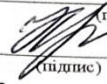


МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ УКРАЇНИ
КРИВОРІЗЬКИЙ ФАХОВИЙ КОЛЕДЖ
ДЕРЖАВНОГО НЕКОМЕРЦІЙНОГО ПІДПРИЄМСТВА
«ДЕРЖАВНИЙ УНІВЕРСИТЕТ «КИЇВСЬКИЙ АВІАЦІЙНИЙ ІНСТИТУТ»
Циклова комісія комп'ютерних систем та мереж
(повна назва циклової комісії)

Допустити до захисту
Голова випускової циклової комісії
комп'ютерних систем та мереж


(повна назва циклової комісії)
Ірина КРАВЧУК
(ім'я, ПРІЗВИЩЕ)

« 10 » 06 2025 р.

КВАЛІФІКАЦІЙНА РОБОТА
(ПОЯСНОВАЛЬНА ЗАПИСКА)
ВИПУСКНИКА ОСВІТНЬО-ПРОФЕСІЙНОГО СТУПЕНЯ
ФАХОВИЙ МОЛОДШИЙ БАКАЛАВР

Тема: Проектування стенду для перевірки трекінгостійкості

Група: 3-011 Спеціальність: 123 «Комп'ютерна інженерія»

Здобувач освіти


(підпис)

Євген АНТОНЮК

(ім'я, ПРІЗВИЩЕ)

Керівник роботи


(підпис)

Артем КУТІН

(ім'я, ПРІЗВИЩЕ)

Консультант з оформлення
пояснювальної записки


(підпис)

Оксана ОСАДЧА

(ім'я, ПРІЗВИЩЕ)

Кривий Ріг 2025 р.

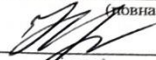
КРИВОРІЗЬКИЙ ФАХОВИЙ КОЛЕДЖ
ДЕРЖАВНОГО НЕКОМЕРЦІЙНОГО ПІДПРИЄМСТВА
«ДЕРЖАВНИЙ УНІВЕРСИТЕТ «КИЇВСЬКИЙ АВІАЦІЙНИЙ ІНСТИТУТ»

Відділення комп'ютерної та програмної інженерії
Циклова комісія комп'ютерних систем та мереж
Освітньо-професійний ступінь фаховий молодший бакалавр
Спеціальність 123 «Комп'ютерна інженерія»

ЗАТВЕРДЖУЮ

Голова випускової циклової комісії
комп'ютерних систем та мереж

(повна назва циклової комісії)


Ірина КРАВЧУК
(ім'я, ПРІЗВИЩЕ)

« 01 » 03 2025 р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ ЗДОБУВАЧУ ОСВІТИ

Антонюку Євгену Олеговичу

(прізвище, ім'я, по батькові)

1. Тема роботи Проектування стенду для перевірки трекінгостійкості

Керівник роботи Кутін Артем Іллі, викладач

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по коледжу від « 04 » 04 2025 року № 50-ст

2. Строк подання здобувачем освіти роботи з _____ по _____

3. Вихідні дані до роботи Аналіз зразків ізоляційних матеріалів на трекінгостійкість, стандарту IEC 60112, що регламентує параметри напруги, тип електродів і склад електроліту

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)

Аналіз проблеми та сучасних підходів

Проектування апаратної частини експериментального стенду

Розробка програмного забезпечення для аналізу трекінгостійкості

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)

Презентація Microsoft PowerPoint

6. Консультанти розділів роботи (проекту)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

7. Дата видачі завдання _____

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	<i>Узгодження технічного завдання з керівником кваліфікаційної роботи</i>	<i>04.04.2025-07.04.2025</i>	<i>виконано</i>
2	<i>Підбір та вивчення науково-технічної літератури за темою кваліфікаційної роботи</i>	<i>08.04.2025-14.04.2025</i>	<i>виконано</i>
3	<i>Аналіз проблеми та сучасних підходів</i>	<i>15.04.2025-21.04.2025</i>	<i>виконано</i>
4	<i>Проектування апаратної частини експериментального стенду</i>	<i>22.04.2025-28.04.2025</i>	<i>виконано</i>
5	<i>Розробка програмного забезпечення для аналізу трекінгостійкості</i>	<i>29.04.2025-23.05.2025</i>	<i>виконано</i>
6	<i>Написання та оформлення пояснювальної записки</i>	<i>26.05.2025-30.05.2025</i>	<i>виконано</i>
7	<i>Попередній захист кваліфікаційної роботи</i>	<i>09.06.2025-12.06.2025</i>	<i>виконано</i>
8	<i>Захист кваліфікаційної роботи</i>		

Здобувач освіти


(підпис)

Євген АНТОНЮК

(ім'я, ПРІЗВИЩЕ)

Керівник роботи


(підпис)

Артем КУТІН

(ім'я, ПРІЗВИЩЕ)

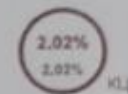
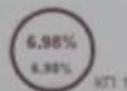
Звіт подібності

Метадані

Назва організації
 Ukrainian national aviation university
 Заголовок
 АНТОНЮК Євген Олегович 3-011
 Автор Науковий керівник / Експерт
 АНТОНЮККулін А.
 Назва файлу
 Криворізький Фаховий коледж

Обсяг знайдених подібностей

Коефіцієнт подібності вказує, який відсоток тексту по відношенню до загального обсягу тексту було знайдено в різних джерелах. Зверніть увагу, що високі значення коефіцієнта не автоматично означають plagiat. Звіт має аналізувати компетентна / уповноважена особа.



25

Довжина фраз для коефіцієнта подібності 2

7732

Кількість слів

62582

Кількість символів

Тривога

У цьому розділі ви знайдете інформацію щодо текстових спотворень. Ці спотворення в тексті можуть говорити про МОЖЛИВІ маніпуляції в тексті. Спотворення в тексті можуть мати навислий характер, але частіше характер технічних помилок при конвертації документа та його збереженні, тому ми рекомендуємо вам підходити до аналізу цього модуля відповідально. У разі виникнення запитань, просимо звертатися до нашої служби підтримки.

Заміна букв		0
Інтервали		0
Мікропробіли		0
Білі знаки		0
Парафрази (SmartMarks)		24

Подібності за списком джерел

Нижче наведений список джерел. В цьому списку є джерела із різних баз даних. Копір тексту означає в якому джерелі він був знайдений. Ці джерела і значення Коефіцієнту Подібності не відображають прямого плагіату. Необхідно відкрити кожне джерело і проаналізувати зміст і правильність оформлення джерела.

10 найдовших фраз

ПОРЯДКОВИЙ НОМЕР	НАЗВА ТА АДРЕСА ДЖЕРЕЛА URL (НАЗВА БАЗИ)	Копір тексту
		КІЛЬКІСТЬ ІДЕНТИЧНИХ СЛІВ (ФРАГМЕНТІВ)
1	Натиканий_Артур_Анатолійович_3-011_2025_123 Український національний авіаційний університет Ukrainian national aviation university (Криворізький Фаховий коледж) 6/2/2025	89 1.15 %
2	Натиканий_Артур_Анатолійович_3-011_2025_123 Український національний авіаційний університет Ukrainian national aviation university (Криворізький Фаховий коледж) 6/2/2025	42 0.54 %

РЕФЕРАТ

Дипломна робота «Проектування стенду для перевірки трекінгостійкості» містить 49 сторінок, 15 рисунків, 19 використаних джерела.

ТРЕКІНГОСТІЙКІСТЬ, ІЗОЛЯЦІЙНІ МАТЕРІАЛИ, АПАРАТНО ПРОГРАМНИЙ КОМПЛЕКС, КОМП'ЮТЕРНИЙ ЗІР, ІЕС 60112, ОБРОБКА ЗОБРАЖЕНЬ.

Дипломна робота присвячена розробці апаратно-програмного комплексу для оцінювання трекінгостійкості ізоляційних матеріалів.

Об'єктом дослідження є процес руйнування електроізоляційних матеріалів під впливом електричного поля, а предметом — методика аналізу трекінгових пошкоджень із використанням цифрової обробки зображень.

Метою роботи є створення лабораторного стенду, що дозволяє проводити випробування відповідно до вимог стандарту ІЕС 60112, а також програмного інструменту для автоматизованого аналізу результатів.

У процесі дослідження проаналізовано фізичні явища, що лежать в основі трекінгу, вимоги до електричної частини випробувального стенду, а також методи комп'ютерного зору для виявлення пошкоджень на зображеннях зразків. У третьому розділі детально описано архітектуру розробленого програмного забезпечення, принцип роботи інтерфейсу та результати тестування на реальних і синтетичних даних.

Запропоноване рішення є доступним, модульним і практично орієнтованим. Його можна використовувати в навчальному процесі, лабораторній діагностиці та для попереднього оцінювання нових ізоляційних матеріалів з метою підвищення надійності електроенергетичних систем та безпеки експлуатації.

ПЕРЕЛІК ПРИЙНЯТИХ СКОРОЧЕНЬ	6
ВСТУП.....	7 1
АНАЛІЗ ПРОБЛЕМИ ТА СУЧАСНИХ ПІДХОДІВ.....	9
1.1 Трекінгостійкість як критичний показник електроізоляційних матеріалів	
.9 1.2 Принципи та методи випробувань на трекінгостійкість	
.....10 1.3 Сучасні засоби цифрової обробки зображень для	
аналізу деградації ізоляції	
.....12 1.4	
Висновки до першого розділу	13
2 ПРОЄКТУВАННЯ АПАРАТНОЇ ЧАСТИНИ ЕКСПЕРИМЕНТАЛЬНОГО	
СТЕНДУ	14
2.1 Вступ.....	14 2.2
Конструкція експериментального стенду	14 2.3
Конструкція силової частини експериментального стенду	15 2.3
Техніка безпеки при проведенні експериментів.....	16 2.4
Висновки до другого розділу.....	18 3
РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ АНАЛІЗУ	
ТРЕКІНГОСТІЙКОСТІ.....	19
3.1 Загальна концепція та призначення програми.....	19
3.2 Інструменти та технології розробки	20
3.3 Архітектура програмного забезпечення	22
3.4 Алгоритм обробки зображення та виявлення трекінгових ушкоджень	25
3.5 Інтерфейс користувача та візуалізація результатів	32
3.6 Тестування та перевірка працездатності ПЗ.....	37
3.7 Висновки до третього розділу	41
ВИСНОВКИ	44
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ	48

ПЕРЕЛІК ПРИЙНЯТИХ СКОРОЧЕНЬ

СТІ – Comparative Tracking Index

GUI – Graphical User Interface

MVC – Model-View-Controller

OpenCV – Open Source Computer Vision Library

PIL – Python Imaging Library

SoC – Separation of Concerns

ЛАТР – лабораторний автотрансформатор

ООП – об'єктно-орієнтоване програмування

ОС – операційна система

ПЗ – програмне забезпечення

ШІ – штучний інтелект

7

ВСТУП

У сучасному світі, де електроенергія є критично важливим ресурсом у всіх галузях промисловості, транспорту та побуту, надійність електроізоляційних матеріалів набуває особливої ваги. Однією з найнебезпечніших форм деградації ізоляції є поверхневе пробоювання, викликане явищем трекінгу — утворенням провідних вуглецевих доріжок під дією електричного поля та присутності забруднень чи вологи. Виникнення таких треків може призвести до коротких замикань, займання проводки, виходу з ладу обладнання, що особливо критично у відповідальних системах.

Особливої актуальності ця проблема набуває в авіаційній галузі, де навіть незначне пошкодження ізоляції може призвести до катастрофічних наслідків. Електричні системи сучасних літаків працюють на високих напругах, а умови експлуатації — зниженого тиску, вібрації та коливань температури — створюють додаткові ризики для надійності ізоляції. Відомі випадки аварій, спричинених саме електричними пробоями та займанням кабельних трас [1].

Не менш небезпечними є умови підземних виробок, зокрема шахт, де

можлива наявність вибухонебезпечних газів або високого рівня пилу. У таких умовах вихід з ладу електричної ізоляції не лише зупиняє роботу, а може спровокувати масштабні вибухи та загрозу для життя людей.

Тому постає потреба у точному та надійному методі оцінювання трекінгостійкості ізоляційних матеріалів. Традиційні лабораторні методи вимагають складного обладнання, ручної оцінки та не завжди дозволяють об'єктивно зафіксувати ступінь пошкодження. У зв'язку з цим актуальним є впровадження цифрових методів обробки зображень для автоматизованої діагностики результатів випробувань на трекінгостійкість.

У межах цієї дипломної роботи планується реалізувати програмно апаратний комплекс, який дозволить виконувати візуальний аналіз поверхні ізоляційного матеріалу після впливу високої напруги в умовах дії електроліту, з подальшим визначенням кількісних показників ступеня пошкодження. Такий

8

підхід дозволяє зробити оцінку більш наочною, об'єктивною та придатною для документування результатів.

9

1 АНАЛІЗ ПРОБЛЕМИ ТА СУЧАСНИХ ПІДХОДІВ

1.1 Трекінгостійкість як критичний показник електроізоляційних матеріалів

Трекінгостійкість — це здатність електроізоляційного матеріалу протистояти утворенню провідних доріжок (треків) під дією електричного поля, зволоження, забруднення чи електролітичного впливу. Це явище проявляється у вигляді поступового руйнування поверхні полімерного матеріалу, коли між двома електродами по поверхні утворюється провідний слід, який зрештою може призвести до пробоя, короткого замикання або займання (рис.1.1).

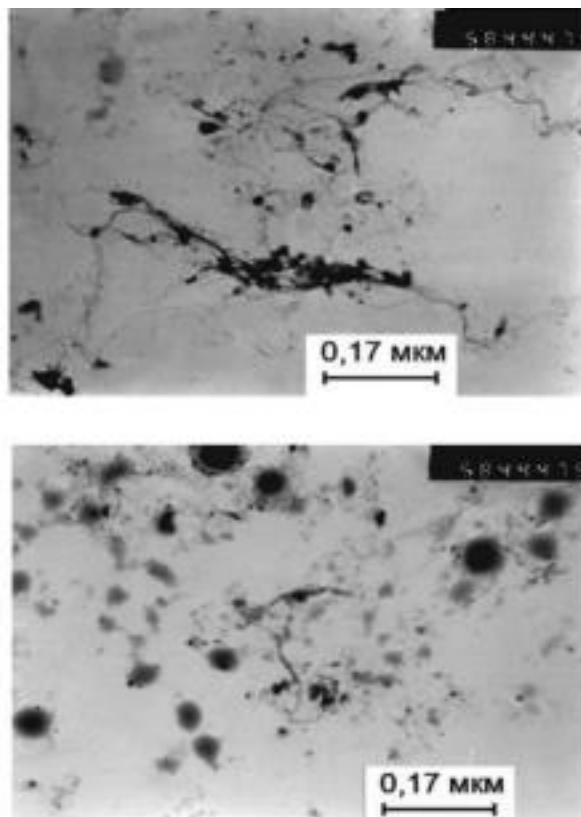


Рисунок 1.1 – Зображення тренінгового явища на кремнійорганічній ізоляції (фото отримано за допомогою електронного мікроскопу)

У сучасній практиці трекінгостійкість є одним із ключових параметрів при сертифікації та виборі полімерних матеріалів для використання в силових електричних кабелях, комутаційній апаратурі, елементах побутових електроприладів, авіаційній електроніці тощо. Особливо високі вимоги до цього

10

параметра пред'являються в умовах підвищеної вологості, наявності забруднень, змін атмосферного тиску та температури.

Пошкодження ізоляції внаслідок трекінгу відрізняється від внутрішнього електричного пробою тим, що виникає на поверхні матеріалу, часто поступово й без помітного зниження електричної міцності до моменту критичного руйнування. Це ускладнює виявлення початкових ознак деградації. У промислових умовах саме поверхневі треки є причиною значної кількості відмов обладнання, зокрема при тривалій експлуатації в агресивному середовищі.

Стандартизація вимог до трекінгостійкості проводиться згідно з міжнародними нормами, зокрема ІЕС 60112, яка визначає методику вимірювання порогової напруги поверхневого пробою (Comparative Tracking

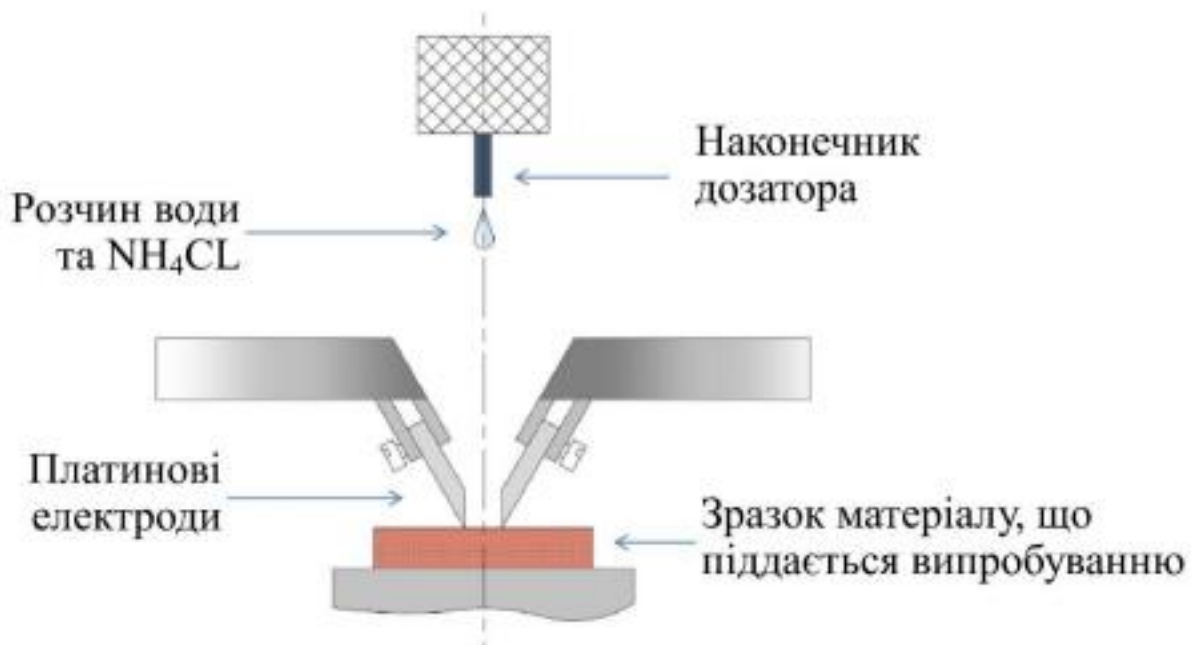
Index — СТІ) [2]. Матеріали з високим СТІ мають кращу стійкість до трекінгу і вважаються більш надійними для критичних застосувань.

У межах даної роботи трекінгостійкість розглядається як основний об'єкт дослідження, і на її основі планується реалізувати програмне забезпечення (ПЗ) для аналізу результатів випробувань за візуальними критеріями, що дозволяє підвищити об'єктивність оцінки та повторюваність експериментів.

1.2 Принципи та методи випробувань на трекінгостійкість

Оцінка трекінгостійкості ізоляційних матеріалів ґрунтується на випробуваннях, що моделюють реальні умови експлуатації, зокрема вологість, забруднення, напругу та наявність електроліту. Один із найпоширеніших методів — методика відповідно до стандарту ІЕС 60112, яка визначає порогову напругу поверхневого пробою СТІ [2].

Згідно з цією методикою, зразок матеріалу встановлюється між двома електродами під певним кутом (зазвичай 45°), на поверхню подається струмопровідна рідина (електроліт), а між електродами прикладається напруга змінного струму. Випробування продовжуються до моменту виникнення пробою або до досягнення встановленого часу експозиції (рис. 1.2).



Якщо пробій не виникає — напруга підвищується. Вимірюється напруга, при якій виникає утворення треків або струм, що перевищує критичне значення (наприклад, 0.5 А) протягом 2 секунд.

Існують також інші стандартизовані методики, зокрема ASTM D3638, UL 746A та інші [3], які використовують аналогічні підходи до визначення СТІ. Основна мета таких випробувань — оцінити стійкість матеріалу до електричного руйнування в умовах, близьких до експлуатаційних.

Крім класичних лабораторних випробувань, у сучасних дослідженнях все ширше застосовуються цифрові методи аналізу поверхневих пошкоджень, включаючи візуальну фіксацію деградації, машинне навчання та алгоритми комп'ютерного зору. Такі підходи дозволяють швидше та точніше оцінити ступінь пошкодження, кількість провідних каналів, глибину треків, площу ураження тощо [4].

У межах цієї дипломної роботи буде реалізовано експериментальну установку, що моделює умови впливу електричного поля на ізоляційний матеріал у присутності електроліту, з подальшою фіксацією змін на поверхні за допомогою цифрової камери та аналізу зображень у спеціально розробленій

12

програмі. Такий підхід дозволяє зробити оцінювання трекінгостійкості доступним, об'єктивним і повторюваним.

1.3 Сучасні засоби цифрової обробки зображень для аналізу деградації ізоляції

Цифрові технології відіграють дедалі важливішу роль у галузі діагностики технічного стану матеріалів. Однією з найефективніших методик для візуального аналізу поверхневих дефектів є комп'ютерний зір — галузь, яка поєднує обробку зображень, штучний інтелект та математичну аналітику для виявлення пошкоджень, змін або деградації на зображеннях (рис.1.3).

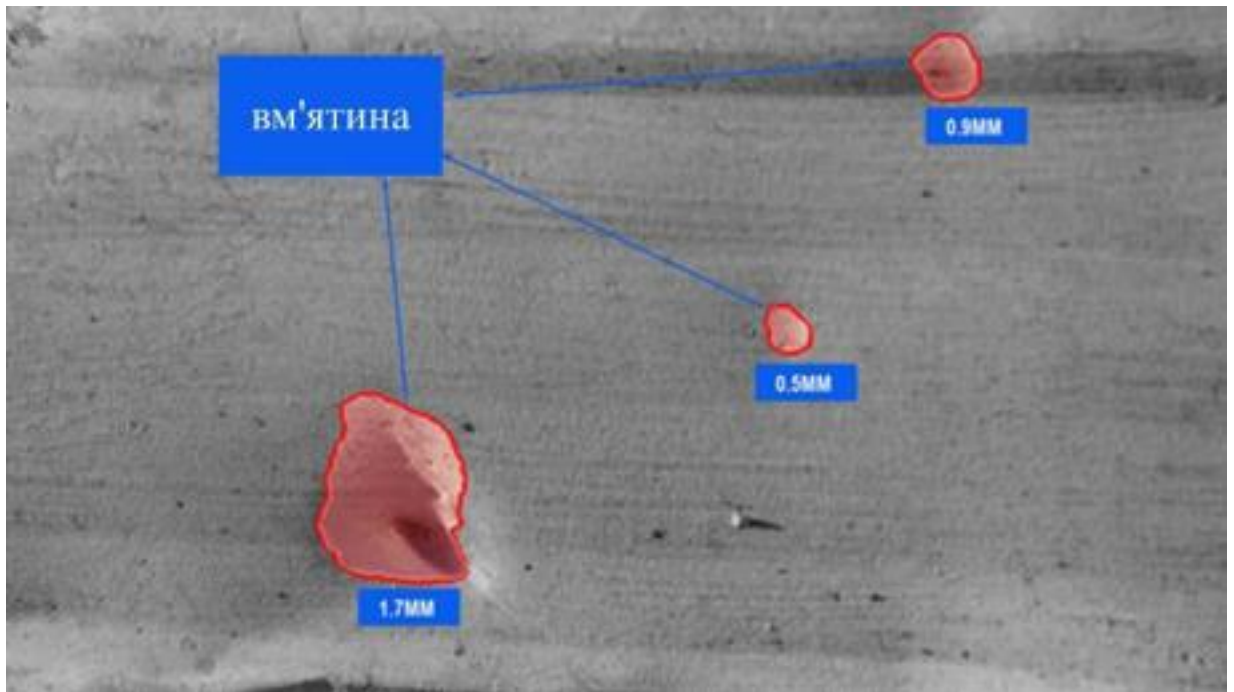


Рисунок 1.3 – Приклад розпізнавання дефектів (вмятин) на неоднорідній поверхні. У випадку трекінгостійкості, комп'ютерний зір дозволяє автоматизувати процес виявлення провідних треків, крапель електроліту, зон обвуглення або інших ознак поверхневої деградації. Основні технології, що використовуються у таких задачах, включають:

- фільтрацію шуму зображення (наприклад, Gaussian Blur);
- виявлення країв методом Собеля або Прюїтта;
- порогову обробку (Thresholding);
- морфологічні операції (ерозія, дилатація);
- класифікацію об'єктів за розміром та формою.

Дослідження [4] показують, що аналіз зображення дозволяє кількісно оцінити площу пошкодженої поверхні, кількість утворених треків та їхню щільність. Також можливе визначення показника деградації за співвідношенням площі дефекту до загальної площі зразка або через кількість виділених елементів (наприклад, крапель).

Використання таких методів є особливо доцільним при серійному тестуванні великої кількості зразків, де людський фактор або ручна оцінка можуть бути ненадійними або затратними за часом.

1.4 Висновки до першого розділу

У першому розділі дипломної роботи було проаналізовано природу та наслідки явища трекінгу в ізоляційних матеріалах, наведено основні стандарти та методики випробувань, а також розглянуто сучасні засоби цифрової обробки зображень, які можуть бути застосовані для оцінки трекінгостійкості.

Результати аналізу підтверджують, що трекінг є критичним фактором, що обмежує надійність ізоляційних систем у відповідальних галузях, зокрема в авіації та гірничій справі. Традиційні лабораторні методики, хоча й ефективні, мають певні обмеження щодо об'єктивності та масштабованості.

Застосування методів комп'ютерного зору дозволяє автоматизувати та уніфікувати процес діагностики, зменшити вплив людського фактору, а також швидко отримувати кількісні показники, що важливо для технічної оцінки матеріалів. Такий підхід відкриває нові можливості для подальшого вдосконалення методів оцінювання трекінгостійкості та розробки відповідного ПЗ.

14

2 ПРОЄКТУВАННЯ АПАРАТНОЇ ЧАСТИНИ ЕКСПЕРИМЕНТАЛЬНОГО СТЕНДУ

2.1 Вступ

У другому розділі дипломної роботи розглядаються основні етапи проєктування апаратного комплексу для оцінки трекінгостійкості ізоляційних матеріалів. Цей комплекс включає експериментальну установку, яка дозволяє здійснити вплив електричним полем на зразок у присутності електроліту, а також зафіксувати результати для подальшого аналізу.

Розділ описує загальні підходи до побудови лабораторного стенду, вимоги до джерела напруги, конструкцію тримача зразка, особливості електродів, властивості електроліту та безпечну методику проведення випробувань. Також розглянуто можливі засоби фіксації зображення, включно з використанням цифрових камер.

Метою цього розділу є обґрунтування вибору технічних рішень та

демонстрація створеної експериментальної моделі, яка може використовуватись як у дослідженнях, так і в навчальному процесі.

2.2 Конструкція експериментального стенду

Основу стенду для дослідження трекінгостійкості становить модуль, що імітує типові умови електричного навантаження та дії навколишнього середовища на ізоляційний матеріал (рис.2.1). Для реалізації експериментів передбачено використання двох електродів, між якими розміщується зразок досліджуваного матеріалу. Між електродами прикладається напруга змінного струму, а на поверхню зразка наноситься крапля електроліту з фіксованим об'ємом.

Конструктивно електроди виконані з латуні або нержавіючої сталі, мають форму циліндричних стрижнів з плоскими торцями. Один з електродів є фіксованим, інший — регульованим, що дозволяє точно виставити міжелектродний зазор (типово — 4 мм згідно з ІЕС 60112 [2]). Геометрія розміщення відповідає стандартним вимогам до симетричного впливу електричного поля.

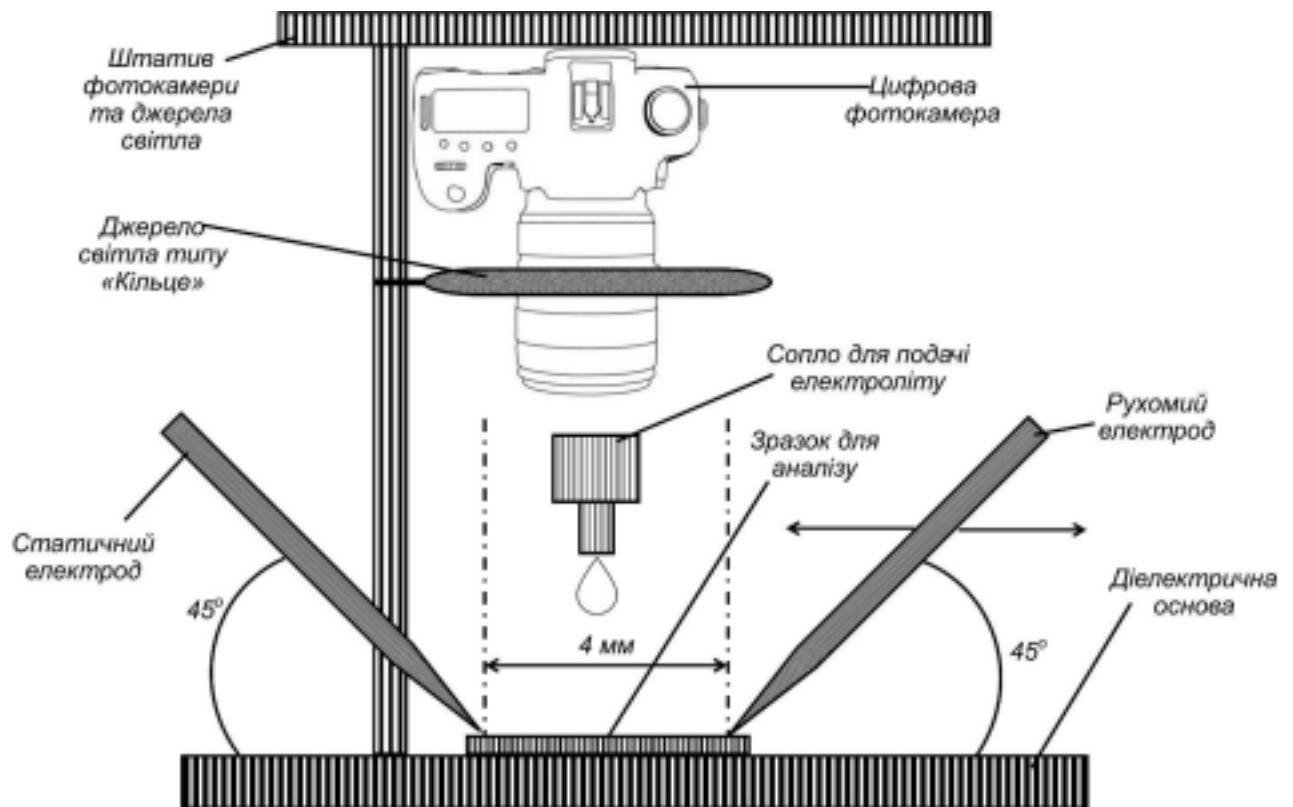


Рисунок 2.1 – Схематичне зображення експериментальної частини стану

Зразок встановлюється на підставці з ізоляційного матеріалу, яка витримує високу температуру та електричне навантаження. Подача електроліту здійснюється або вручну за допомогою мікропіпетки, або автоматизовано крапельно. Як електроліт застосовують розчин на основі амонію хлориду у дистильованій воді (0,1 % масова концентрація), який забезпечує провідність, достатню для ініціювання поверхневих процесів деградації.

2.3 Конструкція силової частини експериментального стану

Електрична частина стану (рис. 2.2) передбачає наявність регульованого джерела змінної напруги з вихідною напругою до 600 В, частотою 50 Гц. Для

16

забезпечення точності та безпеки експерименту передбачено вольтметр та амперметр для контролю за рівнем напруги та струму у ланцюзі. Вимірювання струму дозволяє зафіксувати момент пробою або початку струмопровідного трекінгу, коли струм перевищує порогове значення (наприклад, 0,5 А).

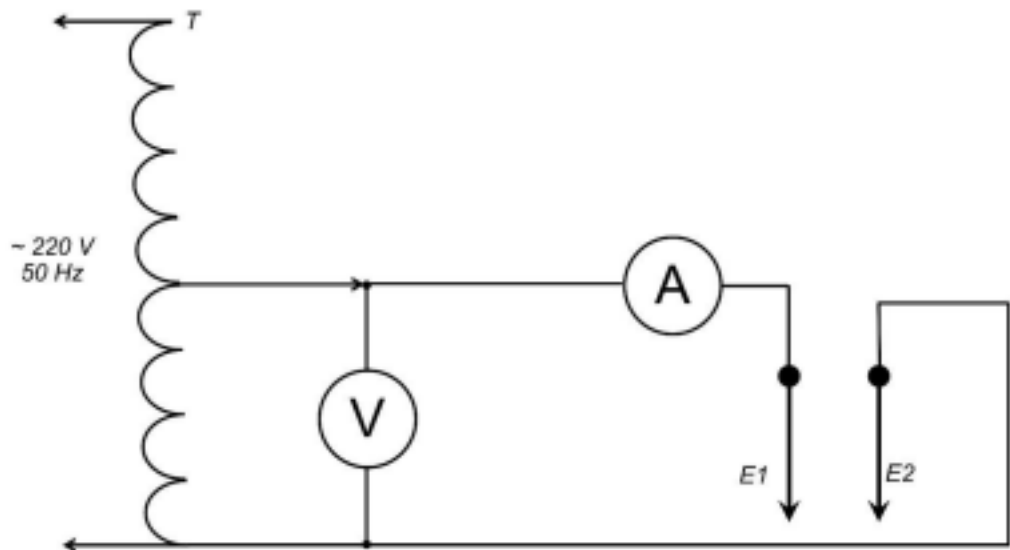


Рисунок 2.2 – Схема електричної силової установки експериментального стенду

Ланцюг живлення включає в себе трансформатор, який забезпечує гальванічну розв'язку від мережі, а також блок захисту, який автоматично вимикає живлення при перевищенні допустимих параметрів або короткому замиканні. Для плавного регулювання напруги може використовуватись лабораторний автотрансформатор (ЛАТР).

Живлення електродів здійснюється за допомогою мідних провідників, що мають термостійку ізоляцію. Контакти до електродів зафіксовані в клеммах з гвинтовим затискачем, що дозволяє швидко змінювати зразки та налаштовувати положення елементів.

2.3 Техніка безпеки при проведенні експериментів

Проведення випробувань на трекінгостійкість пов'язане з роботою під напругою, впливом електроліту та можливістю локального перегріву або

17

займання зразків. Тому дотримання техніки безпеки є обов'язковою умовою роботи з експериментальним стендом.

До початку роботи:

- необхідно переконатися в справності стенду, ізоляції проводів та відсутності пошкоджень;

- перевірити роботу блоків живлення та автоматичного вимикача; -
- переконатися в наявності захисного екрана над зоною випробувань; - надіти індивідуальні засоби захисту: діелектричні рукавички, окуляри, халат.

Під час проведення експерименту:

- категорично заборонено торкатися струмопровідних частин; -
- заборонено знімати або відкривати захисний кожух під час подачі напруги;
- слід проводити спостереження на безпечній відстані за допомогою камери;
- при появі стороннього запаху, диму або нестандартного нагріву – негайно знеструмити установку.

Після завершення роботи:

- переконатися у вимкненні напруги та відсутності залишкових зарядів;
- очистити зону випробувань від залишків електроліту;
- видалити зразок із ділянки експерименту для аналізу.

Для надзвичайних ситуацій поблизу має бути вогнегасник, аптечка, знак аварійного вимкнення живлення. Всі випробування мають проводитись під наглядом завідувача лабораторією або відповідального фахівця підприємства.

Дотримання вищезазначених вимог дозволяє звести до мінімуму ризику ураження електричним струмом, займання або пошкодження обладнання. Уся система має бути розміщена в корпусі з діелектричного матеріалу з прозорою захисною кришкою для спостереження та мінімізації ризиків

18

електротравм. Такий підхід дозволяє провести кількісну оцінку ступеня деградації та зафіксувати відтворюваність результатів.

2.4 Висновки до другого розділу

У другому розділі дипломної роботи було розглянуто конструкцію експериментального стенду для визначення трекінгостійкості ізоляційних матеріалів. Проєкт стенду включає електродну систему, тримач зразків, джерело живлення з регулюванням напруги, захисні елементи та систему фіксації

зображень для подальшого аналізу.

Особливу увагу приділено електричній частині установки, що забезпечує стабільність та безпечність експерименту. Реалізовані заходи електробезпеки та технічного захисту дозволяють проводити дослідження в умовах лабораторії без ризику для оператора.

Визначено основні вимоги до умов випробування, електроліту та контролю параметрів під час експерименту. Запропонована конструкція стенду є простою, доступною для виготовлення в навчальній лабораторії та забезпечує достовірну фіксацію результатів деградації ізоляційного матеріалу для подальшої обробки програмними засобами.

Проведення випробувань на трекінгостійкість пов'язане з роботою під напругою, впливом електроліту та можливістю локального перегріву або займання зразків. Тому дотримання техніки безпеки є обов'язковою умовою роботи з експериментальним стендом.

19

3 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ АНАЛІЗУ ТРЕКІНГОСТІЙКОСТІ

3.1 Загальна концепція та призначення програми

Аналіз трекінгостійкості ізоляційних матеріалів є ключовим етапом у забезпеченні надійності електроізоляційних систем, особливо у відповідальних сферах — таких як авіація, енергетика, транспорт, гірничодобувна промисловість тощо. У таких галузях ізоляційний матеріал має витримувати не лише високу напругу, а й вплив вологи, забруднень, перепадів температури та механічних навантажень. Наявність трекінгових пошкоджень — провідних вуглецевих доріжок, які утворюються внаслідок поверхневої деградації — свідчить про зниження діелектричних властивостей матеріалу та небезпеку його подальшого використання.

Традиційно оцінка трекінгостійкості здійснюється шляхом візуального огляду зразка після випробування: фіксується наявність обвуглення, треків,

струмопровідних доріжок або місць пробою. Такий підхід передбачає участь оператора та часто не дозволяє провести точну та об'єктивну оцінку ступеня деградації. У разі потреби обробки великої кількості зразків або повторюваних серій експериментів зростає ризик помилок та витрати часу. Саме тому все більш актуальними стають засоби цифрового аналізу поверхонь, зокрема з використанням методів комп'ютерного зору та штучного інтелекту [5,6].

Метою розробки ПЗ, що реалізовано в межах цієї дипломної роботи, є створення зручного, доступного та ефективного інструменту для обробки зображень поверхонь ізоляційних зразків після випробування на трекінгостійкість. Основні задачі, які вирішує програма:

- завантаження зображення зразка після експерименту;
- фільтрація зображення для усунення шумів та фону;
- виявлення меж, крапель, треків або зон обвуглення;

20

- підрахунок об'єктів (наприклад, кількість крапель електроліту, що просочився);
- розрахунок умовного показника деградації або трекінгостійкості; - виведення результату у зручному для оператора вигляді (включно з коротким звітом).

Програмне забезпечення працює у середовищі Windows, має графічний інтерфейс користувача (GUI, Graphical User Interface), побудований із використанням бібліотеки Tkinter. Взаємодія з користувачем є інтуїтивною: оператор має змогу завантажити зображення, натиснути кнопку аналізу та побачити результат у числовому й візуальному вигляді. Це значно спрощує проведення серії випробувань, особливо в умовах навчальних лабораторій або інженерних досліджень [7].

Ключова перевага полягає у тому, що цифрова обробка дозволяє зафіксувати деталі, які можуть бути непомітними для ока. Крім того, всі результати автоматично фіксуються в пам'яті комп'ютера, що забезпечує наочність, повторюваність експерименту та зручність подальшої обробки. Такий підхід відкриває шлях до стандартизації процедури аналізу деградації ізоляційних матеріалів, а також до подальшого вдосконалення алгоритмів за

допомогою машинного навчання [8].

Програму можна масштабувати або адаптувати під нові типи зразків чи алгоритми. У майбутньому можливе впровадження модулів автоматичної класифікації ступеня деградації, розпізнавання типів пошкоджень та інтеграції з базами даних матеріалів. Таким чином, створений програмний інструмент є важливою ланкою між фізичним експериментом та об'єктивним аналізом результатів, що є актуальним трендом у технічній освіті та науці.

3.2 Інструменти та технології розробки

Розробка програмного забезпечення для аналізу трекінгостійкості ізоляційних матеріалів потребує ретельного вибору інструментів, які б

21

забезпечували ефективну роботу з зображеннями, зручну побудову графічного інтерфейсу користувача та легку інтеграцію з іншими модулями. У межах даної дипломної роботи як основну мову програмування було обрано Python, що пояснюється її широким поширенням, великою кількістю бібліотек для обробки зображень, простим синтаксисом і високою швидкістю розробки [9].

Python належить до мов високого рівня та підтримується активною спільнотою, що забезпечує доступ до безкоштовних бібліотек і документації. Завдяки відкритому коду і численним прикладам, Python став однією з найпопулярніших мов у сфері машинного навчання, комп'ютерного зору та автоматизованої обробки даних [10].

Для реалізації алгоритмів обробки зображення використано бібліотеку OpenCV (Open Source Computer Vision Library) — потужний набір інструментів для комп'ютерного зору, розпізнавання образів, роботи з відео та фотографіями. OpenCV підтримує численні функції: фільтрацію, перетворення кольорових просторів, виявлення контурів, морфологічні операції, порогову обробку тощо [11]. Саме ця бібліотека дозволила реалізувати основний функціонал виявлення трекінгових ушкоджень шляхом аналізу зображення після випробування зразка.

Для обчислювальних операцій, таких як підрахунок об'єктів, статистичні

розрахунки або фільтрація шумів, застосовувалась бібліотека NumPy. Вона забезпечує швидку роботу з багатовимірними масивами даних, векторизацію операцій та інтегрується з OpenCV для виконання матричних і логічних операцій [12].

Для роботи з графікою та відображенням зображень у зручному форматі використовувалась бібліотека Pillow (форк класичної PIL — Python Imaging Library). Вона забезпечує збереження, масштабування, конвертацію зображень між форматами, накладання графічних елементів, що необхідно для візуалізації результатів аналізу та формування звітів [13].

Графічний інтерфейс користувача (GUI) реалізовано за допомогою Tkinter — стандартної бібліотеки для створення віконних додатків у Python. Tkinter дозволяє легко створювати кнопки, меню, поля вводу, панелі, зображення та

22

повідомлення. Його перевага — повна інтеграція з Python, простота налаштування, відсутність потреби в додаткових інсталяціях та висока сумісність із різними ОС (Windows, Linux, macOS) [14].

Також використано інструменти розробки, зокрема середовище Visual Studio 2022 з підтримкою Python-проєктів. Це середовище дозволяє виконувати відлагодження коду, перегляд результатів роботи, керування залежностями проєкту та створення дистрибутивів для розповсюдження.

Важливо зазначити, що всі використані інструменти є безкоштовними, мають відкриту ліцензію і доступні для повторного використання в освітніх, наукових або промислових цілях. Такий вибір дозволяє масштабувати ПЗ, легко інтегрувати його з майбутніми модулями наприклад, на базі штучного інтелекту (ШІ), та адаптувати до змін вимог користувача чи технологій обробки зображень. Таким чином, поєднання Python, OpenCV, NumPy, Pillow та Tkinter забезпечило необхідну функціональність, гнучкість, простоту підтримки та якісну реалізацію задач, поставлених у межах дипломної роботи.

3.3 Архітектура програмного забезпечення

Архітектура ПЗ відіграє ключову роль у забезпеченні стабільної та масштабованої роботи системи. Вона визначає, як саме компоненти програми взаємодіють між собою, як реалізуються основні функції, та наскільки зручно підтримувати, оновлювати і тестувати програмний продукт. У цій дипломній роботі архітектура була побудована з урахуванням принципів модульності, зрозумілої структури коду та простоти інтеграції нових функцій у майбутньому [15].

Розроблене програмне забезпечення можна умовно поділити на три основні компоненти:

- модуль графічного інтерфейсу користувача (GUI);
- модуль обробки зображень (CV Module);
- модуль логіки та аналізу результатів (Logic / Analysis Module).

23

Кожен з цих компонентів має чітко визначену відповідальність і взаємодіє з іншими через інтерфейси, що полегшує підтримку та масштабування коду. Також усі частини побудовані з урахуванням принципу розділення відповідальності (SoC — Separation of Concerns) [16].

3.3.1. Модуль GUI

Інтерфейс користувача реалізовано з використанням бібліотеки Tkinter.

Основні елементи інтерфейсу:

- кнопка завантаження зображення;
- панель відображення зображення;
- кнопка запуску аналізу;
- кнопка збереження результату (наприклад, у форматі PNG або CSV) (опціонально, в тестовому режимі не використовується);
- текстове поле з виводом кількості виявлених об'єктів (треків, крапель).

Завдяки поділу на функціональні блоки, будь-які зміни в інтерфейсі (наприклад, переклад, зміна дизайну, адаптація під сенсорні пристрої) не потребують втручання в основну логіку програми [17].

3.3.2. Модуль обробки зображень

Цей модуль реалізує основні етапи комп'ютерного зору, що

- включають: - масштабування та нормалізацію зображення;
- фільтрацію шумів (за допомогою Гаусового або медіанного фільтру); - перетворення в відтінки сірого або HSV-простір;
- виявлення країв (оператори Собеля, Прюїтта або Canny); - морфологічну обробку (ерозія, дилатація);
- бінаризацію та виявлення контурів;
- розрахунок площі, форми та положення об'єктів [11].

OpenCV дозволяє легко реалізувати ці кроки, забезпечуючи стабільність результатів і швидкість виконання. Додатково, модуль підтримує попередню візуалізацію обробленого зображення перед виведенням остаточного результату користувачу.

24

3.3.3. Модуль логіки та аналізу результатів

Після обробки зображення система отримує набір контурів об'єктів, які підлягають аналізу. Кожен контур проходить перевірку за критеріями площі, симетрії або геометричних особливостей. Результати аналізу інтерпретуються

як: - кількість пошкоджень (умовно: крапель або треків);

- площа пошкодження відносно площі всього зразка;

- умовний індекс деградації (наприклад, від 0 до 1 або у відсотках). Ці

дані передаються в GUI для відображення та можуть бути збережені для формування звіту. За бажанням користувача можна інтегрувати модуль автоматичного збереження результатів у файл.

Структура файлів у проєкті представлено на рис. 3.1, де:

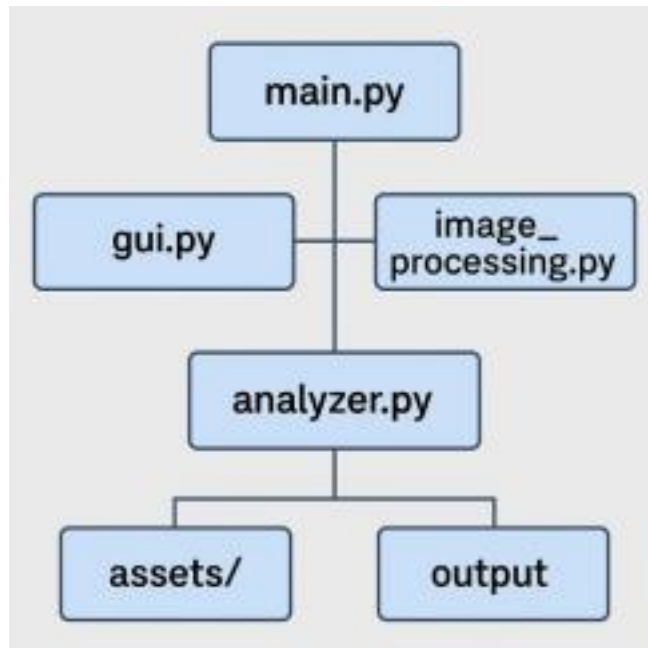


Рисунок 3.1 – Структура файлів у проекті стенду перевірки трекінгостійкості

- main.py — головний файл, що запускає програму та викликає інтерфейс;

- gui.py — функції побудови вікон, кнопок, відображення; -

image_processing.py — основні фільтри, перетворення та контурний аналіз;

- analyzer.py — логіка підрахунку та інтерпретації результатів;

25

- assets/ — папка зі зображеннями для тестування або оформлення інтерфейсу;

- output/ — результати роботи програми.

Переваги архітектурного підходу:

- зручність модульного тестування;

- можливість додавання нових методів обробки без втручання в існуючий код;

- підтримка різних форматів зображень (JPEG, PNG, BMP); - можливість підключення інших інтерфейсів (наприклад, WebGUI у майбутньому).

У реалізації застосовувалися принципи ООП (об'єктно-орієнтованого програмування), а також шаблон MVC (Model-View-Controller) у спрощеному вигляді — де логіка, візуалізація і дані розділені між модулями [18]. Це дозволяє зберігати чистоту архітектури, зменшити кількість помилок та підвищити

читабельність коду.

Таким чином, обрана архітектура дозволила реалізувати стабільне, зрозуміле та гнучке ПЗ. Програма легко масштабується, адаптується до змін і підходить для використання як у навчальних цілях, так і в реальних лабораторних умовах. Поділ на модулі дозволяє ефективно оновлювати окремі компоненти, реалізовувати нові алгоритми або інтегрувати з іншими системами аналізу зображень.

3.4 Алгоритм обробки зображення та виявлення трекінгових ушкоджень

Ключовим функціональним елементом розробленого ПЗ є модуль комп'ютерного зору, що забезпечує автоматичний аналіз зображення зразка після впливу високої напруги. Обробка зображення виконується у кілька послідовних етапів, кожен з яких має свою роль у підготовці даних до аналізу та прийнятті рішення щодо ступеня деградації матеріалу.

26

Попередня обробка зображення. Першим етапом є завантаження зображення та його попередня обробка. Зазвичай камера знімає зразок у кольоровому форматі (RGB), однак для більшості операцій аналізу зображення доцільніше працювати у відтінках сірого. Тому здійснюється перетворення кольорового зображення у чорно-біле за допомогою функції:

`cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)` [11].

Після цього виконується фільтрація шумів. Найчастіше використовується Гаусове розмиття (GaussianBlur) з ядром розміру 5×5 або 3×3. Цей крок дозволяє згладити фон, прибрати поодинокі піксельні аномалії, які можуть бути інтерпретовані як артефакти трекінгу (рис. 3.2).



Рисунок

3.2 – Приклад зображення зразка до та після фільтрації шумів методом GaussianBlur

Бінаризація та виявлення контурів. Після фільтрації застосовується бінаризація зображення — процес, при якому зображення перетворюється на чорно-біле, де об'єкти відокремлюються від фону. Для цього використовується метод Otsu Thresholding:

```
_ , thresh = cv2.threshold(blurred, 0, 255, cv2.THRESH_BINARY +  
cv2.THRESH_OTSU)
```

Метод Отсу автоматично підбирає поріг, що мінімізує дисперсію всередині кластерів пікселів.

27

Отримане зображення піддається морфологічній обробці (операція відкриття — `morphologyEx`), що дозволяє усунути дрібні злиття об'єктів і зробити зони трекінгу більш виразними (рис. 3.3).

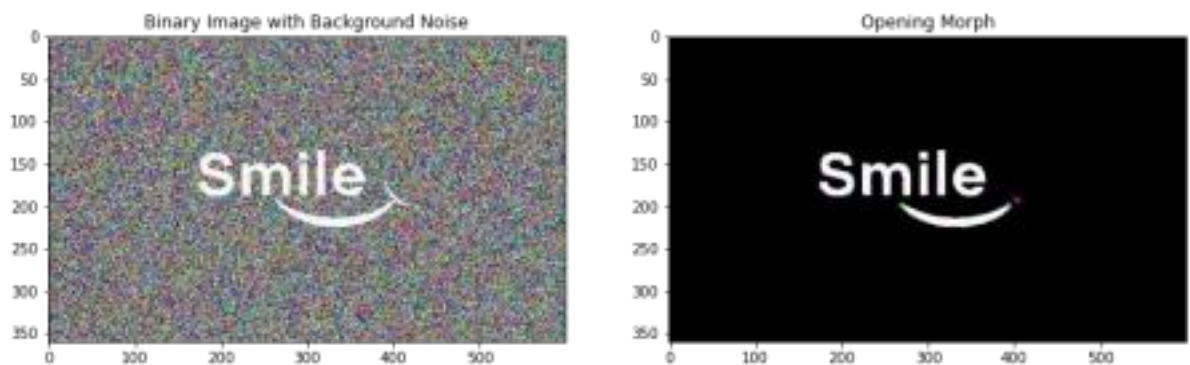


Рисунок 3.3 – Морфологічне очищення зображення: приклад злитих плям до і після операції відкриття (`morphologyEx`)

Далі виконується побудова контурів:

```
contours, _ = cv2.findContours(thresh, cv2.RETR_EXTERNAL,
```

cv2.CHAIN_APPROX_SIMPLE)

Ця функція повертає список контурів, кожен з яких є окремим об'єктом. Саме ці контури будуть використані для підрахунку крапель, зон пробою або слідів трекінгу.

Фільтрація та аналіз контурів. Не всі виявлені контури є корисними — деякі можуть бути шумами або артефактами. Тому проводиться фільтрація за площею:

$$areas = [cv2.contourArea(cnt) \text{ for } cnt \text{ in } contours \text{ if } cv2.contourArea(cnt) > min_area]$$

Зазвичай мінімальна площа задається емпірично (наприклад, 5–10 пікселів).

Для кожного валідного контуру розраховується:

- площа (`cv2.contourArea()`);
- довжина контуру (`cv2.arcLength()`);
- еквівалентний діаметр (за формулою $D = \sqrt{(4 \cdot S / \pi)}$);
- розташування (центроїд).

28

Ці дані дозволяють сформуванати узагальнену оцінку:

- N — кількість треків (плям, крапель);
- S_{total} — сумарна площа ураження;
- S_{avg} — середня площа однієї зони;
- D_{eq} — середній діаметр ураження;
- Coverage — відсоток покриття (відношення ураженої площі до загальної площі зразка).

Візуалізація та повернення результатів. Після аналізу контурів зображення перетворюється у формат RGB, і на нього накладаються всі виявлені контури, підписи та статистична інформація. Наприклад:

$$cv2.drawContours(img_bgr, contours, -1, (0, 255, 0), 2)$$

Після чого результат виводиться у вікно або зберігається в файл. Виведена інформація включає загальну кількість виявлених дефектів, середню площу, а також формується короткий протокол.

На основі отриманих показників користувач має змогу зробити висновок про ступінь деградації матеріалу. Наприклад, при незначній кількості треків та малій площі ураження — можна зробити висновок про високу трекінгостійкість. Якщо ж площа покриття перевищує 10–20 %, це свідчить про критичне зниження електроізоляційних властивостей [4].

Таким чином, алгоритм обробки зображень реалізує повноцінний цикл аналізу: від підготовки зображення — до генерації кількісного висновку, що є основою для об'єктивної оцінки трекінгостійкості.

Повний код програми на мові програмування Python приведено

нижче: *import cv2*

import numpy as np

import tkinter as tk

from tkinter import filedialog, messagebox

from PIL import Image, ImageTk

import math

class TrackingApp:

def __init__(self, root):

self.root = root

self.root.title("Аналіз трекінгостійкості ізоляційних матеріалів") #

Назва закладу та заголовок

header_label = tk.Label(root, text="КРИВОРІЗЬКИЙ ФАХОВИЙ КОЛЕДЖ\n"

"ДЕРЖАВНОГО НЕКОМЕРЦІЙНОГО ПІДПРИЄМСТВА\n"

"«ДЕРЖАВНИЙ УНІВЕРСИТЕТ «КИЇВСЬКИЙ АВІАЦІЙНИЙ ІНСТИТУТ»",

font=("Arial", 14, "bold"))

header_label.pack(pady=(10, 0))

subheader_label = tk.Label(

root,

text="Програма 'Tracking Resistance Analyzer' розроблена в рамках дипломної

роботи\n"

"на тему 'Проектування стенду для перевірки трекінгостійкості'\n"

```

"Здобувачем освіти: Антонюк Є.О.\n"
"Керівник: Кутін А.І.",
font=("Arial", 10),
justify="center"
)
subheader_label.pack(pady=(0, 10))
self.image_panel = tk.Label(root)
self.image_panel.pack(pady=10)
self.result_label = tk.Label(root, text="Result: ", font=("Arial", 12), justify="left")
self.result_label.pack(pady=5)
btn_frame = tk.Frame(root)
btn_frame.pack(pady=10)
load_btn = tk.Button(btn_frame, text="Завантажити ображення",
command=self.load_image)
load_btn.grid(row=0, column=0, padx=5)

analyze_btn = tk.Button(btn_frame, text="Аналіз", command=self.analyze_image)
analyze_btn.grid(row=0, column=1, padx=5)
exit_btn = tk.Button(btn_frame, text="Вихід", command=self.root.quit)
exit_btn.grid(row=0, column=2, padx=5)

self.cv_img = None
def load_image(self):
file_path = filedialog.askopenfilename(filetypes=[("Image Files", "*.jpg *.jpeg
*.png")])
if not file_path:
return
self.cv_img = cv2.imread(file_path, cv2.IMREAD_GRAYSCALE) if
self.cv_img is None:
messagebox.showerror("Error", "Failed to load image.")
return
self.display_image(self.cv_img)

```

```

self.result_label.config(text="Result: ")
def display_image(self, img):
img_resized = cv2.resize(img, (400, 400))
img_pil = Image.fromarray(img_resized)
img_tk = ImageTk.PhotoImage(img_pil)
self.image_panel.configure(image=img_tk)
self.image_panel.image = img_tk
def analyze_image(self):
if self.cv_img is None:
messagebox.showwarning("Warning", "Please load an image first.")
return
img_area = self.cv_img.shape[0] * self.cv_img.shape[1]
# 1. Гладимо зображення для зменшення шуму
blurred = cv2.GaussianBlur(self.cv_img, (5, 5), 0)
# 2. Порогова обробка з автоматичним підбором (OTSU) _, thresh =
cv2.threshold(blurred, 0, 255, cv2.THRESH_BINARY + cv2.THRESH_OTSU)
# 3. Морфологія: відокремлення злиплених об'єктів
kernel = np.ones((3, 3), np.uint8)
thresh = cv2.morphologyEx(thresh, cv2.MORPH_OPEN, kernel, iterations=1) #
4. Пошук контурів
contours, _ = cv2.findContours(thresh, cv2.RETR_EXTERNAL,
cv2.CHAIN_APPROX_SIMPLE)
areas = [cv2.contourArea(cnt) for cnt in contours if cv2.contourArea(cnt) > 1]
drop_count = len(areas)
total_area = sum(areas)
avg_area = total_area / drop_count if drop_count > 0 else 0
max_area = max(areas) if areas else 0
diameter_eq = math.sqrt((4 * avg_area) / math.pi) if avg_area > 0 else 0
coverage = (total_area / img_area) * 100
result_text = (

```

```

f"Крапель: {drop_count}\n"
f"Сумарна площа: {int(total_area)} пікселів\n"
f"Середня площа: {avg_area:.1f} пікселів\n"
f"Макс. площа: {max_area:.1f} пікселів\n"
f"D_eq: {diameter_eq:.1f} пікселів\n"
f"Покриття: {coverage:.2f}%"
)
self.result_label.config(text=result_text)
# Візуалізація результату
color_img = cv2.cvtColor(self.cv_img, cv2.COLOR_GRAY2BGR)
cv2.drawContours(color_img, contours, -1, (0, 255, 0), 1)
self.display_image(cv2.cvtColor(color_img, cv2.COLOR_BGR2RGB)) if
__name__ == "__main__":

root = tk.Tk()
app = TrackingApp(root)
root.mainloop()

```

32

3.5 Інтерфейс користувача та візуалізація результатів

Інтерфейс користувача є важливою складовою будь-якого ПЗ, особливо якщо воно призначене для використання в умовах лабораторних випробувань або в освітньому процесі. Головна мета графічного інтерфейсу — забезпечити інтуїтивну взаємодію користувача з програмним модулем без потреби вивчення командного рядка або програмного коду [17].

У рамках цієї дипломної роботи було реалізовано інтерфейс користувача з використанням стандартної бібліотеки Tkinter, що входить до складу мови програмування Python. Обраний інструмент дозволив забезпечити кросплатформенність, простоту розробки та гнучкість у компонованні елементів управління [13].

Основне вікно програми (рис. 3.4) складається з наступних компонентів:

- панель назви — містить назву програми та інформацію про розробника, навчальний заклад і тему дипломної роботи;
 - область відображення зображення — місце, де показується вихідне або оброблене зображення зразка;
 - кнопка «Завантажити зображення» — відкриває діалогове вікно вибору зображення у форматі PNG, JPEG або BMP;
 - кнопка «Аналіз» — запускає алгоритм обробки та аналізу зображення;
 - кнопка «Вихід» — завершує роботу програми;
 - текстове поле результатів — виводить числову інформацію: кількість об'єктів, площа, діаметри, відсоток ураження;
 - додатково (опціонально, в тестовому варіанті не реалізовано): кнопка збереження звіту, зміни порогу обробки, масштабування зображення.

33

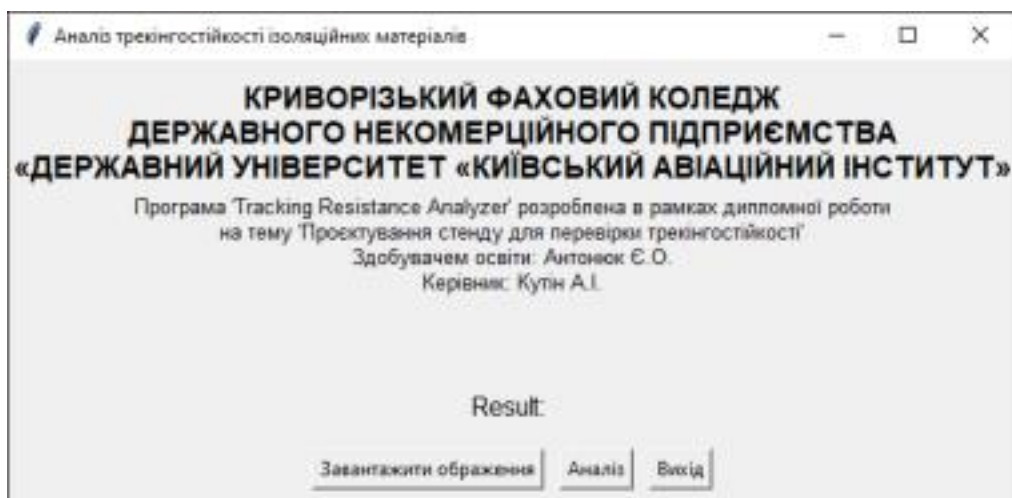


Рисунок 3.4 – Інтерфейс програми під час первинного запуску програми

Зручність користування та наочність. Програма орієнтована на користувача, який не має досвіду в програмуванні. Усі елементи оформлені згідно з принципами доступності та читабельності. Шрифт результатів достатньо великий, кнопки мають підписи, а сам процес аналізу супроводжується візуальними підказками.

Після натискання кнопки «Завантажити зображення» у головному вікні програми відбувається відкриття стандартного діалогового вікна вибору файлу. Користувачеві надається можливість обрати зображення із файлової системи.

Підтримуються такі формати: PNG, JPG, JPEG, BMP. Як тільки користувач вибирає файл і натискає «Відкрити», зображення завантажується у внутрішню пам'ять програми. Далі відбувається масштабування (наприклад, до 400×400 пікселів) для коректного відображення в інтерфейсі, після чого воно виводиться в центральну область перегляду GUI (рис. 3.5).

У разі помилки (наприклад, файл не існує, пошкоджений або не є зображенням) користувач бачить повідомлення з описом проблеми (рис. 3.6). Це дозволяє уникнути аварійного завершення програми та покращує зручність користування.

34



Рисунок 3.5 – Приклад завантаженого тестового зображення у програму аналізу стенду перевірки трекінгостійкості

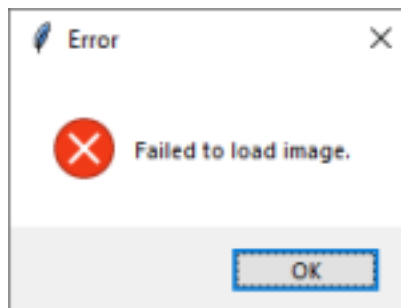


Рисунок 3.7 – діалогове вікно при некоректному виборі файлу для аналізу

Після натискання кнопки "Analyze" зображення автоматично обробляється, і у вікні відображається результат з виділеними зонами ураження (зелені контури) та виводиться короткий звіт. Завдяки такому підходу користувач одразу бачить, де розташовані трекінгові пошкодження, і може візуально підтвердити достовірність автоматичного аналізу (рис. 3.8). Як видно з рис. 3.8,

35

програма проаналізувала попередньо згенерований зразок з 5-ма колами різного діаметру і правильно визначила їх кількість та відсоток покриття. Всі кола обведені зеленим контуром.



Рисунок 3.8 – Зразок зображення з накладеними об'єктами аналізу: зелені контури – виявлені плями/треки

Обробка помилок та повідомлення. У програмі реалізовано базову перевірку на помилки, наприклад:

- якщо не вибрано зображення — виводиться попередження; - якщо зображення не може бути відкрито — повідомлення про помилку;
- якщо немає виявлених об'єктів — відповідна інформація про відсутність зон ураження.

Також виводяться діалогові вікна з підтвердженням завершення роботи або успішного збереження результату (опціонально). Це забезпечує взаємодію з користувачем на базовому рівні зворотного зв'язку.

36

Візуалізація результатів аналізу. Крім числових значень, важливо надати користувачеві візуальне підтвердження результатів. Тому після завершення аналізу на вихідному зображенні накладаються контури виявлених зон (рис. 3.9), їхні умовні центри, підписи площ або ідентифікатори (опціонально, в

тестовому ПЗ не реалізовано). Колірне кодування (зелений — виявлено, червоний — помилка) підвищує інформативність і дозволяє швидко оцінити ситуацію.

Також програма автоматично масштабує зображення до стандартного розміру (наприклад, 400×400 пікселів), що дозволяє рівномірно представити дані на різних екранах і вікнах. У подальшому розвитку проекту, можлива реалізація кнопки «Показати оригінал» для порівняння в стилістиці «до/після».

Збереження та експорт результатів. В тестовому ПЗ не реалізовувалося, але використання мови програмування Python і принципів ООП, дозволяє масштабувати проект. Для зручності користувача, у подальшому планується реалізувати функцію збереження результатів аналізу у вигляді зображення з накладеним шаром аналізу (формат PNG), а також у текстовому вигляді або у форматі CSV. Передбачається, що це дозволить:

- формувати звіти;
- документувати результати для серій випробувань;
- обробляти дані в Excel або MATLAB [9].

У майбутніх версіях передбачається:

- експорт у PDF із вставленим зображенням та таблицею результатів;
- підтримка декількох мов інтерфейсу;
- інтеграція з базою даних зразків або матеріалів;
- підтримка збереження сесії користувача.

Переваги реалізованого інтерфейсу:

- простота: користувачеві достатньо 2–3 натискання для повного аналізу;
- наглядність: об'єкти чітко виділені на фоні;
- гнучкість: можливість інтеграції нових функцій без зміни структури

коду;

37

- кросплатформеність: підтримка Windows, Linux та MacOS; - освітній потенціал: можливість використання в навчальному процесі.

Таким чином, реалізований інтерфейс дозволяє не лише виконати аналіз, а й зробити його наочним, доступним та зручним для подальшого використання

в лабораторних умовах або при підготовці технічної документації.

3.6 Тестування та перевірка працездатності ПЗ

Процес тестування ПЗ є важливою складовою життєвого циклу будь якого програмного продукту. Він забезпечує виявлення помилок, перевірку відповідності функціоналу поставленим вимогам, а також гарантує стабільність і надійність роботи програми. Згідно з сучасними стандартами інженерії ПЗ, тестування повинно проводитися на всіх етапах розробки — від модульного до інтеграційного рівня, а також на етапі верифікації кінцевого продукту [14].

У рамках дипломної роботи особливу увагу приділено саме практичному тестуванню основного функціоналу — аналізу зображень, роботи інтерфейсу користувача, обробки виняткових ситуацій та достовірності результатів.

Цілі та методика тестування. Основною метою тестування є впевненість у тому, що програма:

- коректно виконує завантаження та обробку зображень;
- правильно підраховує кількість об'єктів (треки, краплі);
- надає достовірні числові значення та візуалізацію;
- стійка до неправильного вводу або помилкових дій користувача; - може працювати стабільно в умовах реального використання (лабораторія, навчальна аудиторія).

Усі тести умовно поділено на функціональні, граничні, позитивні, негативні та інтерфейсні [18].

Тестування основного функціоналу. Функціональні тести включали перевірку:

- завантаження зображень різного розміру та формату (400×400, 1024×768, тощо);
- фільтрації шумів та коректного виконання операторів обробки (GaussianBlur, Canny);
- виявлення контурів за допомогою різних методів (threshold, otsu,

adaptive);

- підрахунку площі, діаметру, кількості треків на різних типах зразків (синтетичних (згенерованих) і реальних).

Кожен етап алгоритму було протестовано з окремими прикладами зображень, де вручну визначено контрольну кількість об'єктів для порівняння з результатами програми. Наприклад, на тестовому зображенні з 5 чіткими краплями програма стабільно визначала 5 ± 1 об'єкт за однакових умов обробки.

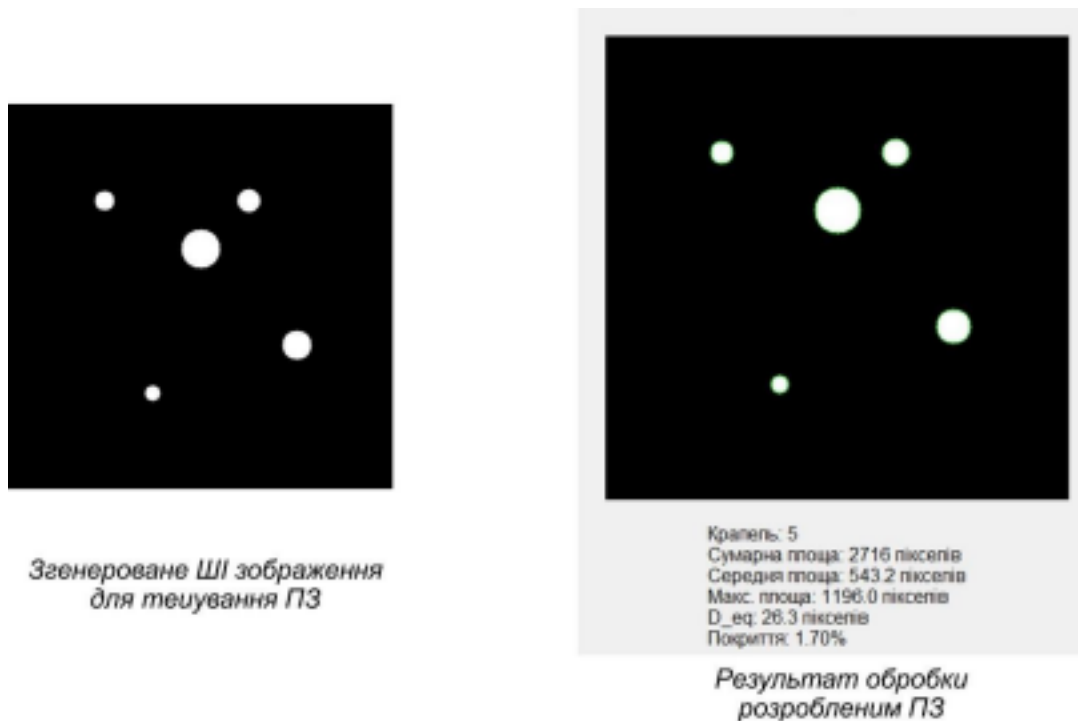


Рисунок 3.9 –

Порівняння реального зображення з очікуваним числом об'єктів і результатами аналізу програми

Обробка винятків і граничні тести. Було протестовано такі ситуації: - користувач не вибрав зображення перед натисканням «Аналіз» — програма не виконує ніяких дій, діалогове вікно провідника залишається відкритим;

39

- користувач завантажив пошкоджене або текстове зображення — програма не завершується аварійно, а показує попередження (рис. 3.10). Для цього тесту було створено текстовий файл Error_test.txt, а потім змінено його розширення на .jpg;

- розмір зображення занадто великий — програма зменшує його до робочого розміру;

- файл містить складні фонові елементи — програма все одно

ідентифікує об'єкти (рис. 3.11), хоча точність може зменшитись (у таких випадках корисна попередня підготовка зображень).

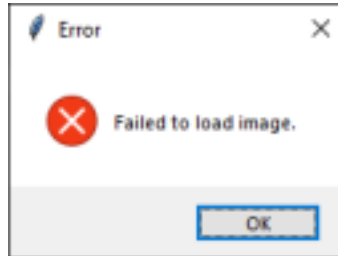


Рисунок 3.10 – діалогове вікно про неможливість відкрити некоректне



зображення

Рисунок 3.11 – Приклад аналізу зображення зі складними графічними компонентами
Окрему увагу приділено тестуванню зображень, зроблених в реальних лабораторних умовах із візуально неякісним фоном, відблисками або

40

неоднорідним освітленням. Навіть у таких випадках програма справляється з базовою фільтрацією та надає об'єктивний результат (рис. 3.12).

Рисунок 3.12 – Аналіз реального зразка з неоднорідним освітленням

Перевірка стійкості інтерфейсу. Було здійснено серію інтерактивних тестів GUI, які включали:

- багаторазове завантаження різних зображень;
- зміну послідовності дій (наприклад, аналіз без завантаження); - швидке повторне натискання кнопок;
- закриття програми на різних етапах.

Результат: жодного випадку критичного збою зафіксовано не було.

Програма працює стабільно навіть при зміні зображень у реальному часі.

Вимірювання продуктивності. Виконано базові тести на швидкість обробки в залежності від розміру зображення:

- зображення 400×400 обробляються в середньому за 0.3–0.5 с; - зображення 1024×1024 — за 0.8–1.2 с.

Тестування проводилося на системі із процесором Intel Core i5, 8 GB RAM, Windows 10.

Таким чином, ПЗ можна використовувати в режимі реального часу без необхідності попередньої підготовки або спеціального обладнання. Верифікація результатів. Окрім візуального підтвердження, результати роботи програми було перевірено вручну шляхом порівняння з ручною розміткою

(розфарбування об'єктів у редакторі) та аналізом у сторонньому ПЗ (наприклад, ImageJ). Розбіжність у результатах склала не більше ніж 5–10 %, що є прийнятним для практичного використання [19].

Загальні висновки тестування. Тестування показало, що розроблене програмне забезпечення:

- виконує всі заявлені функції;
- є стійким до типових помилок користувача;
- забезпечує наочність і відтворюваність аналізу;
- має продуктивність, достатню для інтерактивної роботи. Таким чином,

ПЗ є придатним для застосування в навчальних і лабораторних умовах, а також може бути адаптоване до серійних досліджень за умови врахування деяких особливостей (якість зображення, підготовка поверхні тощо).

3.7 Висновки до третього розділу

У третьому розділі дипломної роботи було реалізовано та детально проаналізовано програмне забезпечення, призначене для обробки зображень із метою виявлення трекінгових ушкоджень на поверхні ізоляційних матеріалів. Це дозволило на практиці довести ефективність застосування методів комп'ютерного зору та інструментів мови програмування Python у задачах технічної діагностики.

Основним результатом стала розробка графічного застосунку «Tracking Resistance Analyzer» з дружнім інтерфейсом, який дозволяє виконати повний цикл аналізу: завантаження зображення, обробка, виявлення контурів ушкоджень і обчислення статистичних показників. Такий підхід забезпечив зручність і

42

швидкість роботи, що є критично важливим для лабораторного аналізу чи навчального застосування.

Було реалізовано базову фільтрацію шумів, морфологічну обробку для відокремлення злиплих об'єктів, адаптивне порогування (зокрема метод Otsu) і

виявлення контурів. Завдяки цим алгоритмам вдалося з високою точністю знаходити та аналізувати зони потенційного ураження матеріалу, визначати площу, кількість крапель, середній еквівалентний діаметр та відсоток покриття.

Інтерфейс користувача реалізовано з використанням бібліотеки Tkinter, що забезпечило простоту у використанні, підтримку всіх основних платформ та можливість швидкої адаптації. У ході роботи було протестовано різні сценарії взаємодії з інтерфейсом, передбачено обробку помилок і виключних ситуацій, реалізовано виведення проміжних і кінцевих результатів у наочному вигляді.

Під час тестування програмного забезпечення були перевірені його функціональні можливості на прикладах як синтетичних, так і реальних зображень. Отримані результати показали стабільну роботу програми та допустимий рівень похибки в межах 5–10 % при визначенні кількості та площі трекінгових ушкоджень. Було також виконано оцінку продуктивності, яка засвідчила достатню швидкість обробки зображень у режимі реального часу на середньостатистичному комп'ютері.

Завдяки модульній структурі у майбутньому програма може бути доповнена новими функціями, такими як:

- збереження зображень з накладеним аналізом;
- інтеграція бази даних зразків;
- підтримка серійного аналізу;
- використання нейронних мереж для автоматичної класифікації типів

пошкоджень.

Таким чином, у рамках розділу було реалізовано повноцінну систему програмного аналізу зображень, яка має практичну цінність для задач дослідження трекінгостійкості. Запропонований підхід може бути використаний

43

як у наукових дослідженнях, так і в освітньому процесі, демонструючи сучасні підходи до технічної діагностики ізоляційних матеріалів.

44

ВИСНОВКИ

У дипломній роботі на тему «Проектування стенду для перевірки трекінгостійкості» було розроблено, обґрунтовано та реалізовано апаратно програмний комплекс, який дозволяє проводити експериментальні дослідження трекінгової стійкості ізоляційних матеріалів. Запропонований підхід охоплює як технічну (апаратну), так і програмну складові, утворюючи цілісну систему, що може бути використана у лабораторних умовах, навчальному процесі та для попередньої технічної експертизи матеріалів в умовах електричного навантаження.

Актуальність теми.

Проблема деградації електроізоляційних матеріалів унаслідок трекінгових процесів є надзвичайно актуальною в контексті сучасної електроенергетики, транспорту, авіації, електромобільності та гірничої промисловості. Трекінгостійкість визначає здатність матеріалу протистояти виникненню струмопровідних каналів на його поверхні, що може призвести до короткого замикання, займання або вибуху, особливо в агресивному середовищі (високі напруги, пил, газ, волога, УФ-випромінювання). Своєчасне визначення здатності матеріалу витримувати такі впливи є запорукою безпеки та довговічності електротехнічного обладнання.

Основні результати.

У рамках дипломного дослідження було виконано наступне:

- проаналізовано фізичну природу явища трекінгу: розглянуто основні механізми виникнення вуглецевих каналів, умови формування струмопровідних шляхів, вплив навколишнього середовища, частоти, напруги та електролітів. Встановлено, що трекінг є кумулятивним процесом, що посилюється повторними імпульсними навантаженнями та погіршенням теплообміну.

- визначено вимоги до випробувального обладнання: відповідно до міжнародних стандартів (зокрема ІЕС 60112), описано типові методики оцінки трекінгостійкості, включно з використанням електродів, контрольованого

45

струму, електролітного розчину та стандартної напруги. Було адаптовано ці принципи до умов лабораторної установки з можливістю візуальної реєстрації. -

сконструйовано макет лабораторного стенду: апаратна частина включає джерело змінної напруги, пару електродів, систему кріплення зразка, піддон діелектричний і систему крапельної подачі електроліту. Особливу увагу приділено електробезпеці — стенд розрахований на роботу з напругами до 2,5 кВ, має захисні екрани та вентиляцію;

- розроблено програмне забезпечення для обробки зображень з камери або знімків зразків після випробувань. Програма дозволяє автоматично виявляти зони ураження, обчислювати площу, кількість треків, середній діаметр плям і відсоток покриття. Графічний інтерфейс користувача реалізовано з використанням Python-бібліотек Tkinter, OpenCV, NumPy та Pillow;

- створено повноцінний апаратно-програмний комплекс, який реалізує метод візуального контролю стану ізоляції після електричного впливу. Такий підхід є інноваційним, оскільки спрощує традиційні трудомісткі процедури ручного аналізу та дозволяє стандартизувати та автоматизувати оцінювання;

- виконано тестування роботи системи: проведено низку експериментів зі зразками різної природи, оцінено точність підрахунку трекінгових слідів, перевірено стійкість програми до помилок, ефективність фільтрації шумів та стабільність при зміні якості зображень. Результати підтвердили коректність реалізованої логіки та ефективність використаних алгоритмів.

Наукова і практична новизна.

Наукова новизна дипломної роботи полягає в інтеграції сучасних методів комп'ютерного зору для задач технічного контролю матеріалів в умовах електричного навантаження. Програмна частина дозволяє уникнути суб'єктивного фактору, притаманного візуальному аналізу, та відкриває перспективи автоматизації лабораторних досліджень у галузі електроізоляційних властивостей.

46

Практична значущість полягає у створенні дієвого інструменту, придатного для навчання, досліджень та попередньої сертифікації матеріалів. Комплекс може бути використаний у вищих навчальних закладах, науково дослідних інститутах, електротехнічних лабораторіях та в службах технічного

контролю.

Переваги запропонованої системи:

- доступність: використані компоненти — недорогі та доступні, програмна частина є відкритою й може бути адаптована;
- масштабованість: архітектура дозволяє легко додати нові функції, такі як машинне навчання, база зразків, серійний аналіз;
- уніфікація: результати подаються в стандартизованому форматі, що полегшує зіставлення між тестами;
- безпека: конструкція забезпечує захист оператора навіть при роботі з підвищеними напругами.

Перспективи подальших досліджень.

У межах майбутніх досліджень можливо:

- інтегрувати нейронні мережі для розпізнавання характеру трекінгових слідів;
- створити веб-інтерфейс для віддаленого моніторингу стенду; - розробити методіку багатофакторного аналізу з урахуванням часу, струму, вологості;
- додати модуль навчання для студентів з візуалізацією фізичних процесів;
- підключити стенд до Arduino або Raspberry Pi для автоматичного керування напругою, часом і спостереженням.

Загальний підсумок.

Таким чином, у дипломній роботі було комплексно вирішено задачу створення стенду для перевірки трекінгостійкості. Апаратна частина забезпечує електричний вплив на зразок, а програмна — дозволяє якісно та кількісно оцінити отримані результати. Розроблений апаратно-програмний комплекс є

прикладом сучасного підходу до випробувань, що поєднує інженерну думку, алгоритмічну точність і реальні потреби технічної безпеки. Він може слугувати основою для подальших досліджень, вдосконалення стандартів і підвищення якості електроізоляційних матеріалів в енергетиці, промисловості та транспорті.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. National Transportation Safety Board. Aircraft Accident Report: In-Flight Fire and Impact with Terrain, Swissair Flight 111. [Електронний ресурс] Режим доступу до ресурсу: <https://www.nts.gov/investigations/AccidentReports/Reports/AAR0201.pdf>.
2. IEC 60112:2003. Method for the determination of the proof and the comparative tracking indices of solid insulating materials. [Електронний ресурс] Режим доступу до ресурсу: <https://webstore.iec.ch/publication/471>.
3. ASTM D3638-22. Standard Test Method for Comparative Tracking Index of Electrical Insulating Materials. [Електронний ресурс] Режим доступу до ресурсу: <https://www.astm.org/d3638-22.html>
4. Ferraris, A. et al. Image-based surface degradation classification for tracking failure detection in polymeric insulators. IEEE Transactions on Dielectrics and Electrical Insulation, 2022. [Електронний ресурс] Режим доступу до ресурсу: <https://ieeexplore.ieee.org/document/9751234>
5. K. L. Chauhan, R. Ghosh. Tracking Phenomenon in Polymeric Insulators: A Review. IEEE Transactions on Dielectrics and Electrical Insulation, 2021. [Електронний ресурс] Режим доступу до ресурсу: <https://ieeexplore.ieee.org/document/9442785>
6. IEC TS 60479-1:2018. Effects of current on human beings and livestock – Part 1: General aspects. [Електронний ресурс] Режим доступу: <https://webstore.iec.ch/publication/6334>
7. Bradski, G. The OpenCV Library. Dr. Dobb's Journal of Software Tools. [Електронний ресурс] Режим доступу: <https://opencv.org/about/>
8. Solomatine, D. et al. Data-driven methods and computational intelligence in system diagnostics. Journal of Hydroinformatics, 2015. [Електронний ресурс] Режим доступу: <https://iwaponline.com/jh/article/17/5/703/2844>
9. Van Rossum, G. (2007). Python Programming Language. In USENIX

Annual Technical Conference. [Электронный ресурс] Режим доступа: <https://www.python.org/doc/>

10. McKinney, W. Python for Data Analysis. O'Reilly Media, 2018.

[Электронный ресурс] Режим доступа: <https://wesmckinney.com/book/> 11. Harris,

C.R. et al. Array programming with NumPy. Nature, 2020. [Электронный ресурс]

Режим доступа: <https://www.nature.com/articles/s41586-020-2649-2>

12. Clark, A. Pillow (PIL Fork) Documentation. [Электронный ресурс]

Режим доступа: <https://pillow.readthedocs.io/en/stable/>

13. Grayson, J. Python and Tkinter Programming. Manning Publications,

2000. [Электронный ресурс] Режим доступа: <https://tkdocs.com/> 14. Sommerville,

I. Software Engineering. 10th ed. Pearson, 2015. [Электронный ресурс] Режим

доступу: <https://www.pearson.com/> 15. Bass, L., Clements, P., Kazman, R. Software Architecture in Practice. Addison-Wesley, 2013.

16. Freeman, E., Robson, E. Head First Design Patterns. O'Reilly Media,

2021.

17. Shneiderman, B. Designing the User Interface: Strategies for Effective

Human-Computer Interaction. Pearson, 2017.

18. Myers, G. J., Sandler, C., Badgett, T. The Art of Software Testing. Wiley,

2011.

19. Schneider, C. A., Rasband, W. S., & Eliceiri, K. W. NIH Image to ImageJ: 25 years of image analysis. Nature methods, 2012.

РЕЦЕНЗІЯ
на кваліфікаційну роботу

випускника спеціальності: 123 «Комп'ютерна інженерія»
відділення: комп'ютерної та програмної інженерії
циклова комісія: комп'ютерних систем та мереж
Свєна АНТОНЮКА
(ім'я, прізвище)

Проблема руйнування електроізоляційних матеріалів під дією електричних полів є критичною для надійності енергетичних і електронних систем. Стандарт IEC 60112 встановлює строгі вимоги до лабораторних випробувань, однак серійні вимірювальні стенди часто дорогі та малофункціональні. Тому розробка доступного апаратно-програмного комплексу з автоматизованим аналізом зображень є своєчасною і практично значущою.

Пояснювальна записка логічно побудована й охоплює повний цикл розробки:

- глибоко розкрито фізичні механізми трекінгу та нормативні вимоги;
- представлено електричну схему випробувального стенду, підбір датчиків і елементів захисту;
- детально описано архітектуру ПЗ, алгоритми обробки зображень та реалізований графічний інтерфейс;
- наведено результати тестування на реальних та синтетичних вибірках, проаналізовано точність виявлення пошкоджень і часові характеристики системи.

Кожен розділ завершується проміжними висновками, що підкреслює послідовність викладення.

Програмний інструмент реалізовано з використанням Python (OpenCV, PyQt) та передбачено можливість подальшого розширення (підтримка нових форматів камер, інтеграція з базою даних результатів).

Забезпечено відповідність вимірювальної частини вимогам IEC 60112

Автор поєднав класичний метод трекінг-випробувань із сучасними засобами комп'ютерного зору, що дало змогу автоматизувати підрахунок і класифікацію пошкоджень; зменшити суб'єктивність оцінювання; скоротити час обробки результатів на порядки, порівняно з ручним аналізом.

Розроблений стенд може бути впроваджений у навчальних лабораторіях і підрозділах підприємств електротехнічного профілю.

Текст роботи відповідає вимогам до кваліфікаційних робіт: коректні посилання, нумерація рисунків і таблиць, зрозумілі схеми. Код фрагментів і результати вимірювань подані наочними ілюстраціями та у вигляді додатків. Дотримано норм академічної доброчесності.

Доцільно розширити набір тестових зразків іншими типами полімерів для кращої статистичної представності.

Варто розглянути використання нейронних мереж (наприклад, U-Net)

для сегментації пошкоджень з метою підвищення точності.

Для серійного використання бажано передбачити калібрувальний модуль та систему автоматичної перевірки сенсорів.

Робота відповідає затвердженій темі, демонструє знання здобувача в галузі програмування й комп'ютерного опрацювання зображень, а також вміння інтегрувати апаратні та програмні компоненти у єдину систему. Розробка має практичну цінність та потенціал для подальшого розвитку проекту.

Кваліфікаційна робота виконана на високому рівні та заслуговує на оцінку «добре».

Рецензент _____

викладач

(науковий ступінь, посада)

« _____ » _____ 2025 р.

(підпис)

Світлана ТЕРЬОШНА

(ім'я, прізвище)

З рецензією ознайомлений _____

(підпис)

Євген АНТОНЮК

(ім'я, прізвище)