що отримував розробник до впровадження результатів наукової розробки – 4000 грн.

Спрогнозуємо збільшення чистого прибутку від впровадження результатів наукової розробки у кожному році відносно базового.

Отже, збільшення чистого продукту $\Delta \Pi_1$ протягом першого року складатиме:

$$\Delta \Pi_1 = 5000 \cdot 1 + (4000 + 5000) \cdot 50 = 455000 \text{ грн.}$$

Протягом другого року:

$$\Delta \Pi_2 = 5000 \cdot 1 + (4000 + 5000) \cdot (50 + 25) = 680000 \text{ грн.}$$

Протягом третього року:

$$\Delta\Pi_3 = 5000 \cdot 1 + (4000 + 5000) \cdot (50 + 25 + 10) = 770000 \text{ грн.}$$

4.4 Розрахунок ефективності вкладених інвестицій та період їх окупності

Визначимо абсолютну і відносну ефективність вкладених інвестором інвестицій та розрахуємо термін окупності.

Абсолютна ефективність $E_{\text{абс}}$ вкладених інвестицій розраховується за формулою:

$$E_{\text{абс}} = (\text{ПП} - PV) \quad (4.11)$$

де ПП – приведена вартість всіх чистих прибутків, що їх отримає підприємство (організація) від реалізації результатів наукової розробки, грн;

PV – теперішня вартість інвестицій $PV = 3B$, грн.

Рисунок, що характеризує рух платежів (інвестицій та додаткових прибутків) буде мати вигляд, рисунок 4.1.

Розрахуємо вартість чистих прибутків за формулою:

$$\text{ПП} = \sum_1^m \frac{\Delta\Pi_i}{(1 + \tau)^t} \quad (4.12)$$

де $\Delta\Pi_i$ – збільшення чистого прибутку у кожному із років, протягом яких виявляються результати виконаної та впровадженої НДДКР, грн;

t – період часу, протягом якого виявляються результати впровадженої НДДКР, роки;

τ – ставка дисконтування, за яку можна взяти щорічний прогнозований рівень інфляції в країні; для України цей показник знаходиться на рівні 0,1;

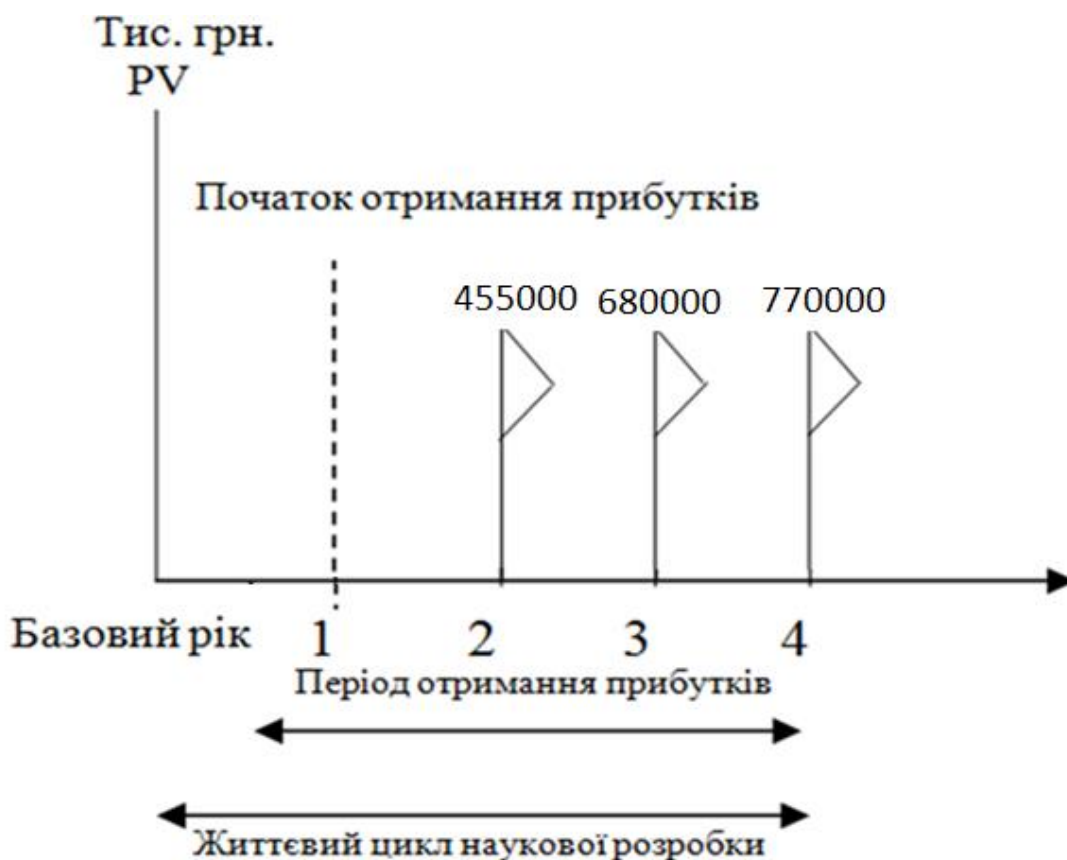


Рисунок 4.1 – Вісь часу з фіксацією платежів, що мають місце під час розробки та впровадження результатів НДДКР

t – період часу (в роках) від моменту отримання чистого прибутку до точки.

Отже, розрахуємо вартість чистого прибутку:

$$ПП = \frac{455000}{(1 + 0,1)^2} + \frac{680000}{(1 + 0,1)^3} + \frac{770000}{(1 + 0,1)^4} = 1412847,48(\text{грн})$$

Тоді розрахуємо $E_{\text{абс}}$:

$$E_{\text{абс}} = 1412847,48 - 253365 = 1159482,48 \text{ грн.}$$

Оскільки $E_{abc} > 0$, то вкладання коштів на виконання та впровадження результатів НДДКР буде доцільним.

Розрахуємо відносну (щорічну) ефективність вкладених в наукову розробку інвестицій E_B за формулою:

$$E_B = \sqrt[T]{1 + \frac{E_{abc}}{PV}} - 1 \quad (4.13)$$

де E_{abc} – абсолютна ефективність вкладених інвестицій, грн;

PV – теперішня вартість інвестицій $PV = 3B$, грн;

T – життєвий цикл наукової розробки, роки.

Тоді будемо мати:

$$E_B = \sqrt[4]{1 + \frac{1159482,48}{253365}} - 1 = 0,54 \text{ або } 54 \%$$

Далі, розраховану величина E_B порівнюємо з мінімальною (бар'єрною) ставкою дисконтування τ_{\min} , яка визначає ту мінімальну дохідність, нижче за яку інвестиції вкладатися не будуть. У загальному вигляді мінімальна (бар'єрна) ставка дисконтування τ_{\min} визначається за формулою:

$$\tau = d + f \quad (4.14)$$

де d – середньозважена ставка за депозитними операціями в комерційних банках; в 2020 році в Україні $d = 0,1$;

f – показник, що характеризує ризикованість вкладень, величина $f = 0,1$.

$$\tau = 0,1 + 0,1 = 0,2$$

Оскільки $E_B = 54\% > \tau_{\min} = 0,2 = 20\%$, то інвестор буде зацікавлений вкладати гроші в дану наукову розробку.

Термін окупності вкладених у реалізацію наукового проекту інвестицій. Термін окупності вкладених у реалізацію наукового проекту інвестицій $T_{ок}$ розраховується за формулою:

$$T_{ок} = \frac{1}{E_B}$$

$$T_{ок} = \frac{1}{0,54} = 1,85 \text{ роки}$$

Обрахувавши термін окупності даної наукової розробки, можна зробити висновок, що фінансування даної наукової розробки буде доцільним.

4.5 Висновок

В даному розділі було здійснено оцінювання комерційного потенціалу розробки інформаційної технології підрахунку голосів на виборах.

Проведено технологічний аудит з залученням двох експертів. Аналіз експертних даних показав, що рівень комерційного потенціалу розробки високий. Дослідження комерційного потенціалу розробки підтвердило, що програмний продукт за своїми характеристиками випереджає аналогічні програмні продукти і є перспективною розробкою. Він має кращі функціональні показники, а тому є конкурентоспроможним товаром на ринку.

Згідно із розрахунками всіх статей витрат на виконання науково-дослідної, дослідно-конструкторської та конструкторсько-технологічної роботи загальна вартість витрат на розробку і впровадження складає 253365 грн.

Розрахована абсолютна ефективність вкладених інвестицій в сумі 1159482,48 грн свідчить про отримання прибутку інвестором від впровадження програмного продукту у діяльність підприємства.

Щорічна ефективність вкладених в наукову розробку інвестицій складає 54%, що вище за мінімальну бар'єрну ставку дисконтування, яка складає 20%. Це означає потенційну зацікавленість інвесторів у фінансуванні розробки.

Усе це, узятє разом, забезпечує прийняття рішення про доцільність виготовлення нового продукту.

ВИСНОВКИ

У даній магістерській кваліфікаційній роботі було проведено аналіз предметної області електронного голосування, обґрунтовано вибір типу системи в залежності від поставлених вимог. Окрім цього було здійснено порівняльний аналіз існуючих систем, виявлено їх недоліки та визначено, додаткові вимоги до інформаційної технології

Далі було: визначено місце у виборчому процесі для застосування технології; визначено основні елементи системи, обґрунтовано спосіб розпізнавання та його покращення. Розроблено структуру інформаційної технології підрахунку голосів на виборах.

Було обґрунтовано вибір Python як основної мови програмування для реалізації розроблюваної технології. На базі обраних елементів було розроблено систему електронного голосування.

Під час тестування розробленого програмного забезпечення було перевірено його працездатність, відповідність поставленим вимогам, а також показано розширення функціоналу.

Здійснено оцінювання комерційного потенціалу розробки інформаційної технології електронного голосування та підтверджено доцільність даної розробки. Розрахована абсолютна ефективність вкладених інвестицій в сумі 1159482,48 грн свідчить про отримання прибутку інвестором від впровадження програмного продукту у діяльність підприємства. Щорічна ефективність вкладених в наукову розробку інвестицій складає 54%. Це означає потенційну зацікавленість інвесторів у фінансуванні розробки.

В результаті виконання даної магістерської кваліфікаційної роботи поставлені задачі були виконані в повній мірі, мета роботи досягнута.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Питання впровадження електронної системи голосування [Електрон. ресурс] // Державне будівництво та місцеве самоврядування в Україні. – Режим доступу: <http://proeto.com.ua/pravo/derzhavne-budivnictvo-i-miscevesamovryaduvannya-v-ukrayini/pitannyaavprovadzhennya>
2. Про приєднання України до Статуту Ради Європи: Закон України від 31 жовтня 1995 р. № 398/98-ВР // Відомості Верховної Ради України. – 1995. – № 38. – Ст. 287.
3. Статуту Ради Європи від 5 травня 1949 р. // Офіційний вісник України. – 2004. – № 26 – Ст. 215.
4. Про інформацію: Закон України від 2 жовтня 1992 р. № 2657-ХІІ // Відомості Верховної ради України. – 1992. -- № 48. – Ст. 651
5. Спажев, О. М., «Порівняльний аналіз найпопулярніших мов програмування в області штучного інтелекту, машинного навчання та глибинного навчання» в Матеріали конференції «XLIX Науково-технічна конференція підрозділів Вінницького національного технічного університету (2020)», Вінниця, 2020. [Електронний ресурс].
Режим доступу:
<https://conferences.vntu.edu.ua/index.php/allvntu/index/pages/view/zbirn20>
Дата звернення: Черв. 2020
6. Панцир С., Когут А. Е-демократія в Україні: рекомендації щодо впровадження політики та забезпечення її результативності: довідник. Київ, 2018. 20 с.
7. Впровадження електронного голосування. Основні аспекти / пер. з англ. В. Ардицка. Ресурси International IDEA з виробничих процесів, 2018, 39 с.
8. Современные избирательные системы. Вып. 2: Аргентина, Германия, Швеция /А.Г. Орлов, Ю.И. Лейбо, И.А. Ракитская; науч. ред. Ю.А.

- Веденеев, В.И. Лысенко; Центральная избират. комиссия РФ. – М.: РЦОИТ; Норма, 2012. – 320 с
9. Система голосования с оптическим сканирование – Optical scan voting system [Электронный ресурс] – Назва з екрану – https://ru.qwe.wiki/wiki/Optical_scan_voting_system
 - 10.Виборчий кодекс України від 19 вересня 2020 р. № 396-IX // Відомості Верховної Ради України. – 2020. – № 7, № 8, № 9. – Ст. 48.
 - 11.Unified Modeling Language – Вікіпедія [Електронний ресурс]. - Режим доступу: URL: [https://uk.wikipedia.org/wiki/ Unified_Modeling_Language](https://uk.wikipedia.org/wiki/Unified_Modeling_Language). - Назва з екрану.
 - 12.Фаулер М., Скотт К. UML. Основи. — Пер. с англ. — СПб: Символ-Плюс, 2002. — 192 с., ил. ISBN 5-93286-032-4
 - 13.Л. Кочубей. Бюлетень виборчий // Політична енциклопедія. Редкол.: Ю. Левенець (голова), Ю. Шаповал (заст. голови) та ін. — К.: Парламентське видавництво, 2011. — 808 с.
 - 14.Schantz, Herbert F. (1982). The history of OCR, optical character recognition. [Manchester Center, Vt.]: Recognition Technologies Users Association. ISBN 9780943072012.
 - 15.Suen, C.Y.; Plamondon, R.; Tappert, A.; Thomassen, A.; Ward, J.R.; Yamamoto, K. (May 29, 1987). Future Challenges in Handwriting and Computer Applications. 3rd International Symposium on Handwriting and Computer Applications, Montreal, May 29, 1987. Retrieved October 3, 2008.
 - 16.ABBYY Fine Reader 15 – Назва з екрану [Електронний ресурс]. Режим доступу: <https://www.abbyy.com/uk/finereader/>
 - 17.Tesseract OCR – Назва з екрану [Електронний ресурс]. Режим доступу: <https://github.com/tesseract-ocr/tesseract>
 - 18.Amazon Textract – Назва з екрану [Електронний ресурс]. Режим доступу: <https://aws.amazon.com/ru/textract/>
 - 19.Cloud Vision API– Назва з екрану [Електронний ресурс]. Режим доступу: <https://cloud.google.com/vision/docs/ocr>

20. Stefan Leutenegger, Margarita Chli, Roland Siegwart: "BRISK: Binary Robust Invariant Scalable Keypoints". Computer Vision (ICCV), pp. 2548 – 2555, 2011.
21. Rosten, Edward, Tom Drummond: "Machine learning for high-speed corner detection", 9th European Conference on Computer Vision (ECCV), pp. 430 – 443, 2006
22. Michael Calonder, Vincent Lepetit, Christoph Strecha, Pascal Fua, "BRIEF: Binary Robust Independent Elementary Features", 11th European Conference on Computer Vision (ECCV), pp. 778 – 792, 2010.
23. X. Yang, K. T. Cheng: "LDB: An ultra-fast feature for scalable augmented reality". In IEEE and ACM Intl. Sym. on Mixed and Augmented Reality (ISMAR), pp. 49 – 57, 2012.
24. Lowe, David G.: "Object recognition from local scale-invariant features". Proceedings of the International Conference on Computer Vision, pp. 1150 – 1157, 1999
25. Ethan Rublee, Vincent Rabaud, Kurt Konolige, Gary Bradski: "ORB: an efficient alternative to SIFT or SURF", Computer Vision (ICCV), IEEE International Conference on. IEEE, pp. 2564 – 2571, 2011.
26. S. A. K. Tareen and Z. Saleem, "A comparative analysis of SIFT, SURF, KAZE, AKAZE, ORB, and BRISK," 2018 International Conference on Computing, Mathematics and Engineering Technologies (iCoMET), Sukkur, 2018, pp. 1-10, doi: 10.1109/ICOMET.2018.8346440.
27. Лутц М. – Изучаем Python, том 1, 5-е изд.: Пер. с англ. — СПб.: ООО «Диалектика», 2019. — 832 с.: ил. — Парад, тит. англ.
28. Страуструп Б. Язык программирования C++. Специальное издание, – Перев. с англ / Б. Страуструп – М.: Бином, 2011. - 1210 с. ISBN: 978-5-7989-0223
29. Эккель Б. Философия Java. 4-е полное изд. – СПб.: Питер, 2015р. – 1168 с.: ил. – (Серия «Классика computer science»). ISBN 978-5-496-01127-3

30. PyCharm – Назва з екрану [Електронний ресурс]. Режим доступу:
<https://www.jetbrains.com/ru-ru/pycharm/>
31. OpenCV – Назва з екрану [Електронний ресурс]. Режим доступу:
<https://opencv.org/>
32. Django – Назва з екрану [Електронний ресурс]. Режим доступу:
<https://www.djangoproject.com/>
33. PostgreSQL – Назва з екрану [Електронний ресурс]. Режим доступу:
<https://www.postgresql.org/>
34. RabbitMQ – Назва з екрану [Електронний ресурс]. Режим доступу:
<https://www.rabbitmq.com/>
35. Методичні вказівки до виконання студентами-магістрантами наукового напрямку економічної частини магістерських кваліфікаційних робіт / Уклад. В. О. Козловський – Вінниця: ВНТУ, 2012. – 22 с.