

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Автоматики і комп'ютеризованих технологій

(повна назва)

Кафедра Комп'ютерно-інтегрованих технологій, автоматизації та робототехніки

(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА Пояснювальна записка

рівень вищої освіти другий (магістерський)

Удосконалення автоматизованої технології обробки зображень топології
друкованих плат за допомогою систем технічного зору

(тема)

Виконав:

здобувач 2 року навчання,
групи КТРСм-23-1

Ахмад А.М.

(прізвище, ініціали)

Спеціальність 174 Автоматизація та
комп'ютерно-інтегровані технології

(код і повна назва спеціальності)

Тип програми освітньо-професійна

(освітньо-професійна або освітньо-наукова)

Освітня програма Комп'ютеризовані та
робототехнічні системи

(повна назва освітньої програми)

Керівник проф. Косенко В. В.

(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри

(підпис)

Невлюдов І.Ш.

(прізвище, ініціали)

2025 р.

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

Факультет _____ АКТ _____
 Кафедра _____ КІТАР _____
 Рівень вищої освіти _____ другий (магістерський) _____
 Спеціальність 174 Автоматизація та комп'ютерно-інтегровані технології _____
 Тип програм _____ освітньо-професійна _____
 Освітня програма Комп'ютеризовані та робототехнічні системи _____
 (шифр і назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри КІТАР _____ (підпис)

«30» серпня 2024 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві Ахмаду Алі Монзеру _____
 (прізвище, ім'я, по батькові)

1. Тема роботи Удосконалення автоматизованої технології обробки зображень топології друкованих плат за допомогою систем технічного зору

Затверджена наказом по університету від 25.11.2024 р. № 1239 Ст

2. Термін подання студентом роботи до екзаменаційної комісії 23.01.2025 р

3. Вихідні дані до роботи: _____

3.1 Аналіз існуючих систем автоматизованого контролю друкованих плат;

3.2 Використання алгоритмів машинного навчання для обробки зображень;

3.3 Дослідження систем технічного зору для виявлення дефектів;

3.4 Проведення досліджень впливу порогу бінаризації зображення на знаходження елементів плати.

4. Перелік питань, що потрібно опрацювати в роботі:

4.1 Вступ;

4.2 Аналіз предметної області;

4.3 Опис методів обробки зображень;

4.4 Проведення експериментальних досліджень;

4.5 Охорона праці;

4.6 Висновки.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів) Демонстраційний матеріал у вигляді презентації формату PowerPoint (*.ppt) 15 с. формату А4

6. Консультанти розділів роботи

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Отримання завдання	01.09.2024	виконано
2	Збір матеріалів для дослідження	25.09.2024	виконано
3	Розробка розділу «Аналіз предметної області»	09.10.2024	виконано
4	Розробка розділу «Опис макету для дослідження»	15.10.2024	виконано
5	Розробка розділу «Проведення експериментальних досліджень»	19.12.2024	виконано
6	Розробка розділу Охорона праці	30.12.2023	виконано
7	Оформлення кваліфікаційної роботи	06.01.2024	

Дата видачі завдання 01.09.2024

Здобувач _____ Ахмад А.М.
(підпис) (прізвище, ініціали)

Керівник роботи _____ проф. Косенко В. В.
(підпис) (посада, прізвище, ініціали)

Я, як студент ХНУРЕ, розумію і підтримую політику закладу із академічної доброчесності. Я не надавав і не одержував недозовлену допомогу під час підготовки кваліфікаційної роботи. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело.

Дата
«23» січня 2025 р.



ПІБ
Ахмад А.М.

РЕФЕРАТ

Пояснювальна записка: 92 с., 2 табл., 53 рис., 3 дод., 16 джерел.

ТЕХНІЧНИЙ ЗІР, ОБРОБКА ЗОБРАЖЕНЬ, БІНАРИЗАЦІЯ, ДРУКОВАНІ ПЛАТИ, КЛЮЧОВІ ТОЧКИ.

Мета роботи – удосконалення автоматизованої технології обробки зображень для підвищення точності і швидкості виявлення дефектів топології друкованих плат.

Об'єкт розробки – система, призначена для автоматизованого аналізу топології друкованих плат на основі зображень, отриманих з використанням систем технічного зору.

Предмет розробки – технології обробки зображень для виявлення дефектів і аналізу топологічних особливостей друкованих плат.

Методи розробки – аналіз проблемної області, визначення вимог до систем технічного зору, розробка алгоритмів обробки зображень, що включають фільтрацію, сегментацію та класифікацію, застосування методів машинного навчання для розпізнавання дефектів топології друкованих плат.

Взаємозв'язок з іншими розробками – розроблена технологія тісно пов'язана з іншими сучасними системами автоматизованого контролю якості, які використовуються у виробництві електронних компонентів. Вона може бути інтегрована з промисловими системами комп'ютерного зору для підвищення точності та надійності контролю якості друкованих плат. У результаті виконання роботи було розроблено технологію автоматизованої обробки зображень топології друкованих плат, яка інтегрується з системами технічного зору та дозволяє ефективно виявляти дефекти та аномалії топології.

Також, отримані результати роботи можна віднести до "Цілі сталого розвитку" 9 "Промисловість, інновації та інфраструктура", а саме п.9.5 "Підтримувати наукові дослідження та інновації, що сприяють промислового розвитку".

ABSTRACT

Explanatory note: 92 p., 2 tabl., 53 pic., 3 applications, 16 sources.

TECHNICAL VISION, IMAGE PROCESSING, BINARIZATION, PRINTED BOARDS, KEY POINTS.

Objective of the work – to improve the automated image processing technology to increase the accuracy and speed of detecting defects in printed circuit board topology.

Object of development – a system designed for automated analysis of printed circuit board topology based on images obtained using machine vision systems.

Subject of development – image processing technologies for defect detection and analysis of topological features of printed circuit boards.

Development methods – problem area analysis, defining the requirements for machine vision systems, development of image processing algorithms including filtering, segmentation, and classification, application of machine learning methods for defect recognition in printed circuit board topology.

Relationship with other developments – the developed technology is closely related to other modern automated quality control systems used in the manufacturing of electronic components. It can be integrated with industrial machine vision systems to improve the accuracy and reliability of quality control for printed circuit boards. As a result of the work, a technology for automated image processing of printed circuit board topology was developed. This technology integrates with machine vision systems and enables effective detection of defects and anomalies in the topology.

Also, the results of the work can be attributed to "Sustainable Development Goal" 9 "Industry, Innovation and Infrastructure", namely item 9.5 "Support scientific research and innovation that contribute to industrial development".

ЗМІСТ

Перелік умовних скорочень.....	10
Вступ.....	11
1 Аналіз предметної області.....	13
1.1 Аналіз друкованих плат.....	13
1.1.1 Види друкованих плат	14
1.1.2 Параметри друкованих плат.....	19
1.2 Технології виготовлення друкованих плат.....	23
1.3 Дефекти друкованих плат.....	26
1.4 Висновки до першого розділу.....	30
2 Методи обробки зображень.....	31
2.1 Фільтрація зображень.....	31
2.2 Зменшення шумів.....	34
2.3 Вибір фільтрів для обробки зображень.....	35
2.4 Бінаризація зображень.....	38
2.5 Методи вирівнювання зображень.....	43
2.6 Віднімання та накладання зображень	46
2.7 Розрахунок передавальних функцій та складання структурної схеми системи технічного зору.....	47
2.8 Висновки до другого розділу.....	51
3 Розробка програмного забезпечення для автоматизованого аналізу плат.....	53
3.1 Бібліотеки.....	53
3.1.1 OpenCV.....	53
3.1.2 NumPy.....	54
3.1.3 Tkinter.....	54
3.1.4 Pillow.....	55
3.2 Ключові методи.....	56
3.2.1 SIFT.....	56
3.2.2 FLANN.....	57

3.2.3	Метод обчислення гомографії.....	58
3.2.4	Метод бінарізації.....	59
3.2.5	Морфологічні операції.....	60
3.2.6	Обчислення різниці зображень.....	60
3.2.7	Аналіз контурів.....	61
3.3	Огляд програми.....	62
3.3.1	Load_image(label)	63
3.3.2	Load_img(path)	63
3.3.3	Load_images().....	63
3.3.4	Preprocess_image(image_path)	64
3.3.5	Align_images(img1, img2)	64
3.3.6	Apply_align(img1, img2, M)	65
3.3.7	Process_images().....	66
3.4	Експериментальне дослідження та порівняння.....	67
3.5	Висновки до третього розділу.....	73
4	Охорона праці.....	74
4.1	Безпека при виготовленні друкованих плат.....	74
4.2	Висновки до четвертого розділу.....	75
	Висновки.....	77
	Перелік джерел посилання.....	78
	Додаток А Лістинг коду.....	80
	Додаток Б Апробація наукових результатів досліджень.....	84
	Додаток В Демонстраційний матеріал.....	91

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

ДП – друковані плати;

СТЗ – системи технічного зору;

АОІ – automated optical inspection;

FLANN – fast library for approximate nearest neighbors;

PCB – printed circuit board;

PIL – Python imaging library;

RANSAC – random sample consensus;

SIFT – scale-invariant feature transform.

ВСТУП

Сучасний розвиток інформаційних технологій та автоматизованих систем контролю якості активно впливає на виробничі процеси, зокрема в електронній промисловості. Автоматизація процесів контролю друкованих плат (ДП) є важливою складовою підвищення надійності та якості електронної продукції, особливо з огляду на зростання вимог до мініатюризації та складності електронних компонентів. Застосування систем технічного зору для автоматичного аналізу топології друкованих плат дозволяє значно зменшити вплив людського фактора, знизити ймовірність помилок і прискорити виявлення дефектів.

Одним із ключових аспектів такого контролю є вдосконалення технології обробки зображень, що дозволяє з високою точністю виявляти різноманітні дефекти друкованих плат, такі як порушення ширини провідників, дефекти пайки, деформації на вигинах тощо. Використання алгоритмів машинного навчання та методів обробки зображень в системах технічного зору сприяє підвищенню точності і швидкості аналізу, що робить дослідження в цьому напрямку актуальним і необхідним.

Метою даної кваліфікаційної роботи є удосконалення автоматизованої технології обробки зображень для виявлення дефектів друкованих плат. У рамках роботи буде досліджено методи обробки зображень, алгоритми сегментації та класифікації, що дозволяють досягти високої точності ідентифікації дефектів у реальних виробничих умовах.

У роботі буде розглянута розробка та удосконалення програмного засобу для системи технічного зору для контролю відхилень та дефектів друкованих плат. В експериментальній частині роботи буде проведено порівняння значень порогу бінаризації на знаходження дефектів в топології друкованих плат.

Мета роботи – удосконалення автоматизованої технології обробки зображень для підвищення точності і швидкості виявлення дефектів топології друкованих плат.

Об'єкт розробки – система, призначена для автоматизованого аналізу топології друкованих плат на основі зображень, отриманих з використанням систем технічного зору.

Предмет розробки – технології обробки зображень для виявлення дефектів і аналізу топологічних особливостей друкованих плат.

Для досягнення поставленої мети необхідно вирішити такі завдання:

- виконати аналіз видів друкованих плат та їх дефектів;
- розглянути методи обробки зображень;
- розглянути середовище та бібліотеки для розробки програмного засобу;
- розробити програмне забезпечення;
- провести дослідження впливу порогу бінарізації зображення на знаходження різниці між еталонним зображенням топології друкованої плати та виготовленою топологічною структурою;

Пояснювальну записку виконано згідно з [1 – 3]. За результатами проведених досліджень опублікована теза доповідь в студентському збірнику «Виробництво і мехатронні системи 2024» [4].

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Аналіз друкованих плат

Друковані плати (ДП) є основою сучасної електроніки, забезпечуючи з'єднання між електронними компонентами та дозволяючи компактне розміщення складних схем. Основна структура друкованої плати включає шар провідного матеріалу, зазвичай міді, нанесений на ізоляційну основу, а також захисні покриття для ізоляції провідників і захисту від зовнішніх впливів. Залежно від застосування, конструкція і параметри друкованих плат можуть значно відрізнятися, що визначає їхні основні види та характеристики.

Основні види друкованих плат включають одношарові, двошарові та багатошарові. Одношарові плати, з провідниками лише з одного боку, є найбільш простими у виготовленні та використовуються в пристроях з обмеженою функціональністю, таких як побутова електроніка. Двошарові плати мають провідники з обох боків і дозволяють розміщувати більшу кількість компонентів, забезпечуючи ефективне використання простору. Багатошарові плати, що складаються з кількох провідних і ізоляційних шарів, застосовуються в складних пристроях з високою щільністю компонентів, таких як комп'ютери та комунікаційне обладнання. Багатошарові конструкції забезпечують зниження перешкод і мінімальні затримки сигналу, що важливо для високошвидкісних схем.

Параметри друкованих плат, такі як ширина доріжок, відстань між провідниками, кількість шарів і товщина покриття, впливають на їхні електричні та теплові характеристики, що важливо для забезпечення надійної роботи плати. Технології виробництва включають методи травлення, ламінування та нанесення паяльних масок, що дозволяє досягати потрібної точності й захисту від зовнішніх факторів. Залежно від вибраної технології виготовлення, друковані плати

можуть мати різні властивості, що визначає їхню стійкість до механічних і термічних навантажень, а також загальну довговічність.

Аналіз друкованих плат і розуміння їхніх особливостей важливі для розробки автоматизованих систем контролю якості, здатних виявляти дефекти та невідповідності у виготовлених платах. Завдяки автоматизації процесів перевірки якості можна досягти вищої точності, зменшити кількість браку та забезпечити стабільну якість продукції в масштабах серійного виробництва.

1.1.1 Види друкованих плат

Друковані плати (ДП) розрізняють за кількома основними параметрами, такими як кількість шарів, функціональність, гнучкість, щільність компонентів і призначення. Основні типи друкованих плат включають одношарові, двошарові, багатошарові, гнучкі, гнучко-жорсткі, а також спеціалізовані плати для високочастотних і високо потужних застосувань. Кожен тип ДП має свої особливості, які впливають на технологію виготовлення, сферу використання та ефективність роботи пристрою. Далі розглянемо основні види друкованих плат, рисунок 1.1.

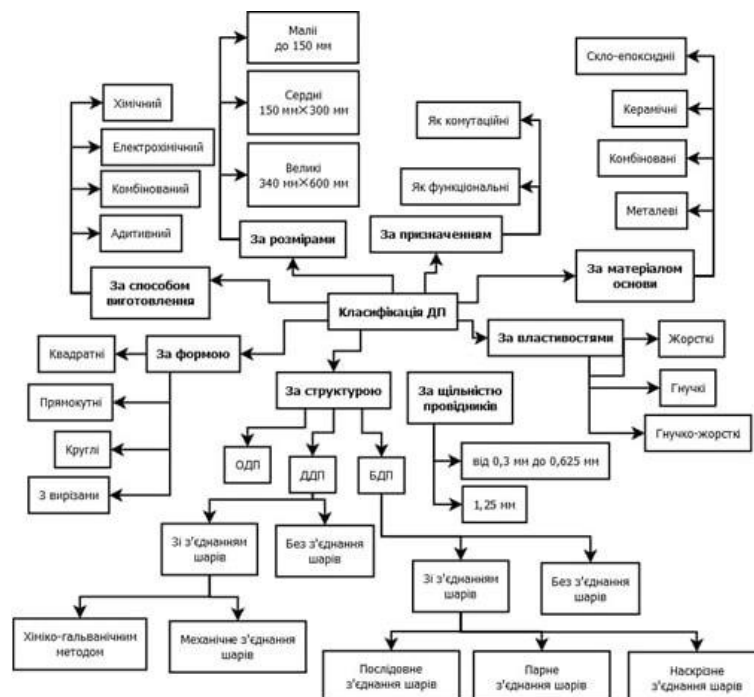


Рисунок 1.1 – Класифікація друкованих плат

Одношарові плати є найпростішим типом друкованих плат. У цьому випадку електричні доріжки та компоненти розташовуються тільки на одній стороні плати, що знижує складність конструкції, рисунок 1.2. Зазвичай основа плати виготовляється з міцного матеріалу, наприклад, FR4 (склотекстоліт), який забезпечує необхідну механічну стійкість. На цей матеріал наноситься тонкий шар міді, з якого формуються доріжки. Поверх доріжок може наноситися захисний шар паяльної маски для запобігання окисленню та коротким замиканням.



Рисунок 1.2 – Одношарова друкована плата

Одношарові плати використовуються в пристроях з низьким рівнем складності, таких як радіоприймачі, побутова електроніка, світлодіодні панелі та прості електронні іграшки. Їх перевагами є низька вартість виробництва та простота монтажу компонентів, але вони обмежені за функціональністю та не підходять для високонавантажених схем.

Двошарові плати мають провідні доріжки з обох боків основи, що дозволяє розміщувати більше компонентів і забезпечувати більш складне електричне з'єднання. У такій конструкції використовуються металізовані отвори (vias), які з'єднують провідники на обох сторонах плати, забезпечуючи комплексні електричні схеми. Це особливо важливо для підвищення щільності розміщення та оптимізації шляхів проходження сигналів, рисунок 1.3.

Двошарова друкована плата



Рисунок 1.3 – Двошарова друкована плата

Двошарові плати мають ширше застосування, ніж одношарові, і використовуються в побутовій техніці, автомобільній електроніці, аудіо- і відео обладнанні. Вони дозволяють знизити розміри пристроїв і підвищити продуктивність за рахунок покращеної електричної ізоляції та зменшення довжини доріжок.

Багатошарові друковані плати складаються з трьох і більше шарів, які чергуються між шарами ізоляції, забезпечуючи високу щільність з'єднань. Ці плати можуть включати до 40 і більше шарів, що дозволяє використовувати їх в особливо складних і високошвидкісних пристроях, рисунок 1.4. Виготовлення багатошарових плат потребує високої точності та спеціальних технологій, зокрема ламінування, для об'єднання шарів. Багатошарові плати часто використовуються в комп'ютерах, мережевому обладнанні, медичній техніці та авіакосмічній галузі.

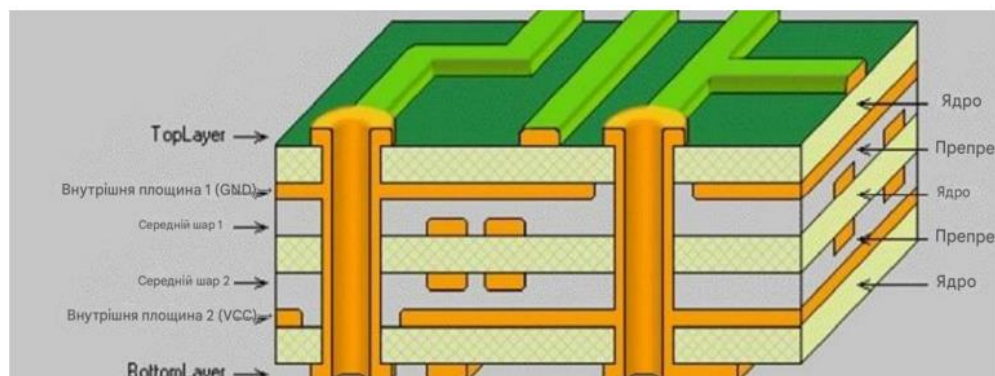


Рисунок 1.4 – Багатошарова друкована плата

Багатошарові плати дозволяють знизити рівень електричних перешкод і покращити стабільність сигналу, що є критичним для високочастотних і високошвидкісних схем. Проте такі плати є дорогими у виробництві та вимагають більш складного контролю якості.

Гнучкі плати виготовляються на основі полімерних матеріалів, таких як поліімід, який надає їм гнучкості та дозволяє згинатися, зберігаючи при цьому працездатність. Ці плати можуть згинатися і розтягуватися без пошкодження доріжок, що робить їх незамінними в пристроях з обмеженим простором або нестандартною формою, рисунок 1.5.

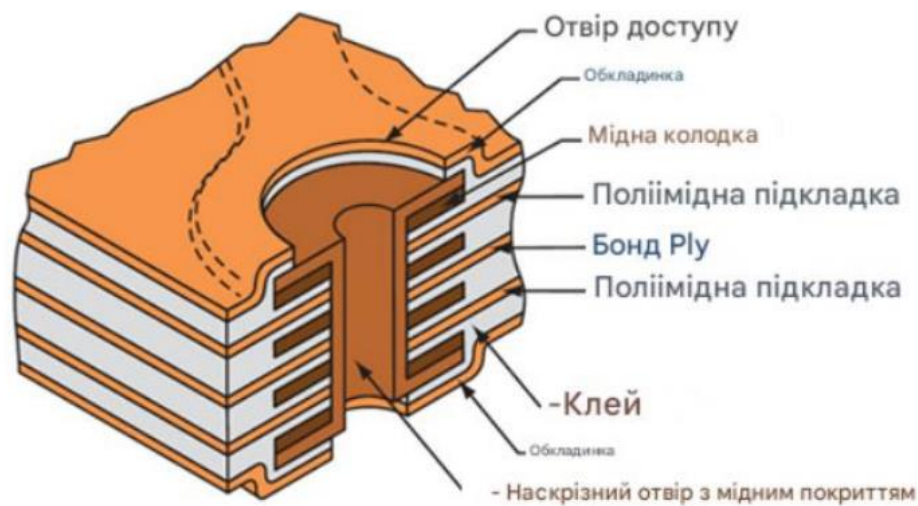


Рисунок 1.5 – Гнучка друкована плата

Гнучкі плати знаходять застосування в мобільних телефонах, камерах, медичному обладнанні та інших пристроях, де важливий гнучкий форм-фактор. Їхні переваги включають зменшення ваги і розміру пристрою, але вони дорожчі у виробництві і потребують спеціального обладнання для монтажу.

Гнучко-жорсткі плати поєднують у собі характеристики гнучких і жорстких плат, рисунок 1.6. Вони складаються з жорсткої основи, яка забезпечує стабільну механічну підтримку, і гнучких частин, що дозволяють підключати компоненти під різними кутами або у важкодоступних місцях. Такі плати широко використовуються в авіакосмічній, автомобільній і медичній галузях, де потрібні надійність та компактність.

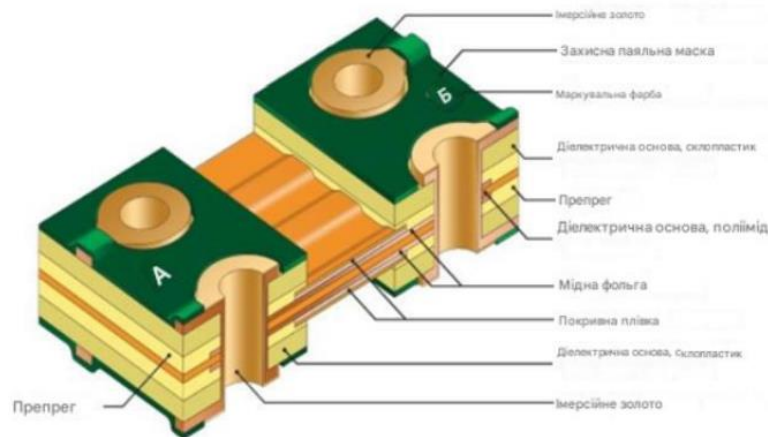


Рисунок 1.6 – Гнучко-жорстка друкована плата

Гнучко-жорсткі плати забезпечують меншу кількість з'єднань і підвищену надійність, оскільки вони дозволяють уникати роз'ємів і з'єднувальних кабелів. Однак, вони мають високу вартість і складний процес виготовлення.

Для передачі високочастотних сигналів або роботи з великими струмами застосовуються спеціальні друковані плати, які виготовляються з матеріалів із низькими втратами сигналу, таких як тефлон або інші діелектрики з низькою діелектричною проникністю. Такі плати часто використовуються в телекомунікаційному обладнанні, радіочастотних пристроях та в обладнанні для передачі даних на великі відстані, рисунок 1.7.

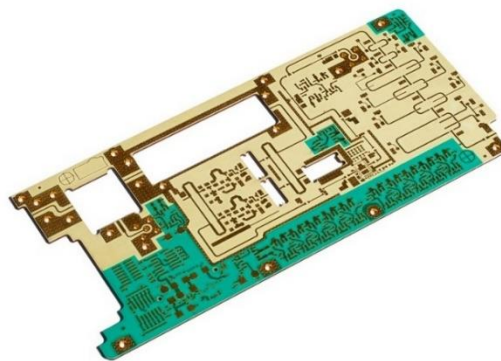


Рисунок 1.7 – Високочастотна друкована плата

Такі плати забезпечують мінімальні втрати сигналу і стабільність параметрів навіть при високих частотах. Їх виготовлення потребує спеціальних матеріалів і точного контролю товщини доріжок, що впливає на кінцеву вартість виробу.

1.1.2 Параметри друкованих плат

Параметри друкованих плат є критичними характеристиками, які визначають електричні, механічні й теплові властивості готової плати. Від точності виконання цих параметрів залежить надійність, продуктивність і термін служби пристрою, до складу якого входить плата. Основні параметри друкованих плат включають товщину та матеріал підкладки, товщину шару міді, ширину провідників, відстань між провідниками, розмір і тип отворів, кількість шарів і наявність захисних покриттів. Нижче розглянемо кожен з параметрів детально.

Підкладка, зазвичай виготовлена з матеріалу FR4 (склотекстоліту) або аналогічних композитних матеріалів, є основою плати, на якій формуються провідники. Товщина підкладки визначає механічну стійкість та міцність плати, впливаючи на її здатність витримувати механічні навантаження і температурні зміни. Типова товщина підкладки варіюється від 0,2 мм для мініатюрних плат до 1,6 мм для стандартних застосувань, хоча можуть бути використані й інші значення залежно від специфічних вимог.

Товщина мідного шару визначає здатність провідників пропускати струм і розсіювати тепло. Зазвичай товщина міді вимірюється в унціях на квадратний фут, де 1 унція на квадратний фут еквівалентна товщині приблизно 35 мікронів.

Високо струмові застосування можуть вимагати товщини від 2 – 4 унцій, тоді як для стандартних пристроїв зазвичай використовується 1 унція. Товщина міді впливає на опір провідників, а також на стійкість до перегріву, особливо в високонавантажених схемах, рисунок 1.8.

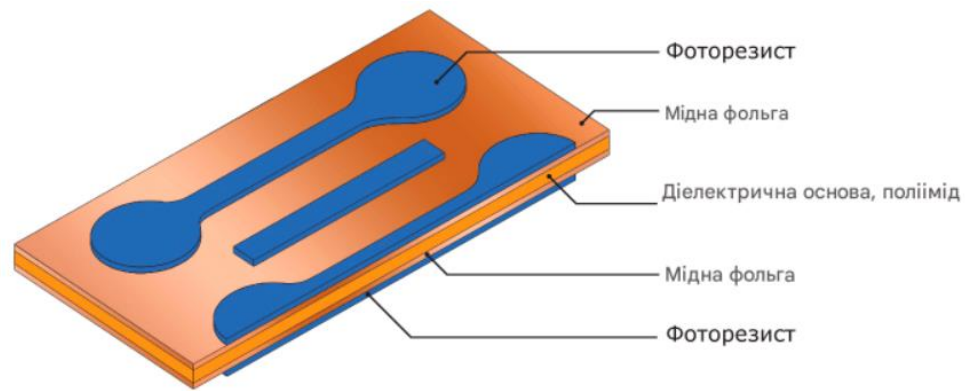


Рисунок 1.8 – Мідний шар друкованої плати

Ширина провідників і відстань між ними є одним із найважливіших параметрів, що визначають щільність плати, а також впливають на здатність плати забезпечувати належні електричні характеристики. Для стандартних застосувань ширина доріжок може становити 0,2 – 0,3 мм, тоді як у високо щільних платах, наприклад у смартфонах, вона може зменшуватися до 0,1 мм або навіть менше. Відстань між доріжками визначається вимогами до ізоляції і залежить від робочої напруги та частоти сигналу, рисунок 1.9.

Товщина мідної фольги	Мінімальне значення (зовнішні шари), милий		Мінімальне значення (внутрішні шари), милий	
	Ширина провідної доріжки	Відстань між провідниками	Ширина провідної доріжки	Відстань між провідниками
5 мкм	50	75	50	50
9 мкм	75	75	65	65
18 мкм	100	100	75	55
35 мкм	150	150	100	110
70 мкм	200	200	150	160
105 мкм	305	305	180	200
140 мкм	350	350	200	250

Рисунок 1.9 – Ширина і відстань між провідниками

Отвори в друкованих платах призначені для монтажу компонентів або для з'єднання шарів у багатошарових платах. Вони можуть бути наскрізними або

глухими. Металізовані перехідні отвори, які з'єднують шари, забезпечують перенесення сигналу між різними рівнями плати. Діаметр отворів зазвичай варіюється від 0,2 мм для мікровіасів до 1 мм для монтажних отворів під більші компоненти, рисунок 1.10.

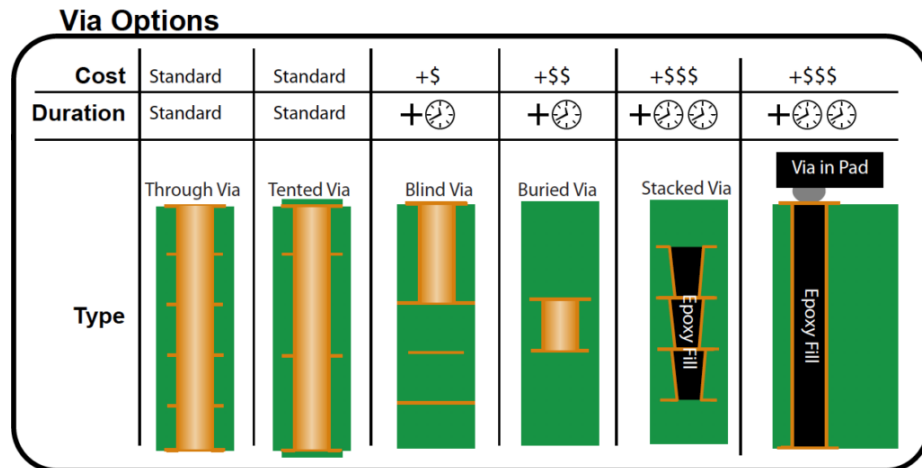


Рисунок 1.10 – Типи отворів

Кількість шарів у друкованій платі є визначальним параметром для її складності та здатності підтримувати багатошарову структуру з численними сигналами, рисунок 1.11. Одношарові та двошарові плати є більш простими, але обмеженими за функціональністю. Багатошарові плати, які можуть мати 4 і більше шарів, дозволяють розміщувати значну кількість компонентів на обмеженій площі, що особливо важливо для високочастотних схем і приладів з високою щільністю.



Рисунок 1.11 – Шари багатошарової друкованої плати

Паяльна маска – це захисний шар, що наноситься на поверхню плати для захисту мідних доріжок від окислення і для запобігання випадковим коротким замиканням під час монтажу компонентів, рисунок 1.12. Паяльна маска також сприяє зручному нанесенню припою, завдяки чому процес пайки стає точнішим. Зазвичай паяльні маски виготовляються зеленого кольору, але існують також червоні, сині, чорні та інші варіанти залежно від естетичних чи виробничих вимог.

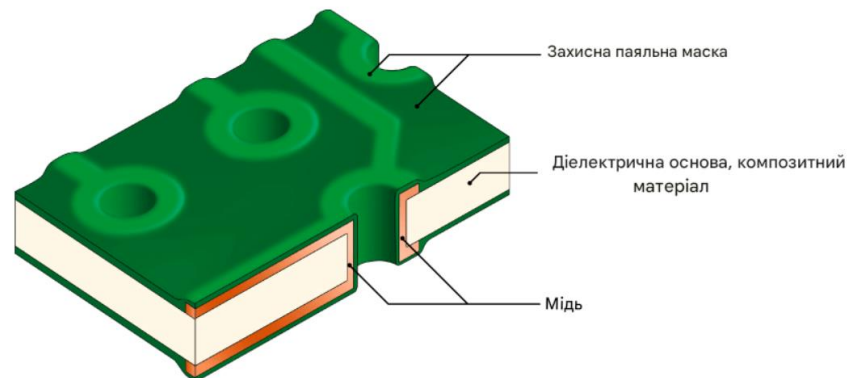


Рисунок 1.12 – Приклад паяльної маски

Для полегшення монтажу та ідентифікації компонентів на платі наноситься маркування, яке включає позначення компонентів (наприклад, резистори, конденсатори) і орієнтири для розташування мікросхем. Це особливо важливо для складних багатошарових плат, де щільність компонентів може бути дуже високою. Маркування допомагає уникнути помилок при монтажі та обслуговуванні плати, рисунок 1.13.



Рисунок 1.13 – Маркування друкованої плати

Для забезпечення надійної роботи у складних умовах експлуатації важливо враховувати температурні характеристики плати, зокрема допустиму робочу температуру та стійкість до теплових навантажень. Плати повинні витримувати теплові цикли без пошкоджень, рисунок 1.14. Багато друкованих плат розроблені для роботи в широкому температурному діапазоні, особливо ті, що використовуються в автомобільній та авіакосмічній галузях.

FR4 Теплові властивості	Значення
Теплопровідність (осі x і y)	~0,9 Вт/(м·К)
Теплопровідність (вісь z)	~0,3 Вт/(м·К)
Коефіцієнт теплового розширення (осі x і y)	~13 ppm/K
Коефіцієнт теплового розширення (вісь z)	~70 ppm/K
Температура склування (Tg)	Від -135 °C до -170 °C
Питома теплоємність	~1100 Дж/(кг·°C)

Рисунок 1.14 – Теплопровідність матеріалу FR4 для виготовлення друкованих плат

1.2 Технології виготовлення друкованих плат

Виготовлення друкованих плат (ДП) – це складний багатостадійний процес, який включає проектування, обробку матеріалів, нанесення провідних доріжок, створення монтажних отворів, нанесення захисних покриттів та контроль якості. З кожним етапом на виробництві працюють технології, що забезпечують високу точність, надійність і відповідність плати технічним вимогам. Усі ці етапи інтегруються у виробничий процес для отримання якісного кінцевого продукту, який задовольняє вимоги сучасної електроніки.

Процес виготовлення друкованих плат починається з проектування електричної схеми. Інженери створюють макет розташування елементів на платі у спеціалізованих програмних засобах (такі як Altium Designer, Eagle, KiCAD), де проектуються схеми з'єднань, доріжок та контактних площадок. Готовий макет перетворюється на файл у форматі Gerber, який містить інструкції для виробничого обладнання і передається на фабрику.

Наступним етапом є підготовка базового матеріалу, зазвичай це склотекстоліт, вкритий шаром міді. Поверхня очищується і готується до нанесення провідного малюнка за допомогою фотолітографії. На плату наноситься тонкий шар фоточутливого матеріалу, який під дією світла змінює свої властивості. Рисунок доріжок наноситься через спеціальну маску, створену відповідно до проектного файлу. Ділянки, що піддалися опроміненню, затверджуються і захищають мідну поверхню під час наступного етапу, рисунок 1.15.

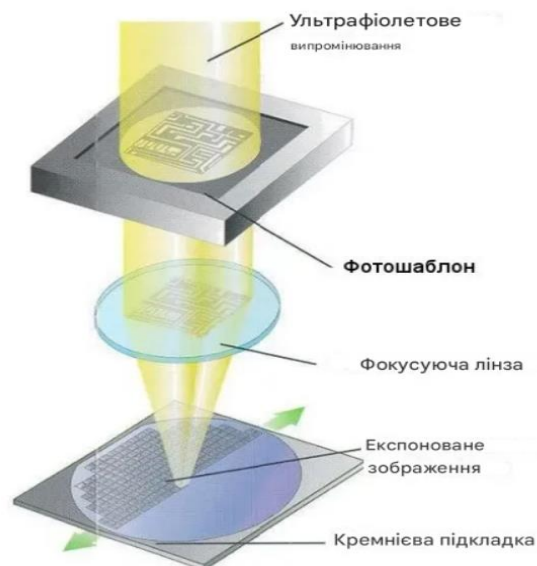


Рисунок 1.15 – Процес фотолітографії

Травлення – це процес видалення зайвої міді, що залишилась на незахищених ділянках. Плата занурюється в хімічний розчин, який реагує з відкритими ділянками міді, видаляючи їх і залишаючи на платі лише потрібний малюнок доріжок, рисунок 1.16. Для травлення найчастіше використовують

розчини на основі персульфату натрію або хлориду заліза. Цей процес дуже важливий, оскільки саме він забезпечує точність створення доріжок і відповідність технологічним нормам.

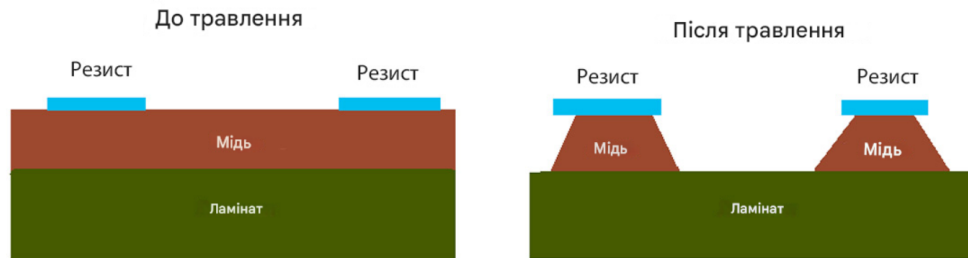


Рисунок 1.16 – Травлення друкованої плати

Після травлення та видалення фоторезисту проводиться свердління отворів для монтажу компонентів. Це завдання виконується високоточним обладнанням, яке забезпечує точність розташування отворів відповідно до проєктного креслення. Після цього важливим етапом є металізація отворів, що забезпечує електричний зв'язок між шарами у багатошарових платах. Для цього плата занурюється у хімічний розчин, у якому іони міді осідають на стінках отворів, утворюючи провідний шар, рисунок 1.17. Цей процес виконується методом хімічного або електрохімічного осадження і є особливо важливим для забезпечення електропровідності між шарами.

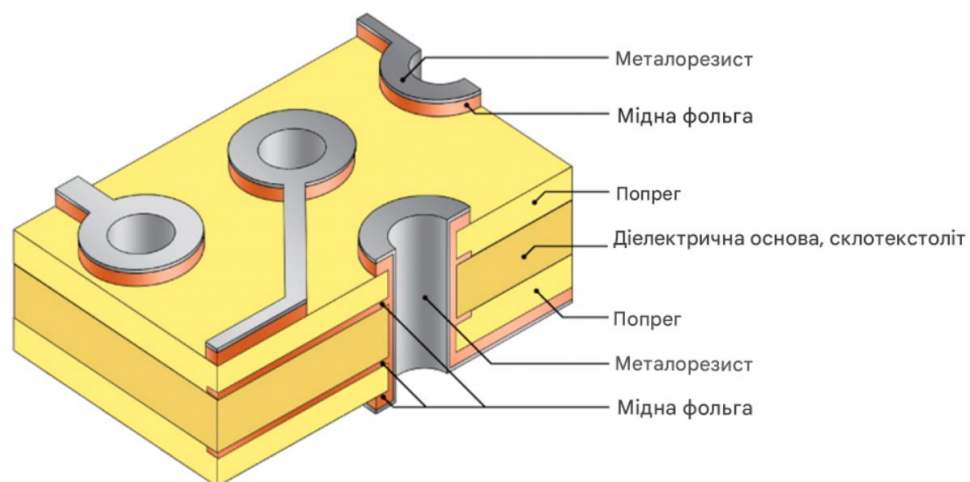


Рисунок 1.17 – Металізація отворів

Щоб захистити доріжки та контактні майданчики від зовнішніх факторів, на поверхню плати наноситься захисне покриття – паяльна маска. Зазвичай вона має зелений колір, хоча можливі й інші варіанти. Паяльна маска ізолює доріжки від можливих замикань і захищає їх від корозії. На контактні площадки наноситься шар олова або нікелю з золотом, що поліпшує паяльні властивості і захищає від окислення.

Останнім етапом є монтаж компонентів на друковану плату. Залежно від виду монтажу, він може виконуватись за допомогою технології поверхневого монтажу (SMT) або наскрізного монтажу (ТНТ). У разі поверхневого монтажу компоненти розміщуються на поверхні плати, і пайка здійснюється шляхом оплавлення у спеціальних печах. При наскрізному монтажі компоненти фіксуються у просвердлених отворах, після чого відбувається пайка за допомогою хвильової паяння.

Фінальний контроль якості передбачає автоматичний оптичний контроль (АОІ), що дозволяє перевірити правильність розміщення доріжок, контактних майданчиків і елементів. За допомогою АОІ також можна виявити дефекти, такі як замикання доріжок, невідповідність за розмірами або відсутність елементів. У деяких випадках проводиться функціональне тестування, яке дозволяє перевірити працездатність плати в реальних умовах. Тільки після проходження всіх етапів контролю плата вважається готовою до експлуатації.

1.3 Дефекти друкованих плат

Процес виготовлення друкованих плат (ДП) включає безліч етапів, на кожному з яких можливе виникнення дефектів, рисунок 1.18. Дефекти можуть суттєво знизити працездатність і надійність плат, тому їхнє виявлення та усунення є ключовими завданнями в процесі виробництва. Найбільш поширеними дефектами ДП є помилки на етапах створення провідних доріжок і

контактних майданчиків, проблеми з пайкою, дефекти, пов'язані зі згинанням матеріалів, а також недоліки під час нанесення захисного покриття.

Одним із найпоширеніших типів дефектів є помилки у провідних доріжках і контактних майданчиках, що виникають на етапі фотолітографії або травлення. Якщо маска під час фотолітографії не точно відповідає необхідній схемі, це може призвести до розмиття доріжок, невідповідності їхньої ширини або навіть обривів. Такі дефекти часто виникають через проблеми з роздільною здатністю чи забрудненням маски. Вузькі або розмиті доріжки, а також обриви можуть порушити електричні з'єднання, що робить плату нефункціональною.

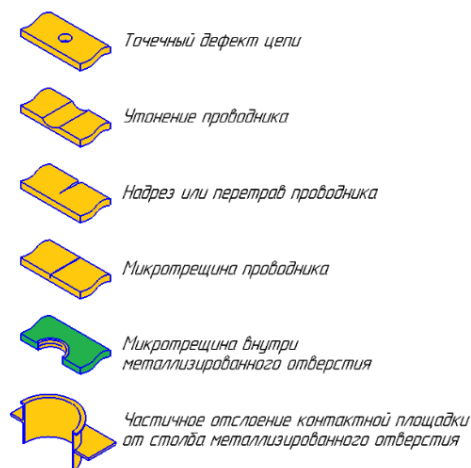


Рисунок 1.18 – Види скритих дефектів друкованої плати

Проблеми зі свердловими отворами та металізацією також є поширеними дефектами. Помилки під час свердління можуть призвести до неправильного розташування отворів або їхньої невідповідності за діаметром, що ускладнює монтаж компонентів. У разі поганої металізації стінок отворів існує ризик порушення зв'язку між шарами багат шарових плат, що критично для їхньої працездатності. Дефекти металізації можуть бути спричинені порушенням хімічного складу розчину або неправильними параметрами процесу осадження міді, рисунок 1.19.



Рисунок 1.19 – Відсутність металізації

Дефекти, пов'язані з пайкою, включають недостатню або надмірну пайку, утворення «холодних» паяних з'єднань і корозію у місцях пайки. Недостатня пайка може виникнути, якщо компонент погано прилягає до контактної майданчика або якщо кількість припою недостатня, рисунок 1.20. Надлишок припою часто призводить до утворення перемичок між сусідніми доріжками або висновками, що може спричинити коротке замикання. «Холодна» пайка виникає за недостатнього нагріву припою, що спричиняє погане з'єднання з контактним майданчиком і збільшує ризик виходу плати з ладу. Корозія в місцях пайки може з'являтися через порушення технологічного процесу або використання неякісного припою.



Рисунок 1.20 – Дефекти пайки друкованої плати

Механічні пошкодження та викривлення плати також є поширеними дефектами, які виникають під час виробництва або транспортування. Викривлення плати може статися через нерівномірне нагрівання під час пайки або надмірний вплив вологи на матеріал. Згинання і деформації порушують правильне положення компонентів, особливо в багатошарових платах, де мінімальні зміни форми можуть призвести до деформацій внутрішніх шарів і обривів доріжок. Механічні пошкодження можуть бути спричинені помилками під час різання або складання і включають тріщини, сколи або навіть обриви доріжок на краях плати, рисунок 1.21.



Рисунок 1.21 – Приклад механічних пошкоджень

Помилки нанесення захисного покриття також впливають на якість плати. Паяльна маска захищає доріжки від зовнішніх впливів, але дефекти під час нанесення можуть спричинити пропуски в покритті, що робить доріжки вразливими до корозії, рисунок 1.22. Крім того, помилки в товщині шару паяльної маски або її погане прилягання до поверхні плати можуть призвести до відшарування покриття, що оголює доріжки і робить плату вразливою до зовнішніх впливів. Важливо, щоб захисне покриття рівномірно покривало всі доріжки та контактні майданчики, адже будь-які пропуски можуть спричинити подальше пошкодження плати в процесі експлуатації.

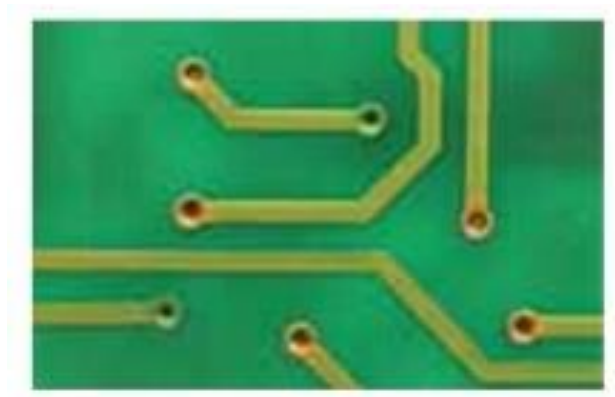


Рисунок 1.22 – Приклад неравномірного нанесення захисного покриття

1.4 Висновки до першого розділу

В ході написання першого розділу кваліфікаційної роботи, було розглянуто питання з приводу видів друкованих плат, технологій виготовлення провідникової топологічної структури та видів дефектів поверхні.

Особливу увагу було надано дефектам пов'язаним з контактними площадками елементів, таким як:

- відсутність площадки;
- нерівний контур площадки;
- зміщення отвору відносно провідникової доріжки;
- скал основи плати;
- відсутність металізації отвору.

Наявність подібних дефектів може бути критичною для працездатності виробу

Таким чином задачі пов'язані з візуальним контролем друкованих плат стають, все більш актуальними. Системи технічного зору (СТЗ), є одною з невіддільних складових для вирішення поставлених завдань.

2 МЕТОДИ ОБРОБКИ ЗОБРАЖЕНЬ

Методи обробки зображень є ключовими для автоматизованого аналізу та виявлення дефектів на друкованих платах. Основні етапи обробки включають фільтрацію, видалення шумів, бінаризацію, вирівнювання та інші техніки, що дозволяють покращити якість зображення для подальшої обробки. Фільтрація та зменшення шумів допомагають усунути перешкоди та зробити зображення чіткішим, а бінаризація перетворює його у двоколірний формат для спрощення аналізу. Також застосовуються методи вирівнювання, які коригують орієнтацію зображення, що особливо важливо для точного порівняння із зразками. Серед інших методів використовуються техніки виявлення контурів, накладання зображень та визначення розмірів об'єктів, що сприяє ідентифікації елементів та пошуку аномалій на друкованій платі.

2.1 Фільтрація зображень

Фільтрація зображень є одним з базових етапів у обробці зображень для автоматизованого аналізу друкованих плат. Метою фільтрації є покращення якості зображення шляхом видалення небажаних елементів, таких як шуми, а також виділення важливих деталей, які необхідні для подальшого аналізу. Це особливо важливо в умовах, коли зображення можуть мати перешкоди через проблеми зі зйомкою, оптичні спотворення або випадкові дефекти у процесі виробництва друкованих плат.

У фільтрації використовують різні типи фільтрів, серед яких найпоширенішими є медіанний, гаусівський та лінійний фільтри. Кожен з цих методів має свої особливості та застосовується в залежності від виду дефектів та особливостей оброблюваного зображення.

Медіанний фільтр є одним з найефективніших для видалення імпульсного шуму (так званого «соляного та перцевого» шуму), який може виникати на

зображеннях друкованих плат. Цей фільтр працює, замінюючи кожен піксель зображення на медіанне значення пікселів у його оточенні, що дозволяє зберегти контури об'єктів та зробити зображення більш чітким без значного розмиття, рисунок 2.1.

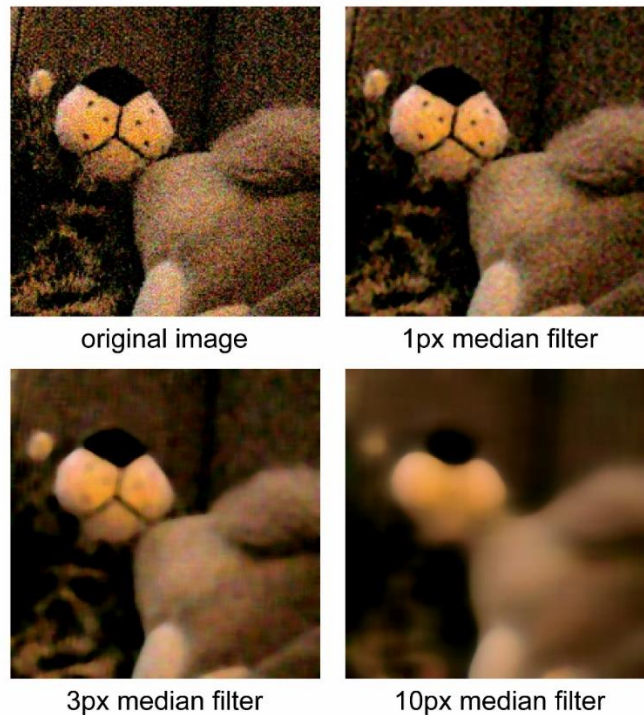


Рисунок 2.1 – Приклад медіанного фільтру

Гаусівський фільтр використовується для розмиття зображень та згладжування дрібних деталей. На відміну від медіанного фільтра, який добре видаляє «сольовий» шум, гаусівський фільтр застосовується, коли потрібно видалити незначні деталі, які не несуть корисної інформації для аналізу. Гаусівський фільтр зменшує рівень високочастотних шумів, що корисно при аналізі контурів елементів друкованих плат, але може розмивати дрібні деталі, тому його слід застосовувати з обережністю, рисунок 2.2.



Рисунок 2.2 – Розмиття за допомогою гаусівського фільтру

Лінійні фільтри, наприклад, фільтр Собеля або фільтр Лапласа, використовуються для виділення контурів об'єктів та різких переходів у зображенні. Це дозволяє більш чітко відобразити межі компонентів та провідників на друкованій платі. Фільтр Собеля, зокрема, дозволяє виявити градієнт яскравості зображення в обраному напрямку, що робить його корисним для визначення орієнтації об'єктів. Лапласіан, який є другим похідним оператором, підкреслює місця з різким зміщенням яскравості і є ефективним для аналізу контурів, рисунок 2.3.

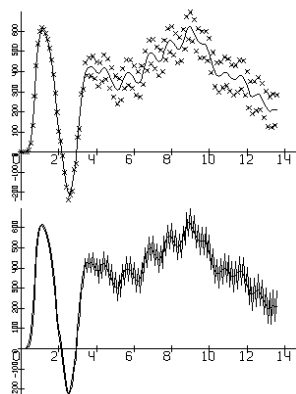


Рисунок 2.3 – Приклад сглажування лінійної функції за допомогою лінійного фільтру

Фільтрація є одним із найважливіших етапів обробки зображень, оскільки від якості відфільтрованого зображення залежить точність подальшого аналізу. Однак кожен метод фільтрації має свої обмеження. Наприклад, медіанний фільтр може зберегти контури об'єктів, але не ефективний для дрібних деталей. Гаусівський фільтр розмиває зображення, що знижує рівень деталізації, а лінійні фільтри можуть підсилювати шуми разом із контурами. Тому важливо вибирати тип фільтрації відповідно до завдань конкретного етапу аналізу.

2.2 Зменшення шумів

Зменшення шумів є важливим етапом обробки зображень у технології автоматизованого аналізу друкованих плат. Шум – це небажана інформація, яка може виникати під час зйомки, сканування або передавання зображення, а також через технічні особливості виробництва. Усунення шумів дозволяє підвищити якість зображення і зробити його чіткішим для подальшої обробки, що особливо важливо при виявленні дрібних дефектів на друкованих платах.

При обробці зображень друкованих плат найчастіше зустрічаються такі види шумів:

- імпульсний шум (сольовий та перцевий шум): проявляється у вигляді випадкових білих і чорних точок, які можуть приховувати важливі деталі;
- гаусівський шум: зазвичай має нормальний розподіл і виникає через коливання освітлення або інші випадкові коливання;
- спекл-шум: характерний для зображень, отриманих за допомогою лазерних або ультразвукових систем і проявляється у вигляді зернистості на зображенні.

Для зменшення цих шумів застосовують різні методи фільтрації, кожен з яких підходить для певного типу шуму. Одним із найбільш ефективних методів для зменшення імпульсного шуму є медіанна фільтрація, при якій кожен піксель замінюється на медіанне значення яскравості сусідніх пікселів. Це дозволяє зберегти контури та важливі деталі об'єктів на зображенні, одночасно видаляючи

випадкові пікселі-шуми. Гаусівський шум усувається за допомогою гаусівської фільтрації, що згладжує яскравість зображення і дозволяє зменшити дрібні коливання, однак при цьому може розмити контури. Тому гаусівську фільтрацію слід застосовувати обережно, щоб не втратити важливі дрібні деталі.

Іншим популярним підходом є використання білінійної та бікубічної інтерполяції, які дозволяють зменшити рівень шуму, одночасно зберігаючи чіткість основних контурів. Ці методи враховують яскравість сусідніх пікселів і коригують її для забезпечення більш згладженого переходу між пікселями. Бікубічна інтерполяція є особливо ефективною при зменшенні шумів, оскільки враховує більшу кількість сусідніх пікселів і зберігає деталі зображення.

Для складних зображень із неоднорідним розподілом шумів застосовують адаптивну фільтрацію, яка підлаштовує свої параметри залежно від рівня шуму в різних ділянках зображення. Це забезпечує гнучкий підхід до видалення шумів, дозволяючи зберегти чіткість в областях із дрібними деталями і видаляти шум у зашумлених ділянках. Адаптивна фільтрація є ефективною для друкованих плат, де важливо одночасно зберегти високий рівень деталізації і видалити випадкові перешкоди.

2.3 Вибір фільтрів для обробки зображень

Лінійні фільтри: лінійні фільтри використовуються для згладжування зображень та зменшення високочастотного шуму. Прикладами є середньозважений та гаусівський фільтри. Лінійні фільтри добре підходять для зменшення невеликого, розподіленого шуму, однак вони можуть розмивати краї об'єктів, що знижує чіткість зображення.

Нелінійні фільтри: до цього типу фільтрів належать медіанний фільтр та інші методи, які застосовують ранжування або медіанне значення для обробки пікселів. Нелінійні фільтри ефективні для видалення імпульсного шуму («соль і перець»), зберігаючи при цьому чіткі контури елементів.

Адаптивні фільтри: адаптивні фільтри налаштовують свої параметри в залежності від локальних характеристик зображення, що робить їх універсальними для роботи зі зображеннями з неоднорідним розподілом шуму. Адаптивна фільтрація дозволяє зберегти важливі деталі у одних ділянках зображення та ефективно видалити шум у інших.

Критерії вибору фільтрів. Вибір оптимального фільтра для обробки зображень друкованих плат залежить від таких факторів:

- тип шуму: для зменшення імпульсного шуму (випадкові білі і чорні точки) ідеально підходить медіанний фільтр. Гаусівський шум, який має нормальний розподіл, краще обробляється гаусівським фільтром або середньозваженим фільтром;

- збереження контурів: у випадках, коли важливо зберегти чіткість меж між елементами, слід використовувати нелінійні або адаптивні фільтри, такі як медіанний або адаптивний медіанний фільтр. Ці методи дозволяють зберігати структуру об'єктів, одночасно зменшуючи шум;

- інтенсивність згладжування: деякі фільтри дозволяють налаштувати рівень розмиття або згладжування, що особливо корисно для контролю якості друкованих плат. Наприклад, гаусівський фільтр дозволяє зменшити інтенсивність розмиття, щоб уникнути втрати дрібних деталей;

- локальні особливості зображення: у випадку неоднорідного шуму адаптивні фільтри, що змінюють свої параметри відповідно до особливостей різних областей зображення, можуть виявитися більш ефективними. Адаптивний медіанний фільтр змінює розмір області навколо пікселя в залежності від рівня шуму, що забезпечує гнучкість обробки.

Для початкового очищення зображення від загального рівня шуму зазвичай використовується гаусівський фільтр, який зменшує високочастотний шум і розгладжує фон. Це особливо корисно для зображень з низьким рівнем детальності, де важливо підвищити загальну чіткість перед детальнішим аналізом. Гаусівський фільтр забезпечує плавний перехід між пікселями, що допомагає зменшити вплив випадкових коливань яскравості.

При наявності імпульсного шуму, характерного для деяких зображень друкованих плат, медіанний фільтр є оптимальним вибором. Він дозволяє видалити випадкові точки шуму без значного розмиття контурів, зберігаючи чіткість елементів. Це особливо корисно при обробці зображень, де чіткість контурів і деталей є вирішальним фактором для виявлення дефектів, таких як обриви доріжок або нерівності контактних майданчиків.

Для високоякісних друкованих плат з дуже дрібними деталями може бути доцільним застосування адаптивних фільтрів. Наприклад, адаптивний медіанний фільтр ефективно видаляє шум, одночасно зберігаючи деталі в різних областях зображення. Цей підхід є ефективним для складних зображень із різним рівнем шуму в різних зонах, де важливо зберегти дрібні структури на поверхні плати.

При виборі фільтрів також важливо правильно налаштувати параметри, такі як розмір області фільтрації або рівень інтенсивності. Наприклад, занадто великий розмір області для медіанного фільтра може призвести до надмірного розмиття і втрати деталей, тоді як надто малий розмір не забезпечить належного видалення шуму. У випадку гаусівської фільтрації важливим є параметр сигми (σ), який контролює рівень згладжування: чим вищий σ , тим сильніше розмиття, що може бути небажаним для детальних зображень.

В обробці зображень друкованих плат нерідко використовують комбіновані методи, що поєднують різні типи фільтрів для досягнення оптимального результату. Наприклад, спочатку можна застосувати гаусівський фільтр для загального зменшення шуму, а потім використати медіанний фільтр для видалення залишків імпульсного шуму, що дозволяє досягти чистоти зображення без втрати чітких контурів, рисунок 2.4.

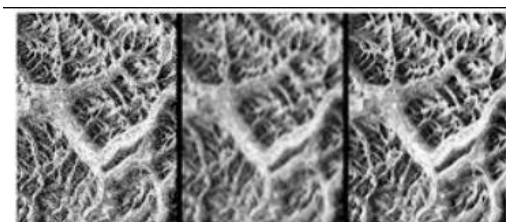


Рисунок 2.4 – Лінійна та нелінійна фільтрація

2.4 Бінаризація зображень

Бінаризація зображень є процесом автоматизованої обробки зображень для виявлення та аналізу дефектів на друкованих платах. Цей процес перетворює зображення у двоколірний формат (чорно-білий), що значно спрощує його аналіз та обробку. Бінаризація використовується для виділення ключових елементів (наприклад, провідників і контактних майданчиків) від фону, завдяки чому система технічного зору здатна краще ідентифікувати об'єкти на зображенні та знаходити дефекти.

Процес бінаризації полягає в поділі пікселів зображення на дві категорії: «чорні» (фон) та «білі» (об'єкти або ключові елементи). Вибір порогового значення є ключовим етапом у бінаризації, оскільки саме воно визначає, чи належить піксель до об'єкту або фону. Від точності вибору порогу залежить, наскільки добре будуть виділені потрібні елементи зображення, і, як наслідок, якість аналізу.

Сучасні методи бінаризації можна розділити на глобальні та локальні. У першому випадку величина порога залишається незмінною протягом усього процесу бінаризації. У другому зображення розбивається області, у кожному з яких обчислюється локальний поріг.

Глобальна бінаризація використовує одне єдине порогове значення, яке застосовується до всього зображення. Це найпростіший і швидкий метод, проте він має свої обмеження. Глобальна бінаризація добре працює для зображень із однорідним фоном і чіткими межами об'єктів, але її ефективність знижується для зображень із неоднорідним освітленням або складним фоном.

Одним із найпоширеніших методів глобальної бінаризації є метод Отсу. Він автоматично обирає порогове значення, мінімізуючи внутрішньокласову дисперсію яскравості пікселів. Метод Отсу є ефективним для однорідних зображень, де є чітка різниця між об'єктом та фоном, рисунок 2.5.



Рисунок 2.5 – Приклад методу Отсу

Локальна бінаризація розглядає кожен ділянку зображення окремо, підбираючи порогове значення відповідно до локальних особливостей. Це дозволяє компенсувати різні рівні освітлення на різних частинах зображення і забезпечує більш точне виділення об'єктів. Локальна бінаризація є особливо корисною для аналізу зображень друкованих плат, де освітлення може бути нерівномірним через особливості конструкції або відблиски.

Адаптивна бінаризація, як правило, працює шляхом обчислення середнього або медіанного значення для кожної ділянки зображення та встановлення порогу на основі цих значень. Вона дозволяє враховувати дрібні зміни в яскравості та робить процес виділення об'єктів більш точним.

Існує кілька різновидів адаптивної бінаризації, що використовуються залежно від характеристик зображення.

Адаптивна бінаризація з середнім порогом: для кожної невеликої області обчислюється середнє значення яскравості, яке потім використовується як поріг. Цей метод добре працює для зображень із плавними змінами освітлення. Адаптивна бінаризація з медіанним порогом: замість середнього значення використовується медіанне значення яскравості пікселів. Цей метод ефективний для зображень з високою різкістю змін освітлення, рисунок 2.6. Адаптивна бінаризація з граничним відхиленням: цей метод враховує відхилення яскравості пікселів у кожній області, що дозволяє компенсувати локальні варіації освітлення та забезпечує більш точне виділення об'єктів.

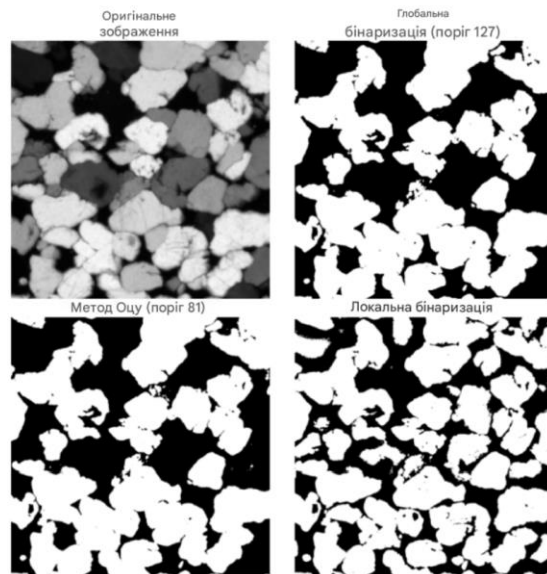


Рисунок 2.6 – Приклад роботи методів бінаризації

Бінаризація має низку переваг, серед яких спрощення обробки зображення та зменшення кількості даних для аналізу. Проте важливо враховувати і недоліки цього процесу, зокрема можливість втрати дрібних деталей або поява небажаних артефактів, якщо порогове значення вибрано невдало. Неправильно обране порогове значення може призвести до появи шумів або недостатньо чітких контурів об'єктів, що впливає на точність аналізу.

Глобальна бінаризація ефективна для зображень із однорідним освітленням, але її недоліком є низька ефективність на зображеннях із неоднорідним фоном або змінним освітленням. Наприклад, якщо одна частина друкованої плати освітлена сильніше, ніж інша, то глобальне порогове значення не дозволить коректно виділити всі об'єкти, що призведе до спотворення зображення та появи артефактів. Адаптивна бінаризація, на відміну від глобальної, здатна враховувати локальні особливості зображення. Це робить її незамінною для аналізу складних зображень, де рівень освітлення може змінюватися по всій площі. Вона дозволяє уникнути втрат важливих деталей і забезпечує більш чітке виділення об'єктів навіть при нерівномірному освітленні.

Бінаризація зображень є надзвичайно корисною при аналізі друкованих

плат, оскільки дозволяє виділити контури провідників, контактні майданчики та інші важливі елементи, які необхідно перевірити на наявність дефектів. Завдяки перетворенню зображення в двоколірний формат стає можливим більш точно і швидко розпізнавання дефектів, таких як обриви доріжок, порушення ширини провідників або неправильне розташування елементів.

На зображенні друкованої плати білі пікселі можуть представляти провідники та контактні майданчики, а чорні – фон або ізоляційні області. Після бінарizaції система технічного зору може ефективно аналізувати виділені області для виявлення аномалій.

Налаштування параметрів бінарizaції є критично важливим для досягнення оптимальних результатів. Основний параметр, який потрібно налаштувати, це порогове значення, яке може бути статичним для глобальної бінарizaції або динамічно змінюваним для адаптивної. Додатково, для адаптивної бінарizaції важливим є розмір області, в межах якої обчислюється локальне порогове значення. Занадто велика область може призвести до втрати дрібних деталей, тоді як занадто мала до появи зайвих шумів.

Порогове значення є центральним параметром при бінарizaції. Це значення інтенсивності, що визначає межу між фоном і об'єктами на зображенні. Кожен піксель, яскравість якого перевищує це значення, перетворюється на «білий» (належить до об'єкта), тоді як пікселі з яскравістю нижчою за поріг стають «чорними» (належать до фону).

Для простих випадків можна використовувати статичне порогове значення, яке залишається незмінним для всього зображення. Такий підхід є ефективним для зображень з однорідним освітленням і чітким контрастом між фоном і об'єктами. Однак глобальне порогове значення часто є недостатньо точним для складних зображень друкованих плат, де рівень освітлення може бути нерівномірним.

Для обробки зображень з неоднорідним освітленням часто використовується адаптивне порогове значення. У цьому випадку поріг обчислюється для кожної області окремо, що дозволяє враховувати локальні

особливості освітлення. Адаптивна бінаризація забезпечує більш точний поділ між об'єктами і фоном у випадках, коли рівень освітлення значно варіюється по всій площині зображення.

Розмір області для локальної бінаризації є важливим параметром, який визначає, у яких межах буде обчислюватися локальне порогове значення. Зазвичай для кожного пікселя вибирається невелика сусідня область, наприклад 3×3 , 5×5 або більша, і на основі значень яскравості пікселів у цій області встановлюється порогове значення.

Менші області, наприклад 3×3 або 5×5 пікселів, забезпечують точніше врахування локальних змін освітлення і можуть краще виділяти дрібні деталі. Однак вони також можуть виявляти дрібні шуми, що призводить до появи артефактів на зображенні. Тому для таких областей часто використовують додаткову фільтрацію, щоб уникнути помилкових спотворень.

Великі області (наприклад, 15×15 або 21×21 пікселів) підходять для згладжування великих змін освітлення, але вони можуть втратити дрібні деталі. Це може бути прийнятним для простих зображень, але в випадках, коли необхідно зберегти дрібні контури і деталі (наприклад, тонкі доріжки на друкованих платах), великі області можуть бути неефективними.

У деяких адаптивних алгоритмах бінаризації додатково використовується параметр порогового відхилення, який встановлює значення, що додається або віднімається від середнього (або медіанного) значення в локальній області. Це дозволяє налаштувати бінаризацію так, щоб краще підходити для специфічних умов, наприклад, коли потрібно зберегти деталі на яскравих об'єктах або приглушити їх на темних. Зазвичай порогове відхилення вводиться як додаткова константа C , яка дозволяє коригувати порогове значення в залежності від особливостей зображення.

Формула для адаптивної бінаризації з пороговим відхиленням може виглядати так, формула (2.1):

$$T(x, y) = \frac{1}{N} \sum_{(i,j) \in S(x,y)} I(i, j) - C, \quad (2.1)$$

де $T(x, y)$ – локальне порогове значення для пікселя в координаті (x, y) ;

$S(x, y)$ – локальна область (вікно) навколо пікселя (x, y) ;

N – кількість пікселів у локальній області $S(x, y)$;

$I(i, j)$ – значення яскравості пікселя (i, j) у локальній області;

C – параметр порогового відхилення, який коригує порогове значення.

2.5 Методи вирівнювання зображень

Загальний етап обробки зображень у системах технічного зору, також особливо важливий при використанні для контролю друку печатних плат, полягає в вирівнюванні зображень. Цей процес зводить зображення до одного орієнтованого виду, що полегшує подальший аналіз. Вирівнювання усуває проблему випадкових зсувів, поворотів або деформацій, які можуть виникати під час фотографування або пакування предметів.

Процес вирівнювання полягає в тому, щоб знайти перетворення, яке максимально узгоджує поточне зображення з шаблоном або еталоном. Основні задачі вирівнювання зображень включають:

- зміщення – корекція лінійних зсувів зображення вздовж осей X та Y ;
- поворот – корекція орієнтації зображення, щоб забезпечити однакове положення щодо базового об'єкта;
- масштабування – зміна розміру зображення для забезпечення співпадіння масштабів зразка та об'єкта.

Метод кореляції. Кореляційний метод є одним з найбільш розповсюджених для вирівнювання, він передбачає обчислення коефіцієнта кореляції між зображенням і еталоном. Коефіцієнт кореляції визначається наступною формулою (2.2):

$$r = \frac{\sum_{x,y}(I(x,y)-\bar{I})(T(x,y)-\bar{T})}{\sqrt{\sum_{x,y}(I(x,y)-\bar{I})^2 \sum_{x,y}(T(x,y)-\bar{T})^2}}, \quad (2.2)$$

де $I(x,y)$ – інтенсивність пікселя в зображенні;
 $T(x,y)$ – інтенсивність пікселя в еталоні;
 \bar{I} та \bar{T} – середні значення інтенсивності пікселів.

Метод ключових точок. Використовується для вирівнювання зображень, що мають чіткі особливості, такі як кути або грані. Алгоритми SIFT (Scale-Invariant Feature Transform) або SURF (Speeded Up Robust Features) визначають ключові точки, які потім зіставляються між зображеннями для вирівнювання, рисунок 2.7. [9]

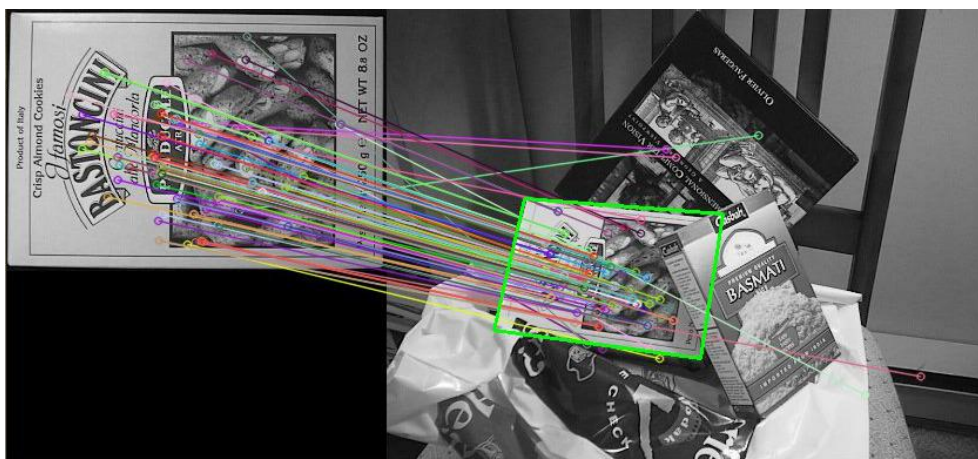


Рисунок 2.7 – Приклад методу ключових точок

Метод на основі гістограм. Цей метод корисний для вирівнювання зображень з нерівномірним освітленням, зокрема для налаштування яскравості і контрасту, рисунок 2.8. Його можна описати наступною формулою для вирівнювання гістограм, формула (2.3):

$$H'(I) = \frac{H(I)-H_{min}}{H_{max}-H_{min}} \cdot (L - 1), \quad (2.3)$$

де $H(I)$ – гістограма інтенсивностей;
 X_{min} і H_{max} – мінімальні і максимальні значення;
 L – кількість рівнів яскравості (зазвичай 256).

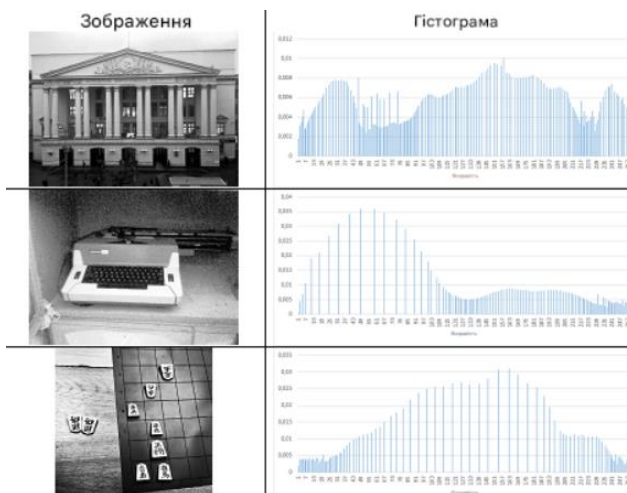


Рисунок 2.8 – Метод на основі гістограм

Афінне перетворення. Виконує зміщення, поворот, масштабування та зсув одночасно, рисунок 2.9. Формула афінного перетворення:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix},$$

де a, b, c, d – параметри для повороту і масштабування;
 t_x, t_y – зміщення.

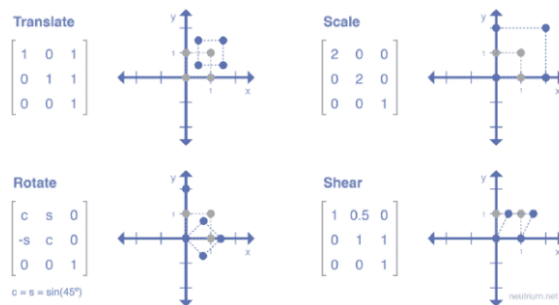


Рисунок 2.9 – Афінне перетворення

2.6 Віднімання та накладання зображень

Віднімання та накладання зображень є важним аспектом в обробці зображень, особливо у випадках, коли потрібно виявити відмінності між двома зображеннями або об'єднати їх для подальшого аналізу. Ці операції широко застосовуються у системах технічного зору для контролю якості друкованих плат, де навіть найменші відхилення можуть сигналізувати про дефекти або відхилення від технологічного процесу.

Віднімання зображень полягає у порівнянні двох зображень шляхом вирахування значень яскравості пікселів одного зображення з відповідними значеннями іншого зображення. У результаті такої операції виходить нове зображення, що містить лише ті області, де є відмінності. Це дозволяє легко виявити дефекти, такі як тріщини, порушення доріжок або інші непередбачені зміни, які могли виникнути у процесі виробництва друкованих плат. Для віднімання зображень використовується формула, де кожен піксель результату є абсолютним значенням різниці інтенсивностей відповідних пікселів двох зображень [11].

Таким чином, якщо два пікселі мають однакову інтенсивність, то результат їх віднімання дорівнює нулю, що означає, що на цьому місці немає змін. У випадку ж, якщо інтенсивності різняться, у результаті ми отримаємо область з високою контрастністю, яка вказує на відмінності. Цей підхід є дуже зручним для автоматизації процесу контролю якості, оскільки дозволяє системі технічного зору швидко виявляти відхилення від еталона без втручання оператора. Накладання зображень, у свою чергу, є операцією, що дозволяє поєднати два або більше зображень для створення нового зображення, яке включає інформацію з усіх об'єднаних зображень. Це може бути корисним у ситуаціях, коли потрібно порівняти кілька шарів друкованої плати або оцінити розташування компонентів відносно один одного.

Накладання зображень часто реалізується шляхом взяття середньозваженого або середньоарифметичного значення яскравості

відповідних пікселів на двох зображеннях, що дозволяє створити результат із збереженням основних рис кожного зображення. Це забезпечує можливість бачити загальну структуру плати з усіма елементами, що допомагає у виявленні невідповідностей у багатошарових платах. Однак накладання потребує обережного підходу до налаштування вагових коефіцієнтів для кожного зображення, оскільки неправильні значення можуть призвести до втрати контрастності або недостатньо чіткого відображення деталей.

Перевагою накладання є можливість одночасного перегляду кількох шарів або різних частин плати, що дає більш повну картину розташування елементів. Існують певні труднощі, пов'язані з відніманням і накладанням зображень. Однією з них є необхідність точного вирівнювання зображень, адже найменше зміщення може призвести до помилкових результатів і виявлення неіснуючих дефектів. Щоб уникнути цього, перед операціями віднімання або накладання зазвичай застосовуються алгоритми вирівнювання, які коригують положення зображень і забезпечують точне співпадіння їх елементів. Крім того, при накладанні може виникати проблема надлишкової яскравості або розмиття, тому після цієї операції інколи додатково використовують корекцію контрастності або фільтрацію, щоб покращити якість результату.

Операції віднімання та накладання зображень є ефективними інструментами для аналізу та контролю якості друкованих плат. Вони дозволяють автоматизувати процеси виявлення дефектів та перевірки відповідності, що значно підвищує точність і швидкість перевірки без необхідності ручної інспекції.

2.7 Розрахунок передавальних функцій та складання структурної схеми системи технічного зору

Структурна схема динамічної системи відображає геометрію системи і характеризує як склад її елементів, так і зв'язки між ними. На структурній схемі системи вказують шляхи та напрямки передачі інформації (сигналів).

Стан системи, а також входять до неї елементів описується певним числом незалежних координат, частина яких доступна вимірюванню: наприклад, для електричних систем – це струм, напруга, потужність; для механічних систем – кут повороту, переміщення, швидкість і т.д

Для опису взаємодії системи із зовнішнім середовищем, а також підсистеми з іншими підсистемами на вході і виході вибирають відповідні векторні координати (вхідні і вихідні). Будемо поки вважати, що координати, або змінні динамічної системи – скалярні функції часу, нехай вхідна величина $g(t)$, а вихідна $z(t)$.

Так, плата Raspberry PI це в нас головний пристрій, який контролює плату Arduino Mega та проектор. У той же час, плата Arduino Mega управляє моторчиком шторки, який закриває наш проектор. Arduino Mega постійно обмінюється даними з платою Raspberry PI у ході роботи.

Лампа проектору (Л) містить: лампа як об'єкт регулювання, плату Raspberry PI у ланцюзі негативного зворотного зв'язку, елемент порівняння, підсилювач (ПС), плату Arduino Mega (АРД) і камера (К), рисунок 2.10. Значення параметрів системи автоматичного управління наведено в табл. 2.1.

Таблиця 2.1 – Значення параметрів САУ

Найменування	Позначення	Значення
Коефіцієнт підсилення ПС	$K_{ПС}$	1500
Коефіцієнт підсилення Л	K_L	1,2
Стала часу	$T_1 [с]$	1,25
Стала часу	$T_2 [с]$	0,4
Коефіцієнт підсилення АРД	$K_{АРД}$	0,0002
Стала часу АРД	$T_{АРД}$	0,02

До табл. 2.2 заносимо вимоги, яким повинна задовольняти вихідна САУ.

Таблиця 2.2 – Вимоги, щодо якості процесу управління

Найменування	Позначення	Значення
Час перерегулювання	σ	30
Час регулювання	$t_{рег}$	3

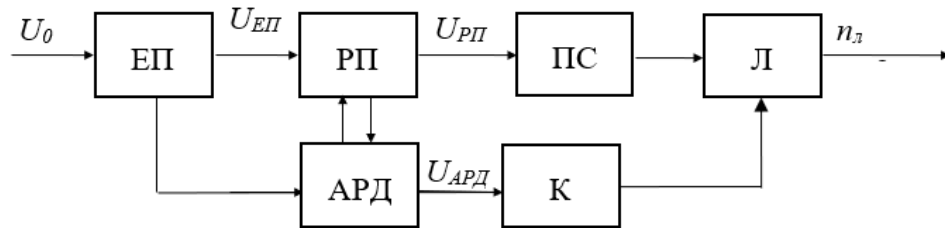


Рисунок 2.10 – Функціональна схема САУ

Елемента порівняння, можливо визичити, формула (2.4):

$$\begin{aligned}
 U_{EP} &= U_0 - U_{ARD}, \\
 U_{EP} &= U_0 - U_{ВП},
 \end{aligned}
 \tag{2.4}$$

де U_{EP} – вихідна напруга ЕП;
 U_0 U_{ARD} ідне значення напруги;
 $U_{ВП}$ – напруга на виході АРД.

Елемент керування можливо визичити, формула (2.5):

$$\begin{aligned}
 U_{RP} &= K_{RP} U_{EP}, \\
 U_{П} &= K_{П} U_{EP},
 \end{aligned}
 \tag{2.5}$$

де U_{RP} напруга на виході підсилювача напруги;
 $K_{П}$ – коефіцієнт посилення K_{RP} підсилювача напруги.

Передавальна функція підсилювача, буде описуватися наступним рівнянням, формула (2.6):

$$\begin{aligned} W_{PI}(s) &= K_{PI} , \\ W_{II}(s) &= K_{II} . \end{aligned} \quad (2.6)$$

Формула підсилювача, можливо виразити, формула (2.7):

$$\frac{dG_T}{dt} = K_{PIC} U_{PI} , \quad (2.7)$$

де G_T – витрата енергії;
 K_{DO} – коефіцієнт K_{PIC} посилення ПС.

Передавальна функція підсилювача, формула (2.8):

$$\begin{aligned} W_{DO}(s) &= \frac{K_{DO} \cdot U_{PI}}{s \cdot U_{PI}} , \\ W_{DO}(s) &= \frac{K_{DO}}{s} , \\ W_{DO}(s) &= \frac{1500}{s} . \end{aligned} \quad (2.8)$$

Керування освітленням зони системи технічного зору можливо виразити, формула (2.9):

$$T_2^2 \frac{d^2 n_L}{dt^2} + T_1 \frac{dn_L}{dt} + n_L = n_L = K_L \left(\tau_1 \frac{dG_T}{dt} + G_T \right) , \quad (2.9)$$

де n_L – частота лампи;
 K_L – коефіцієнт посилення лампи;
 T_1, T_2 – сталі часу лампи для освітлення.

Таким чином, передавальна функція лампи, формула (2.10):

$$W_L(s) = \frac{G_T \cdot K_L \cdot (\tau_1 \cdot s + 1)}{(T_2 \cdot s^2 + T_1 \cdot s + 1) \cdot G_T}$$

$$W_L(s) = \frac{K_L \cdot (\tau_1 \cdot s + 1)}{(T_2 \cdot s^2 + T_1 \cdot s + 1)},$$

$$W_L(s) = \frac{1,2 \cdot (1,1 \cdot s + 1)}{(0,4 \cdot s^2 + 1,25 \cdot s + 1)}. \quad (2.8)$$

Формула для керування плати Arduino, можливо описати за допомогою формули (2.11):

$$T_{\text{АРД}} \frac{dU_{\text{АРД}}}{dt} + U_{\text{АРД}} = K_{\text{АРД}} n_{\text{л}}, \quad (2.11)$$

де $T_{\text{АРД}}$ – стала часу АРД;

$K_{\text{АРД}}$ – коефіцієнт посилення АРД.

Передавальна функція плати Arduino, формула (2.12):

$$W_{\text{АРД}}(s) = \frac{U_{\text{АРД}}}{n_{\text{л}}},$$

$$W_{\text{АРД}}(s) = \frac{K_{\text{АРД}} \cdot n_{\text{л}}}{(T_{\text{АРД}} \cdot s + 1) \cdot n_{\text{л}}},$$

$$W_{\text{АРД}}(s) = \frac{K_{\text{АРД}}}{(T_{\text{АРД}} \cdot s + 1)},$$

$$W_{\text{АРД}}(s) = \frac{0,0002}{(0,02 \cdot s + 1)}. \quad (2.8)$$

Структурна схема нескоригованої системи зображена на рисунку 2.11.

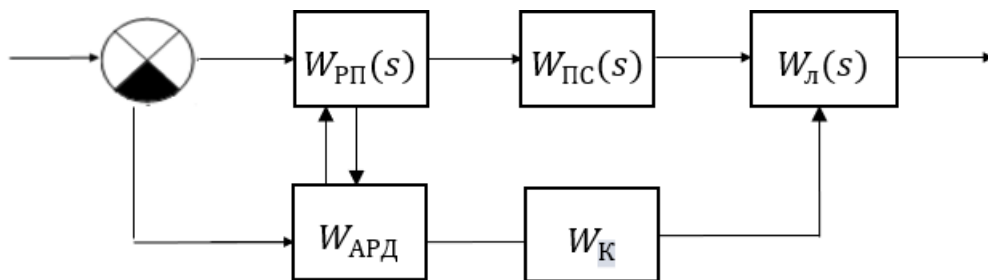


Рисунок 2.11 – Структурна схема нескоригованої системи

2.8 Висновки до другого розділу

У другому розділі розглянуто такі питання:

– етапи підготовки реального зображення для обробки СТЗ. Дане завдання складається з наступних етапів: фільтрація; згладжування зображення; виділення реперних точок; нормалізація (масштабування та поворот) та порівняння отриманого зображення з еталонним;

– були розглянуті методи лінійної фільтрації та згладжування зображення;

– розглянуто просту та адаптивну бінаризацію з використанням методу порогового значення Отцу;

– розглянуті методи вирівнювання зображення;

– розглянуті методи порівняння зображень на основі суми абсолютних різниць SAD.

На основі проведеної роботи можливо буде визначити які методи обробки зображення можливо буде використовувати для розробки програмного забезпечення для візуального контролю друкованих плат.

3 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ АВТОМАТИЗОВАНОГО АНАЛІЗУ ПЛАТ

3.1 Бібліотеки

3.1.1 OpenCV

OpenCV (Open Source Computer Vision Library) – це відкрита бібліотека комп'ютерного зору та обробки зображень, яка забезпечує швидкий і зручний доступ до сучасних алгоритмів комп'ютерного зору та машинного навчання. Вперше вона була створена компанією Intel у 2000 році та наразі активно використовується в таких галузях, як робототехніка, медицина, автономні транспортні засоби, обробка зображень, відеоаналітика та індустрія розваг [5].

OpenCV має широкий спектр функціональних можливостей. Зокрема, вона дозволяє зчитувати та записувати зображення у різних форматах, здійснювати перетворення кольорових просторів, застосовувати фільтрацію та згладжування, виконувати геометричні трансформації, такі як масштабування, повороти та обрізка. Для аналізу зображень бібліотека забезпечує методи виділення контурів, сегментації, виявлення ключових точок і побудови дескрипторів за допомогою алгоритмів, таких як SIFT, SURF або ORB [10].

Ця бібліотека також дозволяє працювати з відеопотоком у реальному часі, відстежувати рух об'єктів, стабілізувати відео та виконувати аналіз відеокadrів. Завдяки інтеграції з мовами програмування Python, C++, Java, а також підтримці глибокого навчання через TensorFlow або PyTorch, OpenCV стала універсальним інструментом у багатьох сферах. Наприклад, у робототехніці бібліотека використовується для навігації та управління роботами через аналіз зображень із камер, у медицині – для аналізу МРТ та рентгенівських знімків, а в безпеці – для розпізнавання облич і відстеження руху. Її широке застосування обумовлено високою швидкістю, точністю, простотою використання та активною підтримкою з боку спільноти розробників.

У контексті даної програми OpenCV була обрана для вирівнювання зображень, виявлення ключових точок і аналізу відмінностей завдяки її здатності забезпечувати високу продуктивність та точність обробки.

3.1.2 NumPy

NumPy – це потужна бібліотека для роботи з багатовимірними масивами даних і виконання високопродуктивних математичних операцій у мові програмування Python. Вона була створена як інструмент для наукових і технічних обчислень та стала стандартом для роботи з числовими даними у Python. Основною функціональною одиницею NumPy є багатовимірний масив (ndarray), який дозволяє зберігати та обробляти великі обсяги даних у структурованому вигляді [7].

NumPy надає широкий спектр інструментів для виконання операцій, таких як математичні обчислення (додавання, множення, трансцендентні функції), статистичний аналіз, обчислення лінійної алгебри (розклади матриць, знаходження власних значень) та робота з випадковими числами. Завдяки оптимізації на базі мов програмування C і Fortran, бібліотека забезпечує високу швидкість виконання операцій, що робить її незамінною для великих обчислювальних задач. NumPy активно використовується у таких сферах, як машинне навчання, комп'ютерне моделювання, обробка зображень, статистика та аналіз даних.

У нашій програмі NumPy забезпечує зручну та ефективну роботу з числовими масивами, зокрема при виконанні операцій зображень, таких як перетворення форматів, робота з масками та матрицями для аналізу. Її використання дозволило спростити та прискорити числові обчислення, які є основою алгоритмів програми.

3.1.3 Tkinter

Tkinter – це стандартна бібліотека для створення графічного інтерфейсу користувача (GUI) у мові програмування Python. Вона є оболонкою над

інструментарієм Tk, що дозволяє створювати зручні й функціональні інтерфейси для десктопних додатків. Tkinter підтримує широкий набір елементів інтерфейсу, таких як кнопки, текстові поля, мітки, меню, полоси прокрутки та інші інтерактивні компоненти, які можна легко налаштовувати для відповідності потребам конкретного додатку [6].

Перевагою Tkinter є його простота використання, що дозволяє швидко розробляти інтерфейси навіть без глибоких знань графічного програмування. Крім того, Tkinter має кросплатформенність, тобто створені додатки працюють однаково як у Windows, так і в Linux чи macOS.

Tkinter забезпечує можливість створювати багатопотокові інтерфейси, обробляти події від користувача (наприклад, натискання кнопок) і динамічно змінювати вигляд елементів залежно від змін у програмі [12].

Я використав Tkinter для реалізації графічного інтерфейсу користувача, що забезпечує зручну взаємодію з програмою. Наприклад, бібліотека надає інструменти для завантаження файлів через діалогове вікно, відображення зображень, а також створення кнопок для запуску основних функцій програми, таких як обробка чи аналіз зображень. Використання Tkinter дозволило забезпечити інтуїтивно зрозумілий інтерфейс, який підвищує зручність і ефективність роботи з програмою.

3.1.4 Pillow

Pillow (або PIL) – це потужна бібліотека для роботи із зображеннями в мові програмування Python, яка забезпечує широкі можливості для обробки та маніпуляції графічними файлами. Pillow є вдосконаленою і підтримуваною версією оригінальної бібліотеки PIL, яка була створена для спрощення роботи з графічними форматами. Основні можливості Pillow включають:

- зчитування і збереження зображень: бібліотека підтримує численні графічні формати, включаючи JPEG, PNG, BMP, GIF, TIFF та інші, що робить її універсальною для роботи з будь-якими зображеннями;

- масштабування та зміна розмірів. Pillow дозволяє легко змінювати розміри зображень, зберігаючи їх пропорції або обрізаючи до заданих розмірів;
- обробка зображень. бібліотека надає інструменти для виконання базових операцій, таких як повороти, обертання, дзеркальне відображення, а також для зміни кольорових форматів (наприклад, перетворення RGB у відтінки сірого);
- фільтри та ефекти. Pillow підтримує різні фільтри, які дозволяють розмивати, згладжувати або підвищувати різкість зображень;
- робота з текстом. бібліотека дозволяє накладати текст на зображення, що може бути корисним для створення водяних знаків чи підписів.

Pillow широко використовується у сфері веб-розробки, машинного навчання, обробки зображень, автоматизації робочих процесів і створення графічних інтерфейсів [8].

В своїй програмі я застосував Pillow для завантаження зображень із файлової системи, їхнього масштабування та відображення в графічному інтерфейсі користувача, розробленому за допомогою Tkinter. Бібліотека також використовується для підготовки зображень перед подальшою обробкою, що забезпечує високу якість візуалізації та спрощує роботу з графічними файлами. Завдяки простоті використання і широким функціональним можливостям, Pillow стала ключовим інструментом у реалізації задач обробки зображень у цьому проєкті.

3.2 Ключові методи

3.2.1 SIFT

Scale-Invariant Feature Transform (SIFT) – це алгоритм, який використовується для знаходження ключових точок на зображенні, стійких до змін масштабу, повороту та освітлення. Цей метод виділяє унікальні особливості зображення (наприклад, кути або перетини ліній), які можна використовувати для порівняння і вирівнювання зображень.

Робота з SIFT у програмі реалізована наступним чином: Спершу за допомогою функції `detectAndCompute` обчислюються ключові точки та їх дескриптори для двох зображень. Потім використовується алгоритм FLANN для пошуку відповідностей між дескрипторами ключових точок. Для фільтрації збігів застосовується критерій Лоу (Lowe's ratio test), який відсіює слабкі або помилкові збіги [13]. Після цього відфільтровані відповідності використовуються для обчислення матриці гомографії за допомогою алгоритму RANSAC. Ця матриця дозволяє вирівняти одне зображення щодо іншого, рисунок 3.1.

```
def align_images(img1, img2):
    sift = cv2.SIFT_create()
    kp1, des1 = sift.detectAndCompute(img1, None)
    kp2, des2 = sift.detectAndCompute(img2, None)

    index_params = dict(algorithm=1, trees=5)
    search_params = dict(checks=50)
    flann = cv2.FlannBasedMatcher(index_params, search_params)

    matches = flann.knnMatch(des1, des2, k=2)

    good_matches = []
    for m, n in matches:
        if m.distance < 0.7 * n.distance:
            good_matches.append(m)

    matched_image = cv2.drawMatches(img1, kp1, img2, kp2, good_matches, None, flags=cv2.DrawMatchesFlags_NOT_DRAW_SINGLE_POINTS)
    cv2.imshow("Співпадання ключових точок", matched_image)

    if len(good_matches) < 4:
        raise ValueError("Недостатньо збігів для вичислення гомографії.")

    src_pts = np.float32([kp1[m.queryIdx].pt for m in good_matches]).reshape(-1, 1, 2)
    dst_pts = np.float32([kp2[m.trainIdx].pt for m in good_matches]).reshape(-1, 1, 2)

    M, mask = cv2.findHomography(dst_pts, src_pts, cv2.RANSAC, 5.0)

    if M is None:
        raise ValueError("Не вдалося порахувати матрицю гомографії.")
    return M
```

Рисунок 3.1 – Фрагмент коду з реалізації SIFT у програмі

3.2.2 FLANN

У програмі для порівняння дескрипторів, отриманих за допомогою алгоритму SIFT, використовується FLANN (Fast Library for Approximate Nearest Neighbors). Це алгоритм, який забезпечує швидкий пошук найближчих сусідів у багатовимірних просторах, що робить його ефективним для роботи з великими обсягами даних. FLANN використовує оптимізовані структури даних, такі як kd-дерева або ієрархічні кластеризаційні дерева, для прискорення пошуку.

Основна задача FLANN – знайти відповідності між дескрипторами двох зображень, щоб визначити спільні ключові точки. Після пошуку відповідностей застосовується критерій Лоу (Lowe's ratio test) для фільтрації ненадійних збігів.

У програмі алгоритм FLANN реалізовано наступним чином, рисунок 3.2.

```
index_params = dict(algorithm=1, trees=5)
search_params = dict(checks=50)
flann = cv2.FlannBasedMatcher(index_params, search_params)

matches = flann.knnMatch(des1, des2, k=2)

good_matches = []
for m, n in matches:
    if m.distance < 0.7 * n.distance:
        good_matches.append(m)
```

Рисунок 3.2 – Використання алгоритму FLANN у кодї

FLANN використовується для знаходження відповідностей між дескрипторами ключових точок двох зображень. Ці відповідності є основою для обчислення матриці гомографії, що дозволяє вирівнювати зображення. Завдяки високій швидкості та точності FLANN ідеально підходить для задач аналізу та порівняння зображень, які реалізовані в програмі.

3.2.3 Метод обчислення гомографії

У програмі для вирівнювання зображень реалізовано обчислення гомографії, для точного накладання зображень одне на одне. Гомографія – це матриця перетворення, яка дозволяє трансформувати одне зображення так, щоб його ключові точки збігалися з ключовими точками іншого зображення. Такий підхід використовується для корекції різних перетворень: зміщення, повороту, масштабу, а також перспективних спотворень.

Обчислення гомографії виконується в кілька етапів:

- зіставлення ключових точок: на основі ключових точок, знайдених за допомогою алгоритму SIFT і зіставлених через FLANN, визначаються координати пар точок, які відповідають одна одній;
- алгоритм RANSAC: використовується для обчислення стійкої матриці гомографії, яка відкидає викиди (помилкові відповідності точок);
- перетворення зображення: отримана матриця застосовується для трансформації зображення з використанням функції `cv2.warpPerspective`.

3.2.4 Метод бінаризації

Бінаризація зображень є ключовим етапом у програмі, який дозволяє перетворити зображення в двоколірний формат (чорно-білий) для подальшого аналізу. Цей процес полегшує виділення контурів об'єктів та зменшує обсяг даних для обробки, що значно підвищує швидкість і точність алгоритмів виявлення аномалій. У програмі реалізовано два підходи до бінаризації: фіксоване порогове перетворення та адаптивна бінаризація [14].

Фіксоване порогове перетворення (`cv2.threshold`) використовується для зображень з рівномірним освітленням, де поріг яскравості залишається незмінним для всього зображення. У випадках, коли освітлення нерівномірне, застосовується адаптивна бінаризація, яка обчислює локальний поріг для кожної області зображення, дозволяючи точніше виділити елементи, рисунок 3.3.

```
def preprocess_image(image_path):
    image = cv2.imread(image_path, cv2.IMREAD_GRAYSCALE)
    if image is None:
        raise FileNotFoundError(f"Не вдалося завантажити зображення: {image_path}")
    _, binary = cv2.threshold(image, 127, 255, cv2.THRESH_BINARY)
    return binary
```

Рисунок 3.3 – Метод `preprocess_image()`, який використовує бінаризацію

Переваги бінаризації:

- спрощення аналізу. Всі пікселі зображення зводяться до двох значень, що дозволяє легко ідентифікувати об'єкти на фоні;

- прискорення обчислень. Завдяки зменшенню обсягу даних, алгоритми обробки працюють швидше;
- гнучкість. Адаптивна бінаризація дозволяє ефективно працювати з нерівномірно освітленими зображеннями.

Бінаризація є підготовчим етапом для аналізу зображень. В моєму випадку вона забезпечує чітке виділення контурів об'єктів, таких як провідники чи контактні майданчики на друкованих платах. Завдяки цьому програмі легше визначати аномалії та дефекти під час подальших етапів обробки, таких як вирівнювання чи виявлення змін.

3.2.5 Морфологічні операції

Морфологічні операції є важливим етапом обробки бінарних зображень у програмі. Вони дозволяють усунути шуми, покращити якість контурів об'єктів і зробити зображення більш придатним для подальшого аналізу. У програмі використовуються такі основні морфологічні операції: закриття, розширення та індивідуальне налаштування структурного елемента, рисунок 3.4.

```
kernel = cv2.getStructuringElement(cv2.MORPH_RECT, (3, 3))
difference_binary = cv2.morphologyEx(difference_binary, cv2.MORPH_CLOSE, kernel)
difference_binary = cv2.dilate(difference_binary, kernel, iterations=2)
```

Рисунок 3.4 – Реалізація морфологічних операцій

Морфологічні операції потрібні для підготовки бінарних зображень для подальшого аналізу. Вони допомагають підготувати зображення для пошуку контурів і виявлення відмінностей, забезпечуючи надійне визначення дефектів друкованих плат. Завдяки цим операціям зменшується кількість помилкових виявлень та підвищується точність аналізу.

3.2.6 Обчислення різниці зображень

Метод обчислення різниці зображень потрібен для виявлення відмінностей між двома вирівняними зображеннями. У програмі використовується функція

cv2.absdiff, яка дозволяє порівняти інтенсивності пікселів на двох зображеннях і отримати зображення різниць. Цей підхід дозволяє ідентифікувати області, що зазнали змін, такі як дефекти чи зміщення елементів, рисунок 3.5.

Етапи реалізації:

- обчислення абсолютної різниці. Абсолютна різниця між пікселями двох зображень обчислюється за допомогою cv2.absdiff. Ця операція виділяє ті області, де пікселі мають значну різницю в яскравості;
- бінаризація результату. Застосовується порогове перетворення для виділення найбільш значущих відмінностей;
- пошук контурів. Використовується функція cv2.findContours для визначення областей, де є відмінності, і аналізу їхніх характеристик.

```

difference = cv2.absdiff(img1_original, img2_aligned)
_, difference_binary = cv2.threshold(difference, 140, 255, cv2.THRESH_BINARY)

kernel = cv2.getStructuringElement(cv2.MORPH_RECT, (3, 3))
difference_binary = cv2.morphologyEx(difference_binary, cv2.MORPH_CLOSE, kernel)
difference_binary = cv2.dilate(difference_binary, kernel, iterations=2)

```

Рисунок 3.5 – Реалізація обчислення різниці зображень

Цей метод потрібен для виявлення дефектів друкованих плат. Завдяки аналізу різниць між вирівняними зображеннями програма може точно ідентифікувати відхилення, такі як розриви провідників, деформації чи інші дефекти.

3.2.7 Аналіз контурів

Аналіз контурів обробки бінарних зображень дозволяє виділити області змін або дефектів для подальшої оцінки. У програмі контури визначаються після морфологічної обробки зображення, а їх параметри аналізуються для відсіву шумів та незначних артефактів. Основними критеріями аналізу є площа контурів та співвідношення сторін обмежувальних прямокутників, рисунок 3.6.

```

contours, _ = cv2.findContours(difference_binary, cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_SIMPLE)
img_with_differences = cv2.cvtColor(img1_original, cv2.COLOR_GRAY2BGR)

for contour in contours:
    area = cv2.contourArea(contour)
    if 100 < area < 10000:
        x, y, w, h = cv2.boundingRect(contour)
        aspect_ratio = w / h
        if 0.2 < aspect_ratio < 5:
            cv2.rectangle(img_with_differences, (x, y), (x + w, y + h), (0, 0, 255), 2)

```

Рисунок 3.6 – Аналіз контурів в програмі

3.3 Огляд програми

3.3.1 Load_image(label)

Ця функція відповідає за завантаження зображення користувачем через графічний інтерфейс. Користувач взаємодіє з діалоговим вікном, у якому обирає файл із зображенням. Після завантаження зображення обробляється, зменшується для зручного відображення та прикріплюється до елемента інтерфейсу, рисунок 3.7. Основні дії функції:

- відкриває вікно вибору файлу за допомогою `filedialog.askopenfilename`;
- завантажує зображення через бібліотеку Pillow (`Image.open`);
- масштабує зображення до розміру 400x400 пікселів методом `thumbnail`, щоб зберегти пропорції;
- створює об'єкт `ImageTk.PhotoImage` для відображення зображення у Tkinter;
- виводить зображення у віджет `Label`, прикріплюючи його як графічний елемент.

```

def load_image(label):
    file_path = filedialog.askopenfilename(filetypes=[("Image Files", "*.png;*.jpg;*.jpeg")])
    if file_path:
        image = Image.open(file_path)
        image.thumbnail((400, 400))
        photo = ImageTk.PhotoImage(image)
        label.config(image=photo)
        label.image = photo
        return file_path
    return None

```

Рисунок 3.7 – Функція `load_image(label)`

3.3.2 Load_img(path)

Функція `load_img(path)` використовується для завантаження зображення з вказаного шляху у градаціях сірого. Вона працює з бібліотекою OpenCV та служить для підготовки зображень у початковому вигляді, без будь-яких додаткових обробок, рисунок 3.8. Основні дії функції:

- завантажує зображення за допомогою `cv2.imread` у відтінках сірого. Використовується параметр `cv2.IMREAD_GRAYSCALE`, який дозволяє отримати чорно-біле зображення;
- перевіряє, чи було успішно завантажене зображення. Якщо зображення не знайдене або шлях неправильний, функція генерує помилку `FileNotFoundError`.

```
def load_img(path):
    image = cv2.imread(path, cv2.IMREAD_GRAYSCALE)
    if image is None:
        raise FileNotFoundError(f"Не вдалося завантажити зображення: {path}")
    return image
```

Рисунок 3.8 – Функція `load_img(path)`

3.3.3 Load_images()

Ця функція відповідає за одночасне завантаження двох зображень. Вона викликає `load_image(label)` для кожного зображення, а шляхи до завантажених файлів зберігаються у глобальних змінних `image1_path` і `image2_path`, рисунок 3.9. Основні дії функції:

- викликає `load_image(label1)` для завантаження першого зображення;
- викликає `load_image(label2)` для завантаження другого зображення;
- зберігає шляхи до файлів у змінні для подальшого використання в інших функціях.

```
def load_images():
    global image1_path, image2_path
    image1_path = load_image(label1)
    image2_path = load_image(label2)
```

Рисунок 3.9 – Функція load_images()

3.3.4 Preprocess_image(image_path)

Ця функція виконує початкову обробку завантаженого зображення для спрощення аналізу. Вона перетворює зображення у градації сірого, а потім застосовує порогову бінаризацію, рисунок 3.10. Основні дії функції:

- завантажує зображення у відтінках сірого за допомогою OpenCV (cv2.imread з параметром cv2.IMREAD_GRAYSCALE);
- перевіряє, чи зображення було успішно завантажено. Якщо ні, викликає помилку;
- застосовує порогову бінаризацію методом cv2.threshold, де всі пікселі з яскравістю понад 127 стають білими (255), інші – чорними (0).

```
def preprocess_image(image_path):
    image = cv2.imread(image_path, cv2.IMREAD_GRAYSCALE)
    if image is None:
        raise FileNotFoundError(f"Не вдалося завантажити зображення: {image_path}")
    _, binary = cv2.threshold(image, 127, 255, cv2.THRESH_BINARY)
    return binary
```

Рисунок 3.10 – Функція preprocess_image(image_path)

3.3.5 Align_images(img1, img2)

Ця функція забезпечує вирівнювання другого зображення щодо першого. Для цього використовуються ключові точки, знайдені алгоритмом SIFT, та матриця гомографії, рисунок 3.11. Основні дії функції:

- створює об'єкт SIFT через cv2.SIFT_create;
- обчислює ключові точки та дескриптори для обох зображень за допомогою detectAndCompute;

- використовує FLANN для пошуку відповідностей між дескрипторами двох зображень;
- застосовує критерій Лоу для відсіву ненадійних збігів;
- обчислює матрицю гомографії за допомогою `cv2.findHomography` і RANSAC, що забезпечує стійкість до помилкових збігів;
- повертає матрицю гомографії для подальшого використання.

```
def align_images(img1, img2):
    sift = cv2.SIFT_create()
    kp1, des1 = sift.detectAndCompute(img1, None)
    kp2, des2 = sift.detectAndCompute(img2, None)

    index_params = dict(algorithm=1, trees=5)
    search_params = dict(checks=50)
    flann = cv2.FlannBasedMatcher(index_params, search_params)

    matches = flann.knnMatch(des1, des2, k=2)

    good_matches = []
    for m, n in matches:
        if m.distance < 0.7 * n.distance:
            good_matches.append(m)

    matched_image = cv2.drawMatches(img1, kp1, img2, kp2, good_matches, None, flags=cv2.DrawMatchesFlags_NOT_DRAW_SINGLE_POINTS)
    cv2.imshow("Співпадання ключових точок", matched_image)

    if len(good_matches) < 4:
        raise ValueError("Недостатньо збігів для вичислення гомографії.")

    src_pts = np.float32([kp1[m.queryIdx].pt for m in good_matches]).reshape(-1, 1, 2)
    dst_pts = np.float32([kp2[m.trainIdx].pt for m in good_matches]).reshape(-1, 1, 2)

    M, mask = cv2.findHomography(dst_pts, src_pts, cv2.RANSAC, 5.0)

    if M is None:
        raise ValueError("Не вдалося порахувати матрицю гомографії.")
    return M
```

Рисунок 3.11 – Функція `align_images(img1, img2)`

3.3.6 `Apply_align(img1, img2, M)`

Функція трансформує друге зображення за допомогою матриці гомографії, обчисленої в `align_images`, рисунок 3.12. Основні дії функції:

- визначає розміри першого зображення (еталонного);
- застосовує матрицю гомографії до другого зображення через `cv2.warpPerspective`;
- повертає вирівняне зображення.

```
def apply_align(img1, img2, M):
    height, width = img1.shape
    aligned_img = cv2.warpPerspective(img2, M, (width, height))
    return aligned_img
```

Рисунок 3.12 – Функція `apply_align(img1, img2, M)`

3.3.7 Process_images()

Головна функція, найважливіша в програмі, яка поєднує всі попередні методи для виконання аналізу зображень: вирівнювання, обчислення різниць та виявлення змін, рисунок 3.13. Основні дії функції:

- завантажує та обробляє два зображення через `preprocess_image`;
- викликає `align_images` для вирівнювання та `apply_align` для трансформації другого зображення;
- обчислює абсолютну різницю між вирівняними зображеннями за допомогою `cv2.absdiff`;
- виконує порогову бінаризацію та морфологічну обробку для покращення результату;
- шукає контури на обробленому зображенні через `cv2.findContours`;
- фільтрує контури за площею та співвідношенням сторін, виділяючи лише релевантні області;
- малює прямокутники навколо виявлених змін та виводить результати на екран через `cv2.imshow`.

```

def process_images():
    if not image1_path or not image2_path:
        print("Спочатку потрібно завантажити два зображення!")
        return

    img1 = preprocess_image(image1_path)
    img2 = preprocess_image(image2_path)

    img1_original = load_img(image1_path)
    img2_original = load_img(image2_path)

    try:
        m = align_images(img1, img2)
        img2_aligned = apply_align(img1, img2_original, m)

    except Exception as e:
        print(f"Помилка вирівнювання зображень: {e}")
        return

    difference = cv2.absdiff(img1_original, img2_aligned)
    _, difference_binary = cv2.threshold(difference, 140, 255, cv2.THRESH_BINARY)

    kernel = cv2.getStructuringElement(cv2.MORPH_RECT, (3, 3))
    difference_binary = cv2.morphologyEx(difference_binary, cv2.MORPH_CLOSE, kernel)
    difference_binary = cv2.dilate(difference_binary, kernel, iterations=2)

    contours, _ = cv2.findContours(difference_binary, cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_SIMPLE)
    img_with_differences = cv2.cvtColor(img1_original, cv2.COLOR_GRAY2BGR)

    for contour in contours:
        area = cv2.contourArea(contour)
        if 100 < area < 10000:
            x, y, w, h = cv2.boundingRect(contour)
            aspect_ratio = w / h
            if 0.2 < aspect_ratio < 5:
                cv2.rectangle(img_with_differences, (x, y), (x + w, y + h), (0, 0, 255), 2)
                print(f"Координати: ({x}, {y}), Ширина: {w}, Висота: {h}, Площа: {area}")

    cv2.imshow("Накладання зображень", difference_binary)
    cv2.imshow("Різниця", difference)
    cv2.imshow("Виявлені відмінності", img_with_differences)
    cv2.waitKey(0)
    cv2.destroyAllWindows()

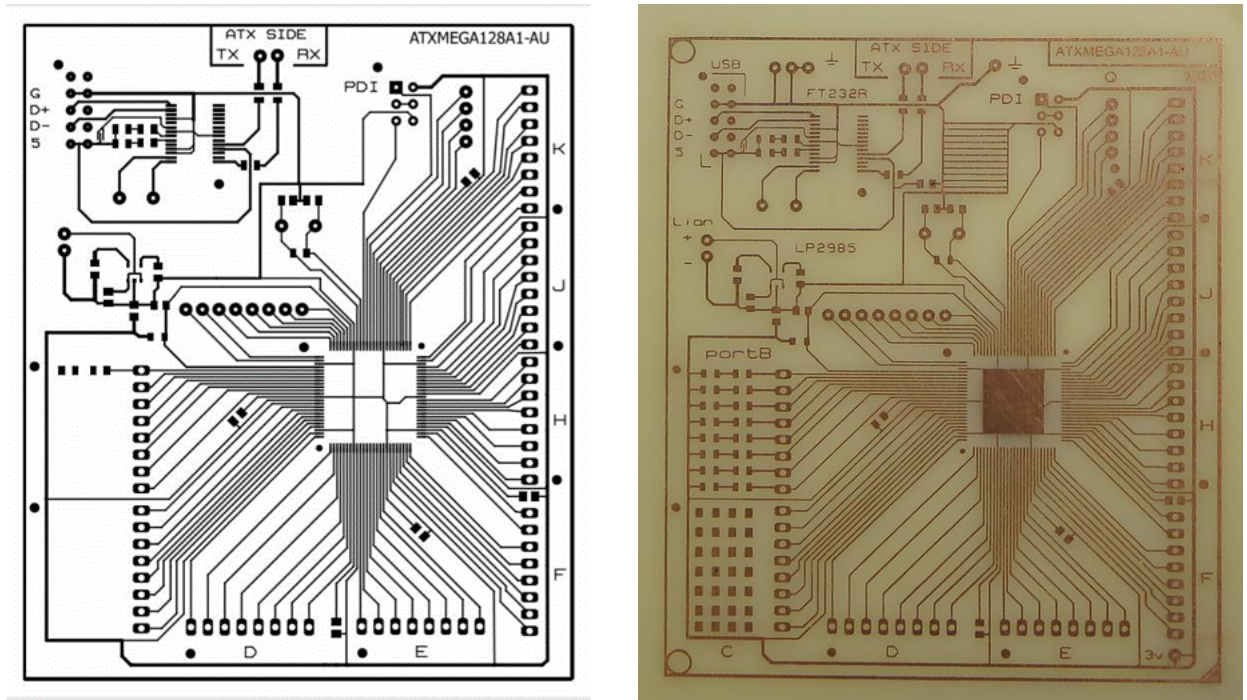
```

Рисунок 3.13 – Функція process_images()

Ці функції утворюють повний робочий процес програми – від завантаження зображень до виявлення змін між ними. Кожна функція має чітко визначене завдання, що забезпечує модульність та зручність у роботі.

3.4 Експериментальне дослідження та порівняння

Завантажимо два зображення в нашу програму, перше зображення – це макет плати, а друге – це вже готова друкована плата, та порівняємо їх. Для першого тесту оберемо поріг – 140, рисунок 3.14.



а)

б)

а – макет плати; б – готова друкована плата

Рисунок 3.14 – Бінаризація зображення

Тепер запустимо нашу програму та відкриємо ці фото. Щоб завантажити їх у програму, потрібно натиснути кнопку “завантажити зображення”, далі обрати два зображення, рисунок 3.15. Перше – це макет нашої плати, а друге – це фото готової плати.

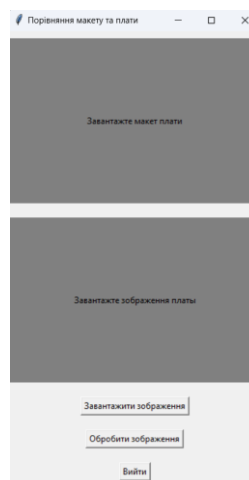
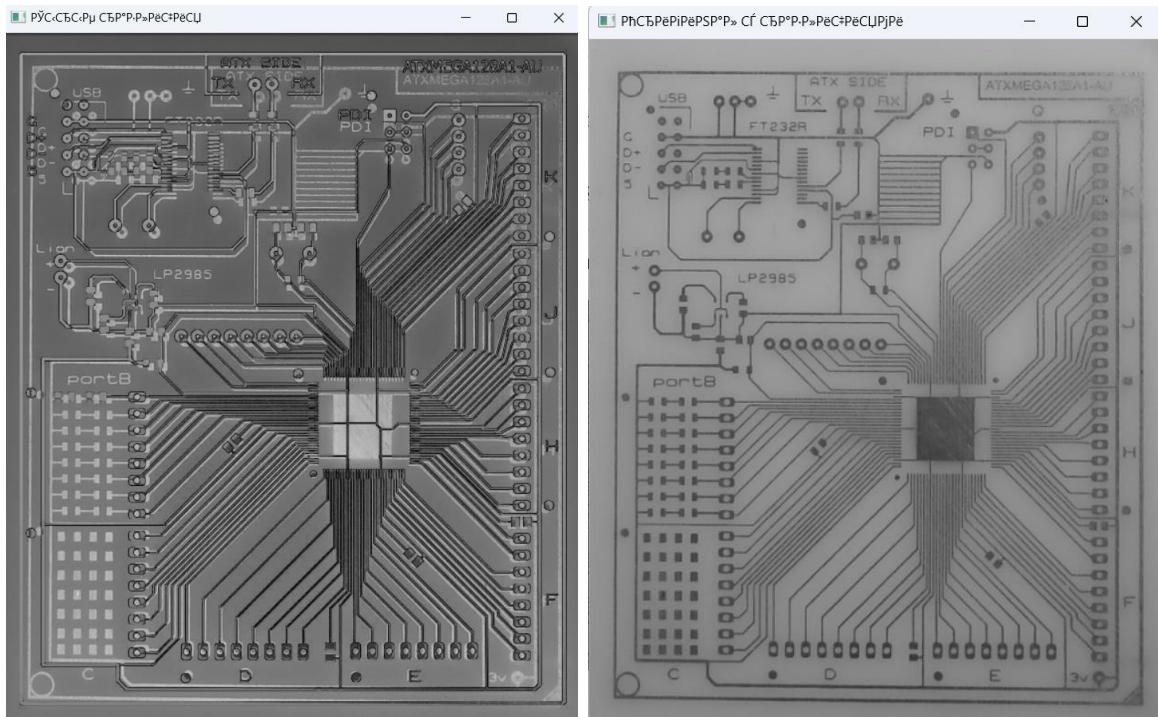


Рисунок 3.15 – Вікно програми

У нас з'явилося два зображення. Перше – це результат накладання макету плати на зображення вже готової плати. Як ми можемо побачити, накладання виявилось поганим, бо контури не співпадають. Друге зображення – різниця між платою та макетом. Програма не знайшла ніякої різниці і виявила помилку, бо зображення не співпадають, через погане вірівнювання, рисунок 3.16.



а)

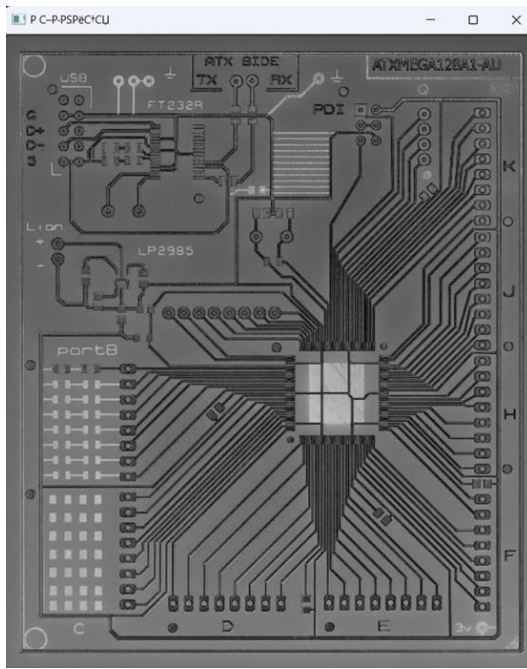
б)

а – накладання та порівняння зображень; б – різниця між макетом та готовою платою

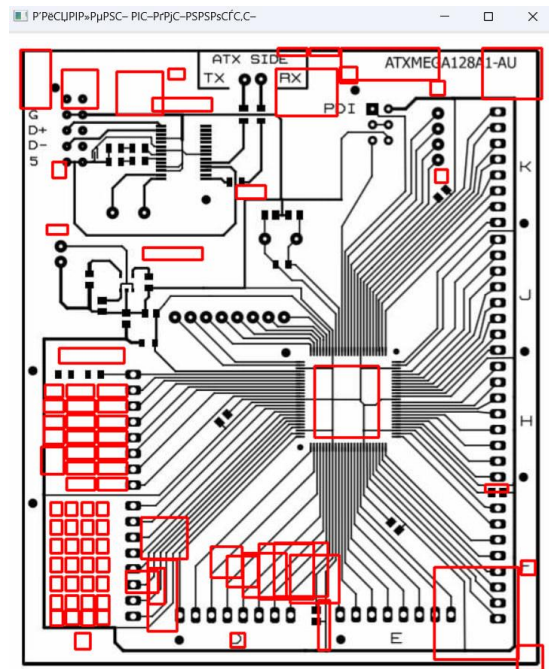
Рисунок 3.16 – Обробка зображень

Спробуємо застосувати алгоритм SIFT та гомографію. Також, додамо ще одно вікно виводу, це буде результат кінцевої бінаризації. Після внесенних змін, запускаємо програму ще раз.

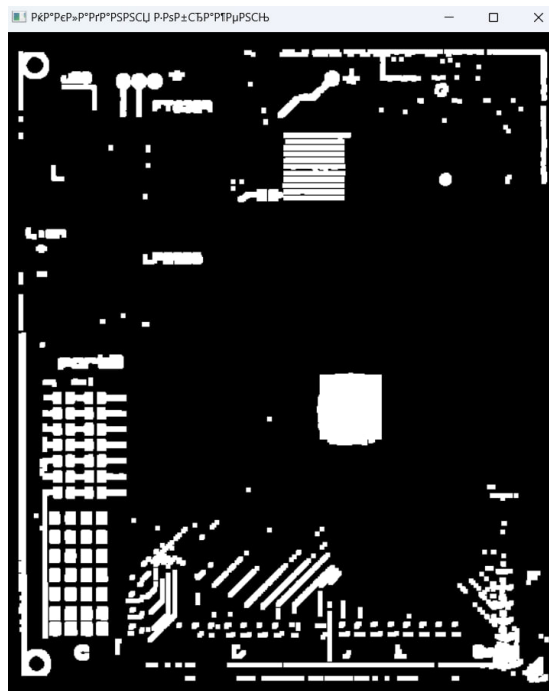
У нас вивелося три зображення. Перше з них, це накладання та порівняння. На зображенні ми можемо побачити сірі ділянки – це різниця між нашим макетом та готовою платою, рисунок 3.17.



а)



б)

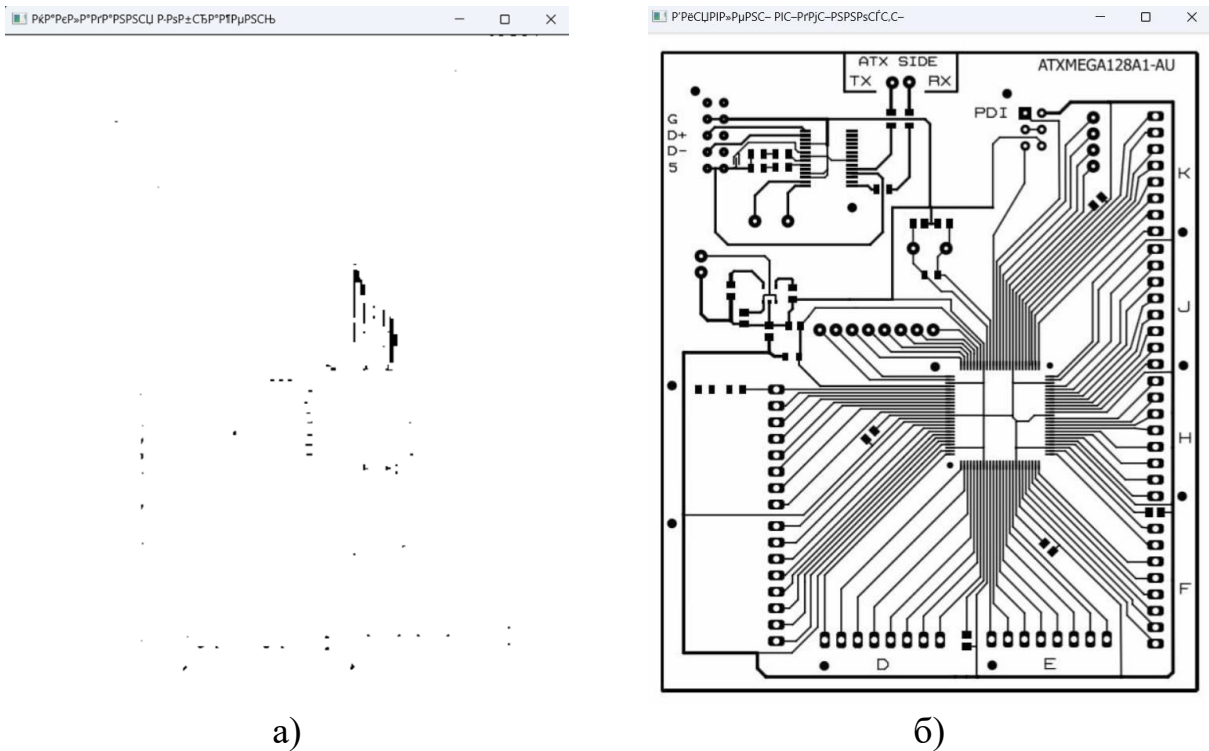


в)

а – накладання та порівняння зображень; б – різниця між макетом та готовою платою; в – результат бінаризації

Рисунок 3.17 – Обробка зображення

Тепер змінимо поріг на 80, та подивимось як програма буде себе поводити, рисунок 3.18.

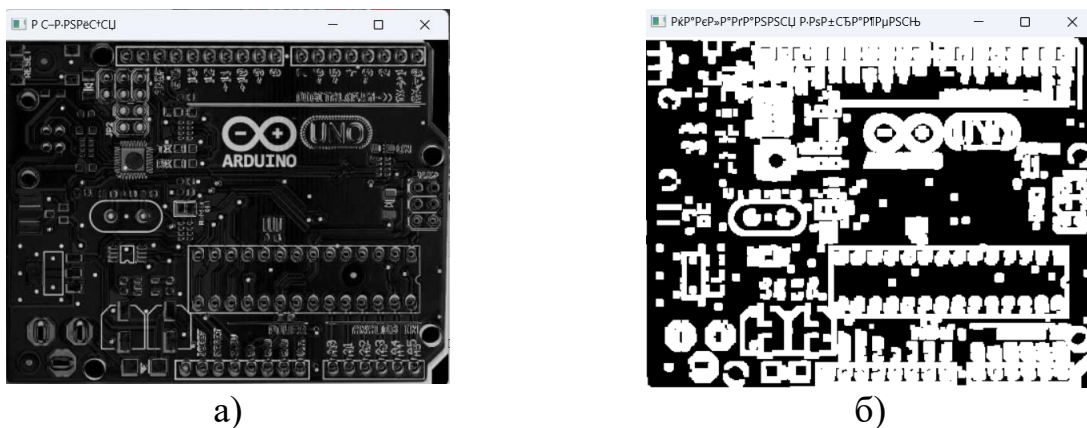


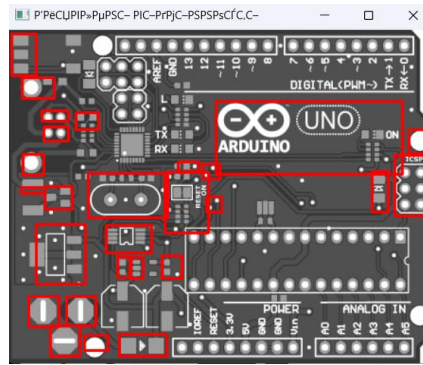
а – результат бінаризації; б – різниця між макетом та готовою платою

Рисунок 3.18 – Бінаризація зображення

З порогом 80 ми можемо побачити, що програма працює неефективно, бо по зображенню бінаризації видно, що вона майже не знайшла відмінностей. Тож для цього зображення, ідеальний поріг бінаризації буде – 140.

Тепер, спробуємо завантажити ще два зображення плати. На цей раз я обрав інший макет та плату, рисунок 3.19. Поріг бінаризації – 80.





в)

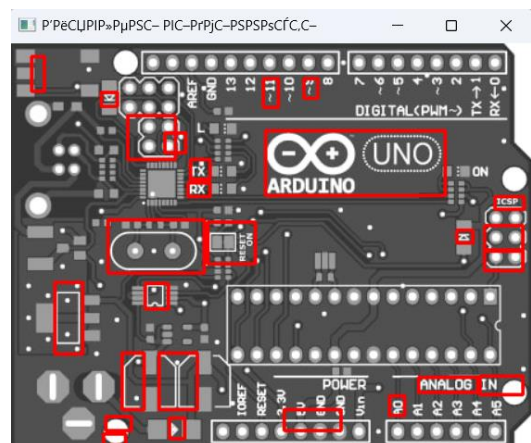
а – накладання зображення макету на готову плату; б – результат бінаризації; в – різниця між макетом та готовою платою

Рисунок 3.19 – Обробка зображення при порозі бінаризації 80

Тепер знову поміняємо поріг на 140, та подивимось, який результат буде, рисунок 3.20.



а)



б)

а – результат бінаризації; б – різниця між макетом та готовою платою

Рисунок 3.20 – Обробка зображення при порозі бінаризації 140

По результатам програми ми бачимо, що вона працює ефективно навіть з зображеннями спеціально не підготовленими для обробки. Незважаючи на різницю масштабу зображень та контрасту, програма все одно чітко змогла накласти одне фото на інше та знайти різницю між макетом та готовою платою.

3.5 Висновки до третього розділу

В ході написання третього розділу були розглянуті методи та бібліотеки OpenCV, для реалізації програмного застосунку. Була написана програма яка дозволяє порівнювати еталоне зображення та топологічної структури та виготовленої топології друкованої плати.

В програмному застосунку були реалізовані наступні функції:

- фільтрація зображення;
- вирівнювання зображення відносно еталону;
- бінаризація та адаптивна бінаризація зображень;
- знаходження ключових точок;
- порівняння зображень між собою.

Особливу увагу було надана проведенню роботи програми та знаходження різниці між зображеннями при різному порозі бінаризації. За результатами досліджень можливо чітко сказати, що для кожної топології необхідно встановлювати власний порог бінаризації.

4 ОХОРОНА ПРАЦІ

4.1 Безпека при виготовленні друкованих плат

Охорона праці є невід'ємною складовою забезпечення безпечних умов роботи при удосконаленні автоматизованої технології обробки зображень топології друкованих плат. Усі роботи виконувалися в навчальній лабораторії розміром 8,3 м × 6 м × 3,1 м, яка відповідає санітарно-гігієнічним вимогам для даного типу приміщень. Робоче місце обладнане столом із персональним комп'ютером (потужність 600 Вт, монітор LCD 24", частота 60 Гц).

У приміщенні працювала одна особа. Площа приміщення становить 49,8 м², об'єм – 154,38 м³, що відповідає вимогам НПАОП 0.00-1.28-10, згідно з якими площа на одне робоче місце має бути не менше 6 м², а об'єм – 20 м³ на людину. Згідно з НПАОП 40.1-1.21-98, приміщення належить до категорії без підвищеної небезпеки (сухе, не заповнене приміщення з ізолюючою підлогою). Використовується система живлення TN-C-S типу, яка забезпечує повторне заземлення нульового проводу відповідно до НПАОП 40.1-1.32-01 [16][17].

Мікрокліматичні умови в приміщенні відповідають нормам, встановленим ДСН 3.3.6.042-99. У холодний період температура повітря підтримується в межах 22–24 °С, вологість – від 40 % до 60 %, а швидкість руху повітря – 0,1 м/с. У теплий період температура становить 23–25 °С, вологість – від 40 % до 60 %, допустима швидкість руху повітря – 0,2 м/с. Освітлення в лабораторії відповідає вимогам ДБН В.2.5-28-2006. Природне освітлення забезпечується через вікна, зорієнтовані на північ, із коефіцієнтом природної освітленості не нижче 1,2 %. Штучне освітлення підтримує рівень освітленості в межах 200–500 лк. Рівень шуму в лабораторії не перевищує 50 дБА, що відповідає ДСН 3.3.6.037-99 та забезпечує комфортні умови для роботи.

Пожежна безпека лабораторії забезпечується відповідно до встановлених норм. Приміщення належить до категорії П–Па за ПУЕ-2011, а ступінь

вогнестійкості будівлі відповідає II класу згідно з ДБН В.1.1-7-2002. За вибухопожежною небезпекою приміщення класифікується як категорія В (НАПБ Б.03.002-2007), оскільки в ньому використовуються тверді горючі матеріали. З метою забезпечення пожежної безпеки встановлено один вогнегасник ВВК-1,4, систему автоматичної пожежної сигналізації з димовими датчиками типу ДИП-1, а також розроблено і розміщено план евакуації персоналу.

Додатково передбачено дотримання правил техніки безпеки при роботі з персональними комп'ютерами, що використовуються під час виконання проєктних робіт. Основні ризики включають можливий перегрів вузлів обладнання, що може призвести до оплавлення ізоляції, а також коротке замикання. Для мінімізації ризиків проводиться регулярний інструктаж персоналу з техніки безпеки, а обладнання підлягає періодичному огляду та обслуговуванню. Усі заходи охорони праці спрямовані на забезпечення безпечних умов роботи та попередження можливих небезпек.

4.2 Висновки до четвертого розділу

Забезпечення безпечних умов праці при виготовленні друкованих плат є комплексним завданням, яке включає технічні, організаційні та освітні заходи. Правильна організація робочого процесу і використання засобів захисту значно зменшують ризик травм і професійних захворювань. Виробництво друкованих плат пов'язане з використанням хімічних речовин (кислоти, луги, розчинники), паянням та обробкою компонентів, які можуть спричинити хімічні опіки, отруєння або травми. Дотримання норм безпеки є критично важливим для запобігання ризикам.

У процесі паяння утворюються шкідливі пари. Використання витяжних систем та засобів захисту органів дихання (респіраторів) дозволяє зменшити негативний вплив на здоров'я працівників. Робітники повинні носити

спеціальний одяг, рукавички, захисні окуляри або щитки для захисту очей та шкіри.

Правильне освітлення, вентиляція і комфортна температура сприяють безпечній роботі та підвищенню ефективності праці. Електробезпека: Під час роботи з електронними компонентами існує ризик ураження електричним струмом. Необхідно дотримуватися правил роботи з електроприладами, використовувати заземлення та перевіряти обладнання на наявність пошкоджень. Працівники повинні регулярно проходити інструктажі з охорони праці, навчання правильному поводженню з обладнанням і матеріалами.

Регулярний технічний огляд обладнання та контроль за дотриманням правил безпеки дозволяють знижувати ймовірність аварійних ситуацій.

ВИСНОВОК

У ході виконання кваліфікаційної роботи було розглянуто основні аспекти та принципи обробки зображень для автоматизованого аналізу друкованих плат. Проведено аналіз існуючих методів обробки та вирівнювання зображень, їх переваг і недоліків, а також можливостей застосування сучасних алгоритмів комп'ютерного зору.

В результаті досліджень та розробки була створена програма для автоматизованого аналізу друкованих плат, яка дозволяє ідентифікувати дефекти шляхом порівняння схеми плати та фактичного зображення. Програма побудована з використанням технологій комп'ютерного зору, таких як SIFT для знаходження ключових точок, FLANN для зіставлення дескрипторів і обчислення матриці гомографії для вирівнювання зображень.

Були реалізовані методи попередньої обробки зображень, включаючи бінаризацію, морфологічну обробку (дилатацію, закриття) та аналіз контурів, що дозволяє виявляти відмінності між двома зображеннями з високою точністю. Розроблений алгоритм дозволяє автоматизувати процес виявлення дефектів, забезпечуючи ефективність роботи та точність результатів.

У роботі було створено зручний графічний інтерфейс користувача на основі бібліотеки Tkinter, який дозволяє легко завантажувати зображення, обробляти їх та отримувати результати аналізу у візуальному форматі. Інтерфейс забезпечує зручність використання, а програма працює стабільно та ефективно.

Результати роботи підтверджують досягнення поставлених цілей та завдань. Розроблена програма є ефективним інструментом для автоматизованого аналізу друкованих плат, дозволяючи знаходити дефекти та покращувати якість виробництва. Застосовані методи та алгоритми забезпечують високу точність і стабільність роботи, а використаний програмний підхід відкриває можливості для подальшого розвитку та вдосконалення системи.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. ДСТУ 3008: 2015. Документація. Звіти у сфері науки і техніки. Структура і правила оформлення [Текст]. – Введ. 2016-01-01. – К.: Держстандарт України, 2015. – 29 с.
2. Методичні вказівки з підготовки та захисту кваліфікаційної роботи здобувачами другого (магістерського) рівня вищої освіти спеціальності 172 Телекомунікації та радіотехніка, освітньої програми «Інтелектуальні технології засобів радіоелектроніки» / Упоряд. Н. П. Демська, В.В. Євсєєв, О. М. Замірець, В. В. Невлюдова, Ю. М. Олександров. – Харків : ХНУРЕ, 2023. – 42 с.
3. Положення про академічну доброчесність [Електронний ресурс]: Наказ ХНУРЕ від 02 лютого 2021 р. No 50. – Режим доступу: https://nure.ua/wp-content/uploads/Main_Docs_NURE/polozhennja-pro-akademichnu-dobrochestnist.pdf
4. Ахмад А. М. Автоматичне визначення області сканування в адаптивній бінарзації зображення / А. М. Ахмад // Automation and Development of Electronic Devices (ADED'2023) : collection of Students' Scientific Paper. – [Electronic edition]. – Kharkiv : Kind of Kharkiv National University of Radio Electronics, 2023. – Part 1. – P. 322-325.
5. OpenCV Documentation [Електронний ресурс]. – Режим доступу: <https://docs.opencv.org>
6. Tkinter Library Documentation [Електронний ресурс]. – Режим доступу: <https://docs.python.org/3/library/tkinter.html>
7. NumPy Documentation [Електронний ресурс]. – Режим доступу: <https://numpy.org/doc/>
8. Pillow (PIL) Documentation [Електронний ресурс]. – Режим доступу: <https://pillow.readthedocs.io>

9. Efficient Retrieval of Images with Irregular Patterns using Morphological Image Analysis: Applications to Industrial and Healthcare datasets / Zhang J., Cosma G., Bugby S., Watkins J. – Arxiv.org, 2023. – 30 с.
10. Bradski G., Kaehler A. Learning OpenCV: Computer Vision with the OpenCV Library. – O'Reilly Media, Inc., 2020. – 576 с.
11. Richard Szeliski. Computer Vision: Algorithms and Applications. – Springer, 2022. – 925 с.
12. Tkinter GUI Programming by Example / David Love. – Packt Publishing, 2018. – 330 с.
13. 1SIFT, FLANN, Homography in Computer Vision [Електронний ресурс]. – Режим доступу: <https://learnopencv.com/>
14. Застосування методів морфологічного оброблення у комп'ютерному стереозорі / Прокопчук О.М., Вовк С.М. – Стаття "Системні технології", вип. 6 (131), 2020. – 16 с.
15. Ахмад А. М. Автоматичне визначення області сканування в адаптивній бінарзації зображення / А. М. Ахмад // Automation and Development of Electronic Devices (ADED'2023) : collection of Students' Scientific Paper. – [Electronic edition]. – Kharkiv : Kind of Kharkiv National University of Radio Electronics, 2023. – Part 1. – P. 322-325.
16. Методичні вказівки до виконання розділу "Охорона праці" у випускних роботах ОКР "магістр" усіх форм навчання / упоряд.: В. А. Айвазов, Т. Є. Стищенко, Н. Л. Березуцька ; М-во освіти і науки України, ХНУРЕ.
17. ДСТУ 7237:2011. Система стандартів безпеки праці. Електробезпека. Загальні вимоги та номенклатура видів захисту.