

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Автоматики і комп'ютерних технологій
(повна назва)

Кафедра Комп'ютерно-інтегрованих технологій, автоматизації та робототехніки
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА

Пояснювальна записка

Перший (бакалаврський)
(рівень вищої освіти)

Розробка програмного забезпечення для ідентифікації виробів на
контрольних операціях технологічних процесів
(тема)

Виконав:

студент 3 прискореного курсу,
групи АКТАКІТу-21-1
Єчевський А. Д.
(прізвище, ініціали)

Спеціальності 151 Автоматизація
та комп'ютерно-інтегровані технології
(код і повна назва спеціальності)

Тип програми Освітньо-професійна
(освітньо-професійна або освітньо-наукова)

Освітня програма Автоматизація та
комп'ютерно-інтегровані технології
(повна назва освітньої програми)

Керівник доц. Сотник С.В.
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри КІТАР _____
(підпис)

Невлюдов І.Ш.
(прізвище, ініціали)

2024 р.

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

Факультет _____ АКТ _____
Кафедра _____ КІТАР _____
Рівень вищої освіти _____ перший (бакалаврський) _____
Спеціальність _____ 151 Автоматизація та комп'ютерно інтегровані технології _____
Тип програми _____ Освітньо-професійна _____
Освітня програма _____ Автоматизація та комп'ютерно-інтегровані технології _____

ЗАТВЕРЖДУЮ:

Зав. кафедри _____

(підпис)

« ____ » _____ 2024 р.

**ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

студентові _____ Єчевському Анатолію Дмитровичу _____
(прізвище, ім'я, по батькові)

1. Тема роботи _____ Розробка програмного забезпечення для ідентифікації виробів
на контрольних операціях технологічних процесів _____

Затверджена наказом по університету від _____ 20.05.2024 №477 Ст.

2. Термін подання студентом роботи до екзаменаційної комісії _____ 11.06.2024

3. Вхідні дані до роботи _____

3.1 ESP32, мікроконтролер для розробки проекту _____

3.2 CMOS-камера OV2640 _____

3.3 Плата програмування на базі CH340G _____

4. Перелік питань, що потрібно розглянути у роботі _____

4.1 Вступ _____

4.2 Аналіз технічного завдання _____

4.3 Розробка структури і алгоритмів системи ідентифікації _____

4.4 Розробка програмного забезпечення ідентифікацій виробів _____

4.5 Охорона праці _____

4.6 Висновки та перелік джерел посилань _____

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій Демонстраційний матеріал, представлений у форматі презентації PowerPoint (*.pptx) – 15 с.

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 до кваліфікаційної роботи	01.03.2024	Виконано
2	Аналіз типового виробничого процесу виготовлення друкованих плат	10.03.2024	Виконано
3	Аналіз процесу збору даних про вироби на різних етапах виробництва	20.03.2024	Виконано
4	Огляд існуючих методів ідентифікації об'єктів	28.03.2024	Виконано
5	Розробка структури і алгоритмів системи ідентифікації	15.04.2024	Виконано
6	Вибір апаратних модулів	17.04.2024	Виконано
7	Вибір середовища розробки програмного забезпечення ідентифікації	19.04.2024	Виконано
8	Розробка макету програмного забезпечення	05.06.2024	Виконано
9	Оформлення звіту з пройденої роботи	10.06.2024	Виконано

Дата видачі завдання 01.03.2024

Студент _____
(підпис)

Єчевський А. Д.
(прізвище, ініціали)

Керівник роботи _____
(підпис)

доц. Сотник С. В.
(посада, прізвище, ініціали)

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

" 10 " червень 2024 р.



Єчевський А. Д.

РЕФЕРАТ

Пояснювальна записка: 96 с., 2 табл., 32 рис., 3 дод., 35 джерел.

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ, ІДЕНТИФІКАЦІЯ ОБ'ЄКТІВ, РОЗПІЗНАВАННЯ, НЕЙРОННА МЕРЕЖА, КОНВЕЄРНА ЛІНІЯ, ARDUINO IDE, ESPRESSIF, ESP32-CAM.

Мета роботи – підвищення точності та надійності контролю якості продукції.

Об'єкт розробки – процес ідентифікації виробів на промисловій лінії.

Предмет розробки – програмне забезпечення для розпізнавання та ідентифікації виробів на промисловій лінії.

Було проведено детальний аналіз типового виробничого процесу виготовлення друкованих плат. Також було проведено огляд існуючих методів ідентифікації об'єктів. Аналізувалися класичні методи, такі як використання штрих-кодів та RFID-міток, а також статистичні методи, які базуються на обробці великих обсягів даних для підвищення точності ідентифікації.

На основі отриманих даних було розроблено структуру програмного забезпечення для ідентифікації виробів. Структура включала модулі для збору даних, їх попередньої обробки, аналізу та ідентифікації.

Розробку програмного забезпечення ідентифікації виробів було розпочато з вибору апаратних модулів для реалізації системи ідентифікації. Зокрема, було обрано платформи Arduino IDE та Espressif ESP32-CAM, які забезпечують необхідну функціональність для збору та обробки зображень. Також було вибрано середовище розробки програмного забезпечення. На основі обраних апаратних модулів та середовища розробки було створено макет програмного забезпечення ідентифікації виробів на контрольних операціях технологічних процесів.

ABSTRACT

Explanatory note: 96 p., 2 tables, 32 figures, 3 appendices, 35 sources.

SOFTWARE, OBJECT IDENTIFICATION, RECOGNITION, NEURAL NETWORK, CONVEYOR LINE, ARDUINO IDE, ESPRESSIF, ESP32-CAM.

Purpose of development – to improve the accuracy and reliability of product quality control.

The object of development is the process of identifying products on an industrial line.

The subject of development is software for recognizing and identifying products on an industrial line.

A detailed analysis of the typical production process of manufacturing printed circuit boards was carried out. We also reviewed existing methods of object identification. Classical methods such as barcodes and RFID tags were analyzed, as well as statistical methods based on processing large amounts of data to improve identification accuracy.

Based on the data obtained, a software structure for product identification was developed. The structure included modules for data collection, pre-processing, analysis, and identification.

The development of product identification software began with the selection of hardware modules for the implementation of the identification system. In particular, the Arduino IDE and Espressif ESP32-CAM platforms were selected, which provide the necessary functionality for image collection and processing. The software development environment was also selected. Based on the selected hardware modules and development environment, a mockup of the software for identifying products at the control operations of technological processes was created.

ЗМІСТ

Перелік скорочень.....	9
Вступ	11
1 Аналіз технічного завдання	15
1.1 Аналіз типового виробничого процесу виготовлення друкованих плат	15
1.1.1 Проектування та підготовка виробництва.....	15
1.1.2 Виготовлення заготовок друкованих плат	16
1.1.3 Нанесення провідного рисунку	17
1.1.4 Металізація отворів та створення контактних площадок.....	19
1.1.5 Нанесення маркування та захисної паяльної маски	20
1.1.6 Встановлення компонентів	21
1.1.7 Контроль якості та тестування	23
1.1.8 Упаковка та відправка	24
1.2 Аналіз процесу збору даних про вироби на різних етапах виробництва	25
1.3 Огляд існуючих методів ідентифікації об'єктів	27
1.3.1 Класичні методи ідентифікації.....	28
1.3.2 Статистичні методи ідентифікації	31
2 Розробка структури і алгоритмів системи ідентифікації.....	33
2.1 Розробка структури програмного забезпечення для ідентифікації виробів.....	33
2.2 Стабілізація зображення.....	38
2.3 Обґрунтування методу збору даних.....	42

	8
2.4 Збір та аналіз даних	46
2.5 Розробка алгоритму ідентифікації виробів на основі нейронної мережі MobileNetV2.....	50
3 Розробка програмного забезпечення ідентифікацій виробів	58
3.1 Вибір апаратних модулів.....	58
3.2 Вибір середовища розробки програмного забезпечення ідентифікації	65
3.3 Розробка макету програмного забезпечення.....	74
3.4 Охорона праці.....	85
Висновки	90
Перелік джерел посилання.....	92
Додаток А Лістинг файлу esp32.ino	97
Додаток Б Лістинг файлу esp32_identification.h	105
Додаток В Демонстраційний матеріал.....	107

ПЕРЕЛІК СКОРОЧЕНЬ

- ГДР – гранично допустимі рівні;
- ЗІЗ – засоби індивідуального захисту;
- ОС – операційна система;
- САПР – система автоматизованого проектування;
- ТАУ – теорія автоматичного управління;
- ШІ – штучний інтелект;
- ADC – Analog-to-Digital Converter – аналого-цифровий перетворювач;
- AIDC – Automatic Identification and Data Capture – системи автоматичної ідентифікації та збору даних;
- AOI – Automatic Optical Inspection – автоматичний оптичний контроль;
- AOT – Automatic Optical Testing – автоматизоване оптичне тестування;
- BLE – Bluetooth Low Energy – Bluetooth з низьким енергоспоживанням;
- BR – Basic Rate – базова швидкість;
- CNN – Convolutional Neural Network – згорткові нейронні мережі;
- DAC – Digital-to-Analog Converter – цифро-аналоговий перетворювач;
- DFD – Data Flow Diagram – діаграма потоків даних;
- DSC – Depthwise Separable Convolutions – глибинно-роздільні згортки;
- EDR – Enhanced Data Rate – підвищена швидкість передачі даних;
- ESP-IDF – Espressif IoT Development Framework;
- I2C – Inter-Integrated Circuit – міжінтегральна схема;
- IDE – Integrated Development Environment – інтегроване середовище розробки;
- IEEE – Institute of Electrical and Electronics Engineers – інститут інженерів з електротехніки та електроніки;
- IoT – Internet of Things – інтернет речей;
- KPI – Key Performance Indicators – ключові показники ефективності;

MES – Manufacturing Execution System – система управління виробництвом;

MVSC – Microsoft Visual Studio Code;

PWM – Pulse Width Modulation – широтно-імпульсна модуляція;

ReLU – Rectified Linear Unit – зрізаний лінійний вузол, іноді випрямлений лінійний вузол;

RFID – Radio Frequency IDentification – радіочастотна ідентифікація;

RGB – Red, Green, Blue – червоний, зелений, синій колірні канали;

SCCB – Serial Camera Control Bus – послідовна шина керування камерою;

SMT – Surface Mount Technology – технологія поверхневого монтажу;

SPI – Serial Peripheral Interface – послідовний периферійний інтерфейс;

SRAM – Static Random Access Memory – статична оперативна пам'ять з довільним доступом;

THT – Through-Hole Technology – технологія монтажу в отвори;

UART – Universal Asynchronous Receiver-Transmitter – універсальний асинхронний приймач-передавач;

Wi-Fi – Wireless Fidelity – бездротова мережа високої точності;

YCbCr – Luminance, Chroma Blue, Chroma Red – колірна модель, в якій колір представляється як 3 компоненти: яскравість; компонент кольоровості, який представляє різницю між синьою компонентою і яскравістю; компонент кольоровості, який представляє різницю між червоною компонентою і яскравістю;

YUV – Luminance, Chrominance Blue, Chrominance Red – колірна модель, в якій колір представляється як 3 компоненти: яскравість; компонент кольоровості, який представляє різницю між синьою компонентою і яскравістю; компонент кольоровості, який представляє різницю між червоною компонентою і яскравістю.

ВСТУП

У сучасному індустріальному виробництві, що стрімко розвивається, велика увага приділяється питанням автоматизації та оптимізації виробничих процесів. Одним із ключових аспектів є ідентифікація виробів на промислових лініях, що відіграє значущу роль у забезпеченні високого рівня якості продукції, контролю за виробничими потоками та підвищенні продуктивності підприємства. Невірне розпізнавання або відсутність системи ідентифікації може викликати великі втрати та проблеми з якістю виробів.

Актуальність теми полягає в тому, що в умовах зростаючого обсягу виробництва та конкуренції, підприємства повинні вдосконалювати свої технологічні процеси для забезпечення високої якості продукції. В Україні та за її межами виробництва різноманітних товарів, починаючи від харчової промисловості до важкої та легкої промисловості, стикаються з необхідністю точної і швидкої ідентифікації виробів під час їх виробництва та руху по ланцюгу постачання.

Ідентифікація виробів вимагає комплексного підходу, в якому знаходять своє застосування сучасні технології. Використання штучного інтелекту (ШІ), комп'ютерного зору, машинного навчання та інших передових методів дозволяє створювати системи ідентифікації, які забезпечують високу точність та швидкість в реальному часі.

Важливо відзначити, що цей напрямок досліджень не обмежується тільки питаннями автоматизації. Відповідна ідентифікація виробів може визначити вагому частину системи управління якістю та відслідковування виробничих потоків. Такий комплексний підхід є ключовим фактором у підвищенні ефективності та конкурентоздатності підприємств.

Метою даної кваліфікаційної роботи є підвищення точності та надійності контролю якості продукції. В той же час на меті забезпечення точності і швидкості ідентифікації та невеликої кількості помилок, що в свою

чергу позитивно вплине на оптимізацію виробничих процесів на підприємствах.

Розгляд сучасних виробничих підприємств свідчить про те, що умови глобалізації та зростаючої конкуренції вимагають від підприємств не лише підтримки високих стандартів якості продукції, але й постійного удосконалення технологічних процесів. В цьому контексті ідентифікація виробів на промислових лініях стає ключовим елементом для впровадження стратегій "Індустрії 4.0".

Національні та міжнародні стандарти якості зазначають важливість ефективної системи ідентифікації виробів як головного засобу контролю за виробничим процесом. Промислові підприємства у всьому світі вивчають та впроваджують нові технології, які можуть покращити якість продукції та забезпечити ефективне ведення бізнесу. У цьому контексті Україна, яка входить у світовий ринок, також стикається з викликами та можливостями сучасного виробництва.

Важливим аспектом є також вплив пандемії COVID-19 на виробничі процеси та ланцюги постачання. Умови карантину та обмежень призвели до необхідності швидкої реакції виробництва на зміни у попиті та умовах праці. Використання ефективної системи ідентифікації може стати вирішальним фактором у збереженні стабільності підприємства в умовах несподіваних та непередбачених ситуацій.

Однак існуючі підходи до ідентифікації виробів мають свої обмеження, серед яких можна виділити складність інтеграції з іншими виробничими системами, високі витрати на впровадження та обслуговування, а також обмежену можливість адаптації до різноманітних умов виробництва.

Тому частиною мети даної кваліфікаційної роботи є вирішення вищезазначених проблем шляхом розробки та експериментального впровадження методу ідентифікації, який враховує вимоги сучасного виробництва, зокрема забезпечує точність та швидкість ідентифікації, а також легкість інтеграції та адаптації до різних виробничих умов.

Дослідження спрямоване на вдосконалення технічних та технологічних аспектів ідентифікації виробів з використанням передових інноваційних методів.

Окрім технічних викликів, є також важливі економічні аспекти, які підкреслюють актуальність дослідження. Впровадження ефективної системи ідентифікації може призвести до значного зменшення витрат на виробництво та контроль за якістю, що в свою чергу позитивно позначиться на конкурентоспроможності підприємства в умовах глобального ринку. Результати цього дослідження можуть стати важливим внеском у розвиток сучасних технологій виробництва та сприяти побудові більш стійких та інноваційних підприємств.

Мета роботи – підвищення точності та надійності контролю якості продукції.

Об'єкт розробки – процес ідентифікації виробів на промисловій лінії.

Предмет розробки – програмне забезпечення для розпізнавання та ідентифікації виробів на промисловій лінії.

Для виконання цієї роботи потрібно вирішити наступні завдання:

- проаналізувати типове виробництво процесу виготовлення друкованих плат;
- проаналізувати процес збору даних про вироби на різних етапах виробництва;
- розробити DFD програмного забезпечення для ідентифікації виробів на контрольних операціях технологічних процесів;
- обґрунтувати метод збору даних;
- провести збір та аналіз даних;
- розробити алгоритм ідентифікації виробів на основі нейронної мережі;
- провести вибір апаратних модулів;
- обрати середовище розробки програмного забезпечення ідентифікації;
- розробити макет програмного забезпечення;

- розглянути питання охорони праці;
- оформити кваліфікаційну роботу згідно ДСТУ 3008:2015 [1], а також з методичними вказівками з підготовки й оформлення кваліфікаційної роботи здобувачами першого (бакалаврського) рівня вищої освіти спеціальності 151 Автоматизація та комп'ютерно-інтегровані технології [2].

1 АНАЛІЗ ТЕХНІЧНОГО ЗАВДАННЯ

1.1 Аналіз типового виробничого процесу виготовлення друкованих плат

Виробництво друкованих плат є складним і багатоетапним процесом, який вимагає точності, якості та ефективності на кожному етапі. Далі розглянуто детальніше типовий виробничий процес виготовлення друкованих плат, зокрема материнських плат для персональних комп'ютерів та інших видів електротехнічних застосувань.

1.1.1 Проектування та підготовка виробництва

Проектування та підготовка виробництва друкованих плат є надзвичайно важливим етапом, який закладає основу для успішного виготовлення високоякісних плат. Цей етап включає кілька ключових кроків:

- розробка принципової схеми друкованої плати, яка визначає всі електричні з'єднання між компонентами. Схема розробляється з урахуванням функціональних вимог, специфікацій компонентів та обмежень проекту. Для створення схеми використовуються системи автоматизованого проектування (САПР), такі як Altium Designer, OrCAD, KiCad тощо;

- створення топології плати на основі принципової схеми. Так розробляється фізичне розташування компонентів та трасування провідників на друкованій платі. Під час створення топології враховуються фактори, такі як розміри плати, розташування роз'ємів, вимоги до електромагнітної сумісності, обмеження виробництва тощо. Інженери використовують САПР для розміщення компонентів, трасування провідників, створення полігонів та інших елементів топології;

- перевірка топології на наявність помилок, таких як перетини провідників, недотримання зазорів, порушення правил проектування тощо.

Виконується аналіз цілісності сигналів та електромагнітної сумісності для забезпечення надійної роботи плати. За необхідності топологія оптимізується для покращення характеристик, зменшення розмірів або спрощення виробництва;

– створення файлів Gerber які містять інформацію про розташування провідників, контактних площадок, отворів та інших елементів на кожному шарі плати. Також генеруються файли для трафаретного друку, які визначають розташування та розміри контактних площадок для нанесення паяльної пасти. Створюються файли для свердління отворів, які містять координати та розміри всіх отворів на платі. Підготовлюються файли для виготовлення фотошаблонів, які використовуються для нанесення захисного рельєфу під час виготовлення плат;

– перевірка виробничих файлів на наявність помилок та відповідність вимогам виробництва. Після затвердження виробничі файли передаються на виробництво для виготовлення прототипів або серійних плат.

Етап проектування та підготовки виробництва друкованих плат вимагає тісної співпраці між інженерами-електроніками, інженерами-конструкторами та фахівцями з виробництва. Ретельне виконання цього етапу дозволяє мінімізувати ризики виникнення проблем на подальших етапах виробництва та забезпечити високу якість кінцевого продукту.

1.1.2 Виготовлення заготовок друкованих плат

Етап виготовлення заготовок друкованих плат є одним з початкових і важливих етапів у виробничому процесі. Він починається з вибору матеріалу для виготовлення плат. Найчастіше використовують фольгований діелектрик FR-4, який складається з декількох шарів склотекстоліту, просочених епоксидною смолою, і має з одного або з обох боків мідну фольгу. FR-4 має хороші діелектричні властивості, механічну міцність та термостійкість. Іноді, в залежності від вимог до плати, можуть використовуватися інші матеріали,

такі як полііміди (для гнучких плат), алюміній (для плат з підвищеним тепловідведенням) тощо.

Після вибору матеріалу, з нього нарізають заготовки потрібного розміру. Зазвичай, для оптимізації виробництва, на одній панелі розміщують декілька однакових плат. Це дозволяє обробляти відразу велику кількість плат та зменшити кількість відходів матеріалу. Розмір панелі та кількість плат на ній визначаються з урахуванням розмірів самих плат, технологічних можливостей обладнання та вимог до економічної доцільності.

Нарізка заготовок відбувається на спеціальному обладнанні, такому як гільйотинні ножиці або верстати з ЧПК. Важливо забезпечити точність нарізки та мінімізувати механічні пошкодження матеріалу. Після нарізки, на заготовках свердлять базові отвори, які використовуються для точного позиціонування плати на наступних етапах виробництва. Ці отвори свердлять на спеціальних верстатах, забезпечуючи високу точність їх розташування.

Перед переходом до наступних етапів виробництва, заготовки проходять візуальний контроль на наявність механічних пошкоджень, подряпин, відшарування фольги тощо. Дефектні заготовки відбраковуються, щоб запобігти проблемам на подальших етапах. Також, за потреби, заготовки можуть пройти додаткову обробку, таку як очищення поверхні, знежирення, підготовка до нанесення захисних покриттів тощо.

Етап виготовлення заготовок друкованих плат, хоча і може здаватися простим, є важливим для забезпечення якості кінцевого продукту. Точність розмірів, якість матеріалу та відсутність дефектів на цьому етапі значно впливають на успішність всіх наступних операцій та на надійність готових плат. Тому виробники приділяють значну увагу контролю якості та дотриманню технологічних вимог саме на етапі виготовлення заготовок.

1.1.3 Нанесення провідного рисунку

Етап нанесення провідного рисунку є одним з ключових у виробництві друкованих плат, адже саме він визначає структуру електричних з'єднань на

платі. Існує декілька методів нанесення провідного рисунку, але найбільш поширеним є субтрактивний метод.

Субтрактивний метод полягає у видаленні зайвої мідної фольги з заготовки, залишаючи тільки провідний рисунок. Цей процес починається з нанесення на заготовку фоторезисту – світлочутливого полімерного матеріалу. Фоторезист наноситься рівномірним тонким шаром за допомогою спеціального обладнання, такого як ламінатори або установки занурення. Після нанесення, фоторезист проходить етап попереднього сушіння для видалення залишків розчинника.

Наступним кроком є експонування фоторезисту через фотошаблон. Фотошаблон – це спеціальна плівка або скляна пластина з нанесеним на неї непрозорим рисунком, який відповідає провідному рисунку плати. Під час експонування, ультрафіолетове світло проходить через прозорі ділянки фотошаблону і полімеризує фоторезист на відповідних ділянках заготовки. Ділянки фоторезисту, які були закриті непрозорими частинами фотошаблону, залишаються неполімеризованими.

Після експонування, заготовка проходить етап проявлення. Під час проявлення, неполімеризовані ділянки фоторезисту розчиняються і змиваються спеціальним розчином – проявником. В результаті, на заготовці залишається рисунок з полімеризованого фоторезисту, який відповідає провідному рисунку плати.

Далі відбувається процес травлення. Заготовку занурюють у розчин травника (зазвичай, розчин хлорного заліза або перекису водню з соляною кислотою), який витравлює незахищені фоторезистом ділянки мідної фольги. В результаті, на заготовці залишається тільки провідний рисунок з міді, захищений полімеризованим фоторезистом.

Після травлення, полімеризований фоторезист видаляють з поверхні плати за допомогою спеціальних розчинників або методом плазмового очищення. Таким чином, на поверхні плати залишається тільки провідний рисунок з міді.

Описаний вище субтрактивний метод є найбільш поширеним, але існують і інші методи, такі як адитивний (нарощування провідного рисунку на діелектричній основі) або напівадитивний (комбінація субтрактивного та адитивного методів).

Незалежно від обраного методу, етап нанесення провідного рисунку вимагає високої точності, дотримання технологічних параметрів та ретельного контролю якості. Будь-які дефекти або відхилення на цьому етапі можуть призвести до порушення електричних з'єднань, що в свою чергу вплине на функціональність та надійність готового виробу. Тому виробники приділяють значну увагу контролю якості на даному етапі виробництва.

1.1.4 Металізація отворів та створення контактних площадок

Металізація отворів та створення контактних площадок є важливими етапами у виробництві друкованих плат, особливо для багатошарових конструкцій. Ці процеси забезпечують надійне електричне з'єднання між шарами плати та підготовку поверхні для подальшого монтажу компонентів

Процес металізації отворів починається зі свердління отворів у потрібних місцях згідно з файлом Drill, що був підготовлений на початковому етапі. Для цього використовуються високоточні верстати з ЧПК, які забезпечують необхідний діаметр та розташування отворів. Після свердління, отвори проходять хімічне очищення для видалення залишків смоли, фольги та інших забруднень [3].

Наступним кроком є активація поверхні отворів. Це робиться для покращення адгезії мідного покриття до діелектрика. Активація може виконуватися різними методами, такими як обробка перманганатом калію, вуглекислим газом або плазмою.

Після активації, отвори проходять процес хімічного міднення. Це безелектролітний процес, під час якого на поверхні діелектрика в отворах осаджується тонкий шар міді. Цей шар служить основою для подальшого гальванічного нарощування міді [4].

Гальванічне нарощування міді виконується для досягнення потрібної товщини мідного покриття в отворах та на контактних площадках. Під час цього процесу, плата занурюється в електроліт, і через неї пропускається електричний струм. Іони міді осаджуються на поверхні плати, збільшуючи товщину мідного шару. Товщина покриття контролюється часом процесу та силою струму [5].

Після нарощування міді, на контактні площадки та в отвори наноситься захисне покриття. Найчастіше використовують імерсійне золото або імерсійне олово. Ці покриття захищають мідь від окислення, покращують здатність до пайки та забезпечують надійний електричний контакт.

На завершення, плати проходять контроль якості. Перевіряється товщина мідного покриття, відсутність дефектів (непрокритих ділянок, наростів, відшарувань тощо), а також електрична провідність отворів та контактних площадок.

Металізація отворів та створення контактних площадок є складними процесами, які вимагають точного контролю багатьох параметрів. Але саме ці процеси забезпечують надійність електричних з'єднань в друкованих платах, що є критично важливим для функціонування електронних пристроїв.

1.1.5 Нанесення маркування та захисної паяльної маски

Нанесення маркування та захисної паяльної маски є завершальними етапами у виробництві друкованих плат перед їх відправкою на монтаж компонентів. Ці процеси виконують важливі функції: маркування допомагає у правильному розміщенні компонентів та ідентифікації плат, а паяльна маска захищає провідний рисунок від випадкових замикань під час пайки.

Маркування зазвичай наноситься методом шовкографії. Цей процес починається з виготовлення трафарету з рисунком маркування. Трафарет виготовляється з тонкої сітчастої тканини, на яку наноситься непроникний для фарби рисунок. Ділянки, які повинні бути задруковані, залишаються

відкритими. Трафарет розміщується над платою, і через нього продавлюється спеціальна фарба. Фарба проходить через відкриті ділянки трафарету і осідає на поверхні плати, формуючи рисунок маркування. Після нанесення фарби, плата проходить сушіння та полімеризацію в спеціальних печах [3].

Альтернативою шовкографії є лазерне маркування. При цьому методі, маркування наноситься на поверхню плати за допомогою лазерного променя. Лазер видаляє тонкий шар матеріалу з поверхні, створюючи контрастне зображення. Цей метод забезпечує високу точність та швидкість нанесення маркування, але вимагає більш дорогого обладнання [4].

Після маркування, на плату наноситься захисна паяльна маска. Паяльна маска – це полімерне покриття, яке наноситься на всю поверхню плати, за винятком контактних площадок та інших ділянок, які повинні залишатися відкритими для пайки. Найчастіше використовуються рідкі фотополімерні маски, які наносяться методом трафаретного друку або занурення. Після нанесення, маска проходить етап експонування через фотошаблон, який визначає ділянки, що повинні залишитися незахищеними. Під дією ультрафіолетового світла, фотополімер на експонованих ділянках полімеризується і стає нерозчинним. Неполімеризовані ділянки видаляються за допомогою проявника, оголюючи контактні площадки. Після проявлення, маска проходить остаточне затвердіння під дією високої температури [4].

Паяльна маска не тільки захищає провідний рисунок від замикань, але й забезпечує додаткову механічну та хімічну стійкість поверхні плати. Крім того, маска покращує електроізоляційні властивості та запобігає росту дендритів металу під час експлуатації плати в умовах підвищеної вологості.

1.1.6 Встановлення компонентів

Встановлення компонентів на друковану плату є ключовим етапом у процесі виготовлення електронних пристроїв. Цей етап передбачає монтаж різноманітних електронних компонентів, таких як мікросхеми, резистори,

конденсатори, роз'єми тощо, на відповідні контактні площадки на поверхні плати.

Процес встановлення компонентів може виконуватися як вручну, так і автоматизовано, залежно від складності плати та обсягів виробництва. При ручному монтажі, оператори вручну розміщують компоненти на платі за допомогою пінцетів або вакуумних захватів. Цей метод є трудомістким і використовується переважно для прототипів, дрібносерійного виробництва або для компонентів, які неможливо встановити автоматично.

Для середньо- та великосерійного виробництва використовуються автоматичні лінії поверхневого монтажу (SMT). Ці лінії складаються з декількох машин, які послідовно виконують операції нанесення паяльної пасти, встановлення компонентів та оплавлення пасти.

Процес починається з нанесення паяльної пасти на контактні площадки за допомогою трафаретного друку. Трафарет з отворами, які відповідають розташуванню контактних площадок, розміщується над платою, і через нього продавлюється паяльна паста. Після цього, плата переміщується до установки поверхневого монтажу (Pick-and-Place Machine), що зображена на рисунку 1.1, яка автоматично захоплює компоненти з стрічок або лотків і встановлює їх на відповідні місця на платі з високою точністю та швидкістю [6].

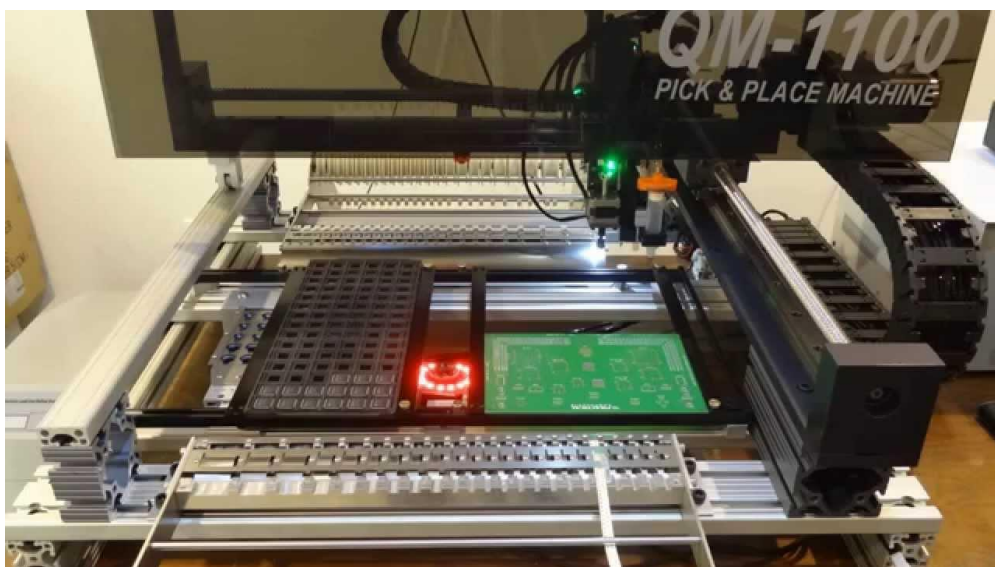


Рисунок 1.1 – QM1100 Automatic Pick and Place Machine [7]

Після встановлення компонентів, плата проходить через піч оплавлення. У печі паяльна паста нагрівається до температури плавлення, створюючи надійне електричне та механічне з'єднання між выводами компонентів та контактними площадками. Профіль нагріву печі ретельно контролюється для забезпечення якісної пайки без пошкодження компонентів.

Для компонентів, які неможливо встановити методом поверхневого монтажу (наприклад, великі роз'єми або компоненти з довгими выводами), використовується монтаж в отвори (ТНТ). При цьому методі, виводи компонентів вставляються в металізовані отвори на платі і припаюються з зворотного боку вручну або за допомогою хвильової пайки [8].

Після встановлення компонентів, плати проходять візуальний та автоматичний контроль якості. Перевіряється правильність встановлення компонентів, якість паяних з'єднань, відсутність дефектів (наприклад, перемичок припою або зміщених компонентів).

1.1.7 Контроль якості та тестування

Контроль якості та тестування є невід'ємною частиною процесу виробництва друкованих плат. Вони гарантують, що друковані плати відповідають необхідним специфікаціям і стандартам. Цей етап включає в себе такі основні операції:

- візуальний огляд на наявність дефектів, таких як неправильно встановлені компоненти, паяльні містки або пошкодження трас;
- автоматизоване оптичне тестування (АОТ) використовує камери та програмне забезпечення для автоматичного виявлення дефектів на друкованих платах. АОТ може виявляти такі дефекти, як нестача припою, зміщення компонентів і неправильне розміщення компонентів;
- функціональне тестування перевіряє функціональність друкованої плати відповідно до специфікації. Це може включати тестування електричних характеристик, таких як напруга, струм і частота;

– випробування на надійність проводяться для оцінки довговічності та надійності друкованої плати. Ці випробування можуть включати термічні удари, вібрацію і випробування на старіння.

Контроль якості та тестування допомагають виявити й усунути дефекти на ранніх етапах виробничого процесу, що знижує ймовірність виходу з ладу друкованих плат в експлуатації.

1.1.8 Упаковка та відправка

Упаковка та відправка є завершальними етапами виробничого процесу виготовлення друкованих плат. Ці етапи забезпечують безпечно транспортування та збереження якості готових виробів під час доставки замовнику.

Процес упаковки починається з візуального огляду плат на наявність дефектів або пошкоджень. Плати, що пройшли перевірку, групуються за замовленнями або партіями. Для запобігання електростатичному розряду, який може пошкодити електронні компоненти, плати упаковуються в антистатичні пакети або контейнери. Ці пакети виготовляються з спеціальних матеріалів, які розсіюють електростатичний заряд і захищають плати від його впливу [9].

Для додаткового захисту від механічних пошкоджень під час транспортування, плати можуть упаковуватися в спеціальні антистатичні коробки або лотки. Ці коробки часто мають внутрішні перегородки або прокладки, які фіксують плати і запобігають їх зміщенню під час перевезення. У випадку особливо чутливих або цінних виробів, можуть використовуватися ударостійкі або водонепроникні контейнери.

Кожна упаковка або контейнер маркується етикеткою, яка містить інформацію про вміст, кількість плат, номер замовлення, дату виготовлення та інші важливі дані. Ця інформація дозволяє легко ідентифікувати та відстежувати вироби під час транспортування та отримання замовником [10].

Перед відправкою, упаковані плати можуть проходити додатковий контроль якості, такий як вибіркова перевірка упаковки або верифікація комплектності замовлення. Після цього, упаковки групуються за замовленнями і передаються до відділу логістики для відправки замовнику.

Після доставки замовнику, процес виробництва друкованих плат вважається завершеним. Однак, виробники зазвичай надають послуги післяпродажної підтримки, такі як гарантійне обслуговування, технічна підтримка або консультації щодо використання виробів.

1.2 Аналіз процесу збору даних про вироби на різних етапах виробництва

Аналіз процесу збору даних про вироби на різних етапах виробництва є важливою складовою розробки методу ідентифікації на промисловій лінії. Збір даних є фундаментальним етапом, оскільки точність та повнота інформації безпосередньо впливають на ефективність розробленого методу [11]. Далі розглянуто різні методи збору даних, які можуть бути використані на різних етапах виробництва.

Існує широкий спектр методів, які використовуються для збору даних про вироби на різних етапах виробництва друкованих плат. Ці методи можна розділити на дві основні категорії: методи прямого збору даних та методи непрямого збору даних.

Методи прямого збору даних передбачають безпосереднє отримання інформації про виріб під час виробничого процесу. Методи непрямого збору даних передбачають отримання інформації про виріб через взаємодію з супутніми системами або документацією. Одним з прикладів непрямого збору даних є використання штрих-кодів або QR-кодів. Кожній платі або партії плат присвоюється унікальний ідентифікатор у вигляді штрих-коду або QR-коду, який містить інформацію про виріб. Цей код може бути відсканований на різних етапах виробництва, що дозволяє отримувати

інформацію про проходження плати через виробничий процес та її поточний стан.

Одним з найпоширеніших методів прямого збору даних є фотографування, тобто використання систем автоматичної ідентифікації та збору даних (AIDC). Ці системи дозволяють автоматично ідентифікувати об'єкти, збирати про них дані та передавати цю інформацію до інформаційних систем підприємства. До таких систем належать штрих-коди, QR-коди, RFID-мітки та інші [12].

На етапі виготовлення друкованих плат, збір даних починається з присвоєння унікального ідентифікатора кожній платі або партії плат. Цей ідентифікатор може бути представлений у вигляді штрих-коду або QR-коду, який наноситься на поверхню плати або на супровідну документацію. Ідентифікатор містить інформацію про тип плати, версію дизайну, дату виготовлення та інші важливі параметри.

Під час виробничого процесу, на кожному етапі, дані про проходження плати через певну операцію можуть збиратися шляхом сканування ідентифікатора та реєстрації відповідної інформації в базі даних. Наприклад, на етапі автоматичного оптичного контролю (AOI), система зчитує ідентифікатор плати і реєструє результати перевірки, такі як кількість виявлених дефектів, їх типи та координати [10].

Для збору більш детальної інформації про процес виготовлення, можуть використовуватися датчики та системи моніторингу. Наприклад, на етапі пайки компонентів, датчики температури та вологості можуть збирати дані про умови пайки для кожної плати. Ця інформація може бути прив'язана до ідентифікатора плати і використана для аналізу та оптимізації процесу.

Окрім автоматичних методів, збір даних може здійснюватися і вручну операторами на різних етапах виробництва. Наприклад, на етапі ручного монтажу компонентів, оператори можуть вводити дані про встановлені компоненти, їх серійні номери та інші параметри в спеціальні форми або

програмне забезпечення. Ця інформація також прив'язується до ідентифікатора плати.

Автоматизовані системи моніторингу виробництва (MES) також є важливим інструментом для непрямого збору даних. Ці системи збирають та інтегрують дані з різних джерел, таких як обладнання, сенсори, системи контролю якості тощо. MES дозволяють отримувати комплексну інформацію про виробничий процес, відстежувати ключові показники ефективності (KPI) та забезпечувати простежуваність виробів на всіх етапах виробництва.

Ще одним методом непрямого збору даних є аналіз записів про виконані операції та результати тестування. Кожна виконана операція та результат тестування реєструються в базі даних або в електронній системі управління документацією. Ці записи містять інформацію про дату та час виконання операції, відповідальну особу, використане обладнання, результати вимірювань та інші важливі дані. Аналіз цих записів дозволяє отримати детальну інформацію про процес виробництва та виявити потенційні проблеми.

Важливо зазначити, що при зборі та використанні даних необхідно забезпечувати їх точність, цілісність та безпеку. Це досягається шляхом використання надійних методів ідентифікації, перевірки даних на етапі вводу, резервного копіювання та захисту баз даних від несанкціонованого доступу.

1.3 Огляд існуючих методів ідентифікації об'єктів

Ідентифікація об'єктів на промислових лініях є ключовим елементом автоматизації виробництва, особливо у сфері виробництва друкованих плат. Цей процес включає в себе розпізнавання та класифікацію об'єктів за допомогою різних технологій, від класичних методів до сучасних підходів машинного навчання.

1.3.1 Класичні методи ідентифікації

Класичні методи ідентифікації об'єктів зазвичай базуються на алгоритмах комп'ютерного зору, які використовують геометричні та статистичні характеристики об'єктів. Наприклад, можуть застосовуватися методи виявлення країв, контурний аналіз, Hough трансформація для виявлення форм, та інші техніки обробки зображень [13].

Одним з основних класичних методів є виявлення країв, яке дозволяє виділити контури об'єктів на зображенні. Для цього можуть використовуватися алгоритми, такі як Canny, Sobel, чи Prewitt. Після виявлення країв застосовується контурний аналіз, який допомагає визначити форму об'єкта та його розміри.

Алгоритм Canny вважається одним з найефективніших методів виявлення країв. Він використовує багатоетапний процес для виявлення широкого спектру країв у зображенні.

Спочатку зображення розмивається за допомогою гауссівського фільтра для зменшення шуму, потім визначаються градієнти зображення, і нарешті застосовується подвійний поріг для визначення сильних і слабких країв.

Приклад роботи алгоритму Canny наведено на рисунку 1.2.

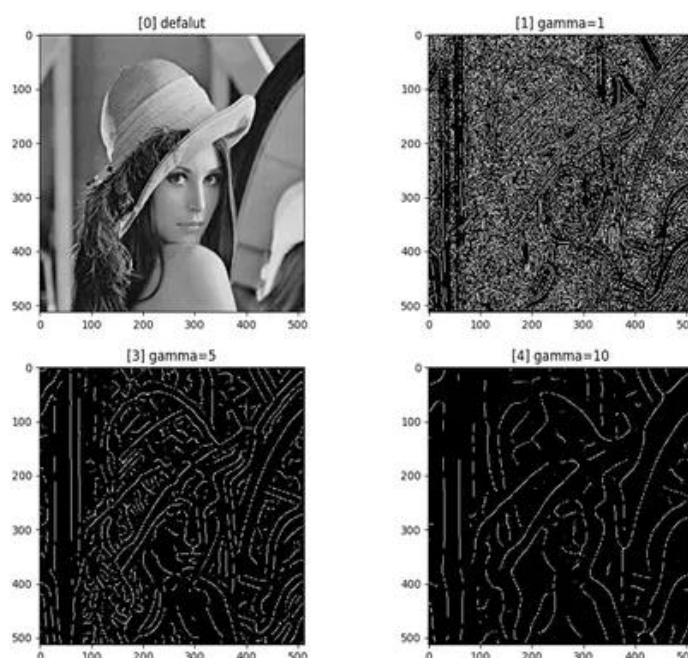


Рисунок 1.2 – Приклад роботи алгоритму Canny [13]

Алгоритм Sobel використовує ядро (маску) для обчислення приблизних значень градієнтів яскравості. Може виявляти краї в горизонтальному та вертикальному напрямках. Часто використовується через свою простоту та ефективність.

Приклад роботи алгоритму Sobel наведено на рисунку 1.3.

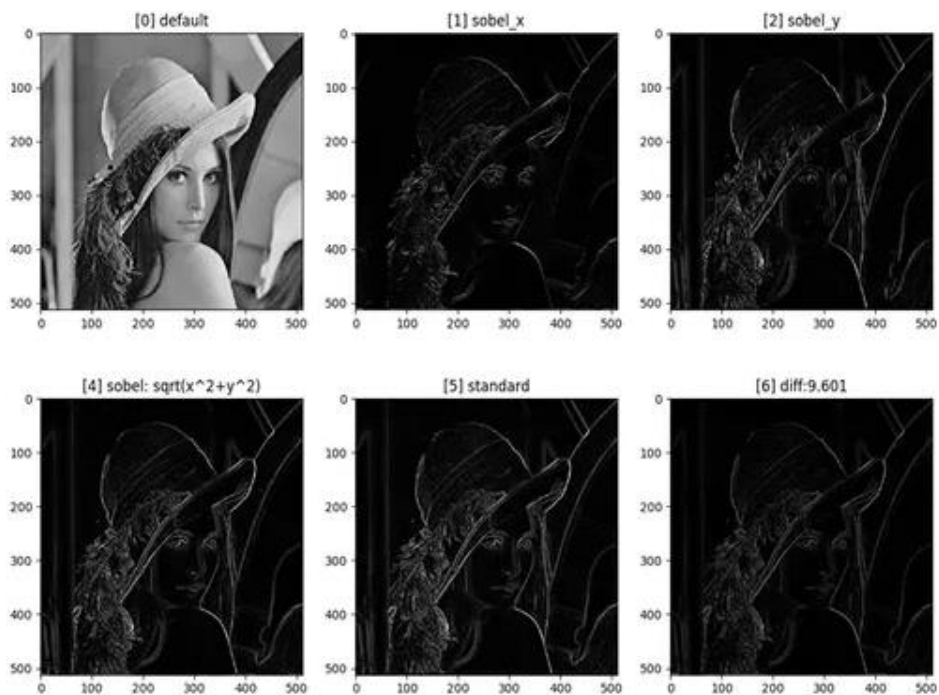


Рисунок 1.3 – Приклад роботи алгоритму Sobel [13]

Алгоритм Prewitt схожий на алгоритм Sobel, але використовує інші ядра для виявлення країв. Не надає переваги пікселям, що знаходяться ближче до центру маски, на відміну від Sobel.

Приклад роботи алгоритму Prewitt наведено на рисунку 1.4.

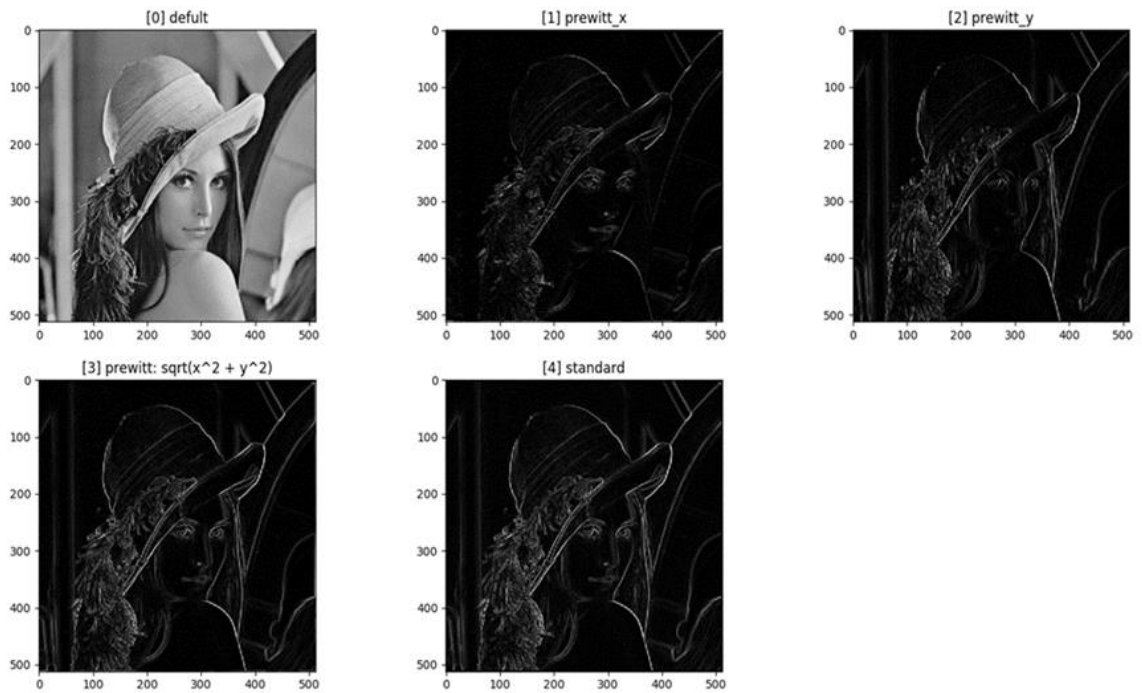


Рисунок 1.4 – Приклад роботи алгоритму Prewitt [13]

Hough трансформація – це метод, що використовується для виявлення простих геометричних форм, таких як лінії, кола, чи еліпси [14]. Цей метод особливо корисний для ідентифікації об'єктів з виразними геометричними характеристиками на конвеєрній лінії, приклад чого й наведено на рисунку 1.5.

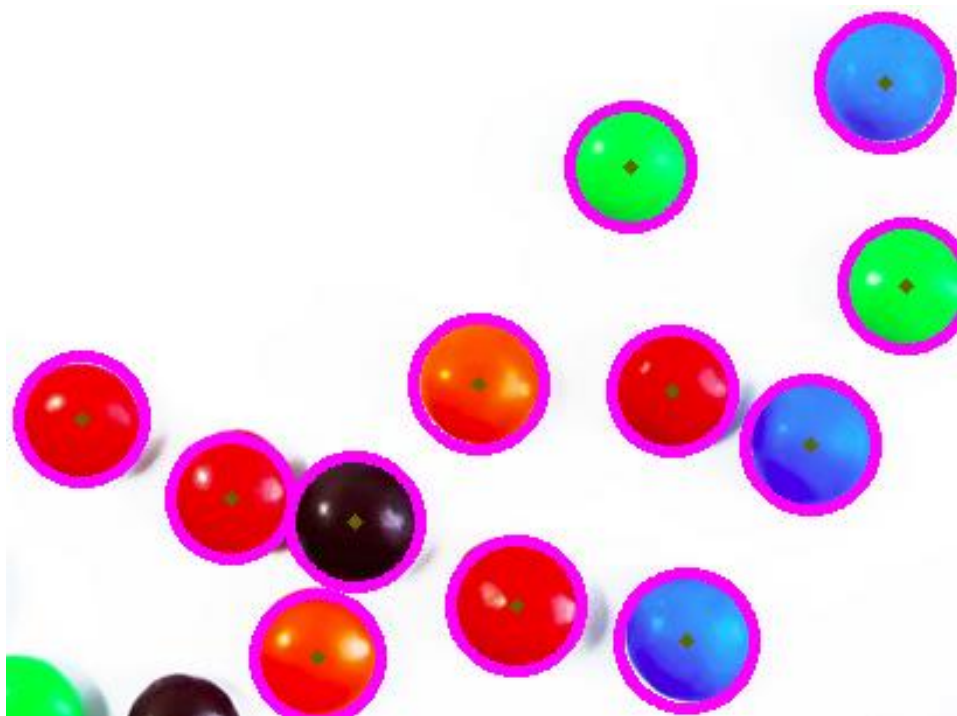


Рисунок 1.5 – Ідентифікація кулькоподібних виробів

Шаблонне відповідання є ще одним класичним методом, який полягає у порівнянні областей зображення з попередньо визначеними шаблонами. Це дозволяє ідентифікувати об'єкти, які мають стандартні розміри та форми, і є ефективним для виробництва, де вироби, такі як друковані плати та їх компоненти, мають уніфіковані характеристики.

1.3.2 Статистичні методи ідентифікації

Статистичні методи, такі як аналіз текстур та гістограми кольорів, використовуються для ідентифікації об'єктів за їхніми візуальними характеристиками. Ці методи можуть бути корисними для визначення якості продукції та виявлення дефектів на ранніх етапах виробництва. Методи базуються на обробці зображень та аналізі даних, що дозволяє автоматизувати процеси та підвищити ефективність виробництва.

Основні статистичні методи включають:

- методи масового спостереження, тобто збір даних про об'єкти на лінії;
- методи зведення та групування базується на класифікації об'єктів за характеристиками;
- методи аналізу рядів розподілу через визначення закономірностей у даних;
- метричні методи використовують вимірювання схожості об'єктів для розпізнавання.

Використання цих класичних методів дозволяє створити надійну систему ідентифікації об'єктів на промисловій конвеєрній лінії, яка може бути інтегрована у виробничий процес для автоматизації та підвищення ефективності виробництва.

Застосування методів машинного навчання значно покращує точність ідентифікації об'єктів завдяки здатності алгоритмів "вчитися" на великих наборах даних і виявляти складні шаблони в даних [15]. Глибоке навчання, зокрема, використовує згорткові нейронні мережі (CNN), які ефективні для

розпізнавання візуальних образів, включаючи текстури та форми елементів на промислових лініях [16].

Використання цих технологій дозволяє автоматизувати процес ідентифікації та забезпечити високу продуктивність та якість на промислових лініях під час виробництва друкованих плат.

2 РОЗРОБКА СТРУКТУРИ І АЛГОРИТМІВ СИСТЕМИ ІДЕНТИФІКАЦІЇ

Даний розділ кваліфікаційної роботи присвячений розробці структури та алгоритмів системи ідентифікації виробів на контрольних операціях технологічних процесів. Цей розділ є ключовим, оскільки в ньому будуть розглянуті основні компоненти та методи, необхідні для створення ефективної системи ідентифікації.

У процесі розробки системи ідентифікації необхідно враховувати специфіку технологічних процесів, характеристики виробів та вимоги до точності та швидкості ідентифікації. Для досягнення поставленої мети потрібно вирішити низку завдань, таких як розробка структури програмного забезпечення, обґрунтування методу збору даних, збір та аналіз даних, розробка алгоритмів розпізнавання та ідентифікації.

2.1 Розробка структури програмного забезпечення для ідентифікації виробів

Для розробки ефективного програмного забезпечення ідентифікації виробів на контрольних операціях технологічних процесів необхідно створити чітку та зрозумілу структуру програмного забезпечення. Одним із найбільш поширених і ефективних методів моделювання та візуалізації структури системи є діаграма потоків даних (DFD).

DFD – це графічний інструмент, який використовується для представлення потоків даних у системі та їх обробки. Діаграма потоків даних дозволяє зрозуміти, як дані рухаються через систему, які процеси їх обробляють та як вони зберігаються. DFD надає чітке і зрозуміле візуальне представлення системи, що робить її незамінним інструментом для аналізу, проектування та комунікації між членами команди розробників та

зацікавленими сторонами проекту. Якщо не приділити достатньо уваги цьому аспекту, можуть виникнути непорозуміння та розбіжності в очікуваннях щодо функціональності та роботи системи.

Аналіз потоків даних між компонентами системи є критично важливим для оптимізації процесів обробки даних. DFD дозволяє виявити надлишкові або неефективні потоки даних, які можуть негативно впливати на продуктивність системи. Якщо не оптимізувати потоки даних, система може страждати від повільної роботи, надмірного використання ресурсів або навіть збоїв.

Ідентифікація потенційних вузьких місць в системі за допомогою DFD є надзвичайно важливою для забезпечення її надійності та ефективності. Вузькі місця можуть призвести до проблем з продуктивністю або якістю даних, що може мати серйозні наслідки для функціонування системи в цілому. Якщо не виявити та не усунути потенційні вузькі місця на ранніх етапах розробки, це може призвести до значних затримок, додаткових витрат та незадоволеності користувачів [17].

DFD складається з чотирьох основних елементів: зовнішні сутності, процеси, сховища даних та потоки даних. Зовнішні сутності представляють собою джерела або одержувачів даних, які знаходяться поза межами системи, що моделюється. Процеси відображають перетворення або обробку даних всередині системи. Сховища даних представляють собою місця, де дані зберігаються для подальшого використання. Потоки даних показують рух даних між процесами, сховищами та зовнішніми сутностями.

Використання DFD для моделювання структури програмного забезпечення для ідентифікації виробів має низку переваг:

– чітке візуальне представлення структури системи, що полегшує розуміння її функціонування, завдяки використанню стандартизованих графічних символів для зображення компонентів системи (джерел даних, процесів обробки, сховищ даних) та потоків даних між ними;

– аналіз потоків даних між компонентами системи, що допомагає оптимізувати процеси обробки даних прослідкувавши шлях даних від їх джерела до кінцевого пункту призначення, виявити зайві або неефективні етапи обробки. Наприклад, можна виявити дублювання обчислень, передачу надлишкової інформації, невиправдано складні алгоритми. Після чого можна оптимізувати алгоритми обробки даних, зменшити кількість операцій, підвищити швидкість роботи системи та зменшити навантаження на апаратні ресурси;

– ідентифікація потенційних вузьких місць в системі, де можуть виникнути проблеми з продуктивністю або якістю даних. Наприклад, це може бути компонент з обмеженою пропускнуою здатністю, складний алгоритм обробки, що потребує значних ресурсів, або ненадійний канал передачі даних. Виявлення вузьких місць на етапі проектування дозволяє заздалегідь вжити заходів для їх усунення: використовувати більш потужні компоненти, оптимізувати алгоритми, забезпечити резервування каналів зв'язку;

– є ефективним засобом комунікації між розробниками, замовниками та іншими зацікавленими сторонами проекту, завдяки чому може бути використана для презентації проекту замовнику, обговорення деталей з тестувальниками, створення документації для користувачів.

DFD діаграма, що наведена на рисунку 2.1, розроблена для програмного забезпечення для ідентифікації виробів на контрольних операціях технологічних процесів. Представлена DFD діаграма є важливим інструментом для розуміння структури і принципів роботи програмного забезпечення для ідентифікації виробів. Вона надає чітке і зрозуміле візуальне представлення системи, дозволяє аналізувати потоки даних, виявляти потенційні проблеми і оптимізувати процеси обробки інформації. Використання DFD діаграми на етапі проектування системи дозволяє створити ефективне і надійне програмне забезпечення, яке відповідає вимогам і очікуванням.

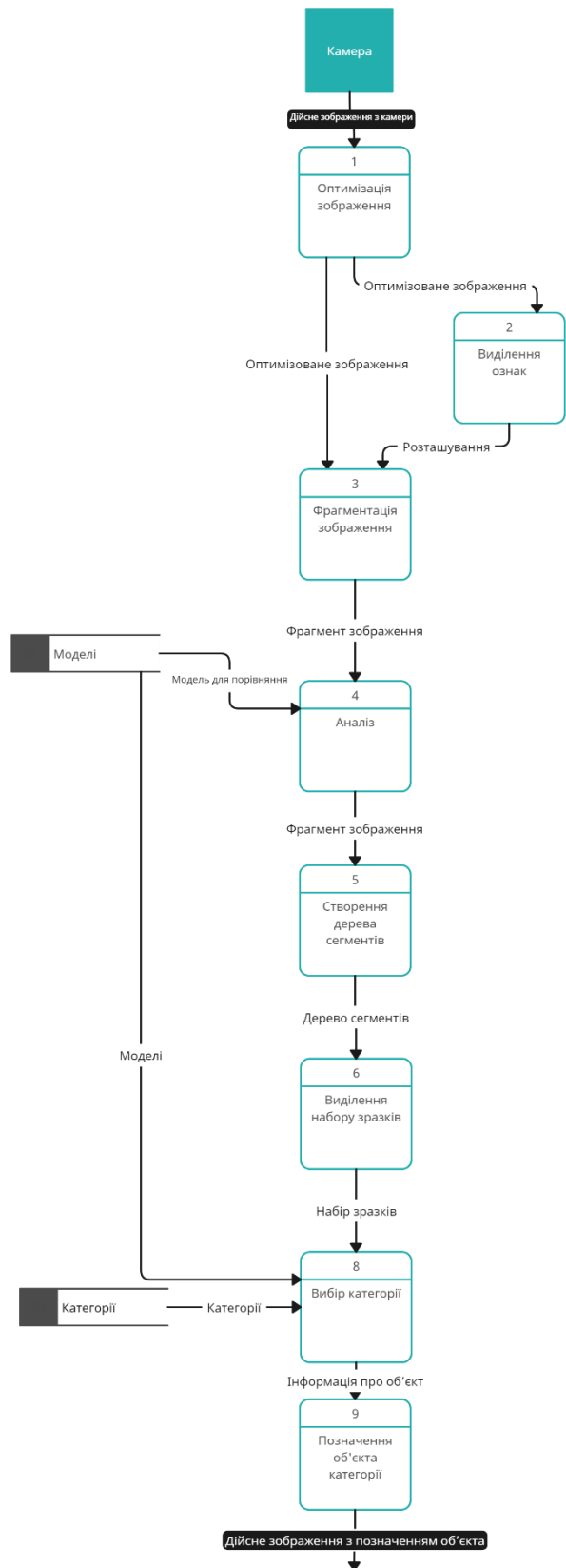


Рисунок 2.1 – DFD діаграма програмного забезпечення

На представленій діаграмі потоків даних зображено основні компоненти системи ідентифікації виробів та потоки даних між ними.

Зовнішньою сутністю є камера, що виконує захват зображення та передачу на обробку програмному забезпеченню.

Процеси, що відбуваються, наведені далі:

- оптимізація зображення відповідає за покращення якості зображення, отриманого з камери. Це може включати корекцію яскравості, контрасту, видалення шумів та інші операції, спрямовані на підвищення чіткості та інформативності зображення для подальшого аналізу;

- виділення ознак, що являє собою аналіз зображення з метою виявлення та виділення характерних ознак, що можуть бути використані для ідентифікації виробу. Це можуть бути геометричні форми, текстури, кольори або інші візуальні характеристики;

- фрагментація зображення для виділення фрагменту з характерною ознакою об'єкта ідентифікації;

- аналіз виділених фрагментів зображення з метою визначення їх характеристик та властивостей;

- створення дерева сегментів, на основі результатів аналізу фрагментів, що відображає взаємозв'язки між різними частинами виробу;

- виділення набору зразків з фрагментів зображення, що містять найбільш інформативні характеристики для ідентифікації;

- вибір категорії, на основі аналізу набору зразків та порівняння з інформацією з бази даних моделей, до якої належить виріб;

- позначення об'єкта категорії відповідним ідентифікатором або назвою.

Сховища даних наступні:

- моделі – сховище, що містить еталонні моделі виробів, з якими порівнюються набори зразків для визначення категорії;

- категорії – сховище, що містить інформацію про доступні категорії виробів для ідентифікації.

Потоки даних між процесами та сховищами перелічені нижче:

- дійсне зображення з камери – потік необроблених даних зображення;

- оптимізоване зображення – потік даних зображення після покращення якості;
- розташування – інформація про розташування ознак для;
- фрагмент зображення з основним набором ознак для ідентифікації об'єкта;
- дерево сегментів – структура, що відображає взаємозв'язки між фрагментами зображення;
- набір зразків;
- інформація про об'єкт – дані про ідентифікований об'єкт, включаючи його категорію та інші характеристики;
- моделі – інформація зі сховища моделей, що використовується для порівняння з характеристиками об'єкта;
- категорії – інформація про категорії та їх характеристики, до яких може належати ідентифікований об'єкт.

Представлена DFD діаграма надає чітке розуміння структури програмного забезпечення для ідентифікації виробів та потоків даних між його компонентами. Це дозволяє ефективно організувати процес розробки системи та забезпечити її надійне функціонування.

2.2 Стабілізація зображення

Для забезпечення високої якості зображень з камери необхідно стабілізувати зображення. Це особливо важливо для пристроїв ідентифікації виробів, де точність розпізнавання залежить від якості зображень. Використання теорії автоматичного управління (ТАУ) дозволяє досягти високого рівня стабілізації за допомогою ПД-регулятора.

ПД-регулятор є одним із найбільш поширених типів регуляторів, який використовується для керування різними процесами. Його основне завдання полягає в тому, щоб мінімізувати відхилення від заданого значення, яке називається бажаним або цільовим значенням [18].

Пропорційний компонент (P) визначає керуючий сигнал, який пропорційний поточній помилці [18].

Помилка ($e(t)$) – це, згідно формули (2.1), різниця між бажаним значенням ($r(t)$) і фактичним значенням ($y(t)$):

$$e(t) = r(t) - y(t), \quad (2.1),$$

де $e(t)$ – помилка (відхилення від бажаного значення);

$r(t)$ – бажане значення;

$y(t)$ – фактичне значення.

Пропорційний компонент визначається за формулою (2.2).

$$P = K_p \times e(t), \quad (2.2)$$

де P – пропорційна частина керуючого сигналу;

K_p – коефіцієнт пропорційності, який визначає, наскільки сильно регулятор реагує на поточну помилку;

$e(t)$ – поточна помилка.

Високе значення K_p може призвести до швидкої реакції на помилки, але може також викликати нестабільність системи та коливання. Занадто низьке значення K_p може зробити систему занадто повільною, що призведе до довшого часу стабілізації.

Інтегральний компонент (I) враховує накопичені помилки з часом, що дозволяє системі усувати залишкові похибки, які можуть виникати через постійне зміщення [18]. Інтегральна частина визначається за формулою (2.3):

$$I = K_i \times \int e(t) dt, \quad (2.3)$$

де I – інтегральна частина керуючого сигналу;

K_i – коефіцієнт інтегрування, який визначає, наскільки сильно регулятор враховує минулі помилки;

$e(t)$ – поточна помилка.

Інтегральний компонент корисний для усунення постійного зміщення, але може призвести до перевищення, якщо коефіцієнт K_i занадто великий. Це може спричинити повільну реакцію системи або навіть нестабільність.

Диференціальний компонент (D) враховує швидкість зміни помилки, що дозволяє регулятору прогнозувати майбутню поведінку системи і діяти на випередження [18]. Диференціальний компонент визначається за формулою (2.4):

$$D = K_d \times \frac{de(t)}{dt}, \quad (2.4)$$

де D – диференціальна частина керуючого сигналу;

K_d – коефіцієнт диференціювання, який визначає, наскільки сильно регулятор реагує на швидкість зміни помилки;

$e(t)$ – поточна помилка;

$\frac{de(t)}{dt}$ – швидкість зміни помилки.

Диференціальний компонент корисний для зменшення коливань і покращення стабільності системи. Однак занадто велике значення K_d може призвести до підсилення шуму в системі, що негативно вплине на стабільність.

Повна формула ПД-регулятора керуючого сигналу, згідно формули (2.5), визначається як сума всіх трьох компонентів:

$$u(t) = K_p \times e(t) + K_i \times \int e(t)dt + K_d \times \frac{de(t)}{dt}. \quad (2.5)$$

Процес налаштування параметрів K_p , K_i , K_d називається тюнінгом ПД-регулятора. Існують різні методи для налаштування параметрів, такі як

метод проб і помилок, метод Зіглера-Нікольса, метод кривих реакції та інші. Метою налаштування є досягнення бажаної динаміки системи, такої як мінімізація перехідного процесу, зменшення перевищення та покращення стабільності [18].

При стабілізації зображення, коли поточне положення об'єкта відхилене від центру зображення. Припустимо, що бажане положення – центр зображення $(0, 0)$, а поточне положення – $(10, 5)$.

Помилка обчислюється за формулою (2.6):

$$e(t) = \sqrt{10^2 + 5^2} = \sqrt{100 + 25} = 125 \cong 11.18. \quad (2.6)$$

Припустимо, що параметри мають наступні значення:

$$- K_p = 1.0;$$

$$- K_i = 0.1;$$

$$- K_d = 0.01.$$

Інтегральна помилка за час Δt становить 2, а швидкість зміни помилки дорівнює 0.5.

Обчислення керуючого сигналу відбувається за формулою (2.5):

$$u(t) = 1.0 \times 11.18 + 0.12 + 0.01 \times 0.5,$$

$$u(t) = 11.18 + 0.2 + 0.005,$$

$$u(t) = 11.385.$$

Таким чином керуючий сигнал $u(t)$ дорівнює 11.385, що буде використовуватися для корекції положення зображення.

Структурна схема контуру автоматичного регулювання для стабілізації зображення наведена на рисунку 2.2. Схема включає наступні компоненти:

- вхідний сигнал;
- помилка;
- ПІД-регулятор;
- стабілізація;
- вихідний сигнал;
- зворотний зв'язок.



Рисунок 2.2 – Структурна схема контуру автоматичного регулювання

Стрілки вказують напрямок сигналів між компонентами, забезпечуючи зрозуміле та організоване відображення системи.

Отже, ПІД-регулятор є потужним інструментом для автоматичного керування, який дозволяє досягти високої точності та стабільності в різних системах. Завдяки правильному налаштуванню його параметрів можна значно покращити якість зображень у системах ідентифікації виробів, що підвищує ефективність і надійність контролю технологічних процесів.

2.3 Обґрунтування методу збору даних

При розробці програмного забезпечення для ідентифікації виробів на контрольних операціях технологічних процесів, вибір методу збору даних є критично важливим. Від правильного вибору методу залежить якість, точність та ефективність роботи всієї системи [19]. Тому далі розглянуто різні методи збору даних, проаналізовано їх переваги та обмеження, та обґрунтовано вибір найбільш підходящого методу для поточної системи.

Існує декілька основних методів збору даних, які можуть бути використані для ідентифікації виробів.

Ручний ввід даних – це метод, при якому оператори вручну вводять інформацію про вироби в систему на контрольних операціях. Цей метод є найпростішим та найдешевшим, оскільки не потребує додаткового обладнання або спеціальних міток на виробках. Оператори можуть вводити різноманітну інформацію, таку як унікальний ідентифікатор виробу, його характеристики, результати контролю якості тощо [19].

Переваги ручного вводу даних:

- простота реалізації через відсутність додаткового обладнання або програмного забезпечення;
- гнучкість, що дозволяє вводити різноманітну інформацію про вироби;
- низька вартість, бо не вимагає значних інвестицій в обладнання або розробку програмного забезпечення.

Недоліки ручного вводу даних:

- висока ймовірність помилок операторами, що знижує точність та надійність системи;
- низька швидкість введення даних, що може стати вузьким місцем у виробничому процесі;
- необхідність постійної присутності операторів для введення даних, що збільшує витрати на оплату праці.

Сканування штрих-кодів – це метод, при якому на вироби наносяться спеціальні штрих-коди, які містять інформацію про вироби. Ці штрих-коди зчитуються за допомогою спеціальних сканерів на контрольних операціях. Штрих-коди можуть містити різноманітну інформацію, таку як унікальний ідентифікатор виробу, його характеристики, дату виробництва тощо [19].

Переваги сканування штрих-кодів наступні:

- швидкість та точність зчитування даних, що зменшує ймовірність помилок;

- можливість автоматизації, що зменшує потребу в ручній праці;
- широке розповсюдження, що робить цей метод доступним та зрозумілим для більшості виробників.

Проте виявлені наступні обмеження сканування штрих-кодів:

- додаткові витрати на обладнання для друку та сканування штрих-кодів, що збільшує витрати на реалізацію системи;
- необхідність нанесення штрих-кодів на кожен виріб, що може бути незручним або неможливим;
- обмеження розміру та якості штрих-кодів, бо штрих-коди повинні мати певний розмір та якість для надійного зчитування, що накладає обмеження на їх розміщення на виробах.

Радіочастотна ідентифікація – це метод, при якому на вироби кріпляться спеціальні RFID-мітки, які містять інформацію про вироби. Ці мітки зчитуються безконтактно за допомогою RFID-зчитувачів на контрольних операціях. RFID-мітки можуть містити більше інформації, ніж штрих-коди, та можуть бути перезаписані в процесі виробництва [19].

Використання RFID-міток має наступні переваги:

- RFID-мітки можуть бути зчитані на відстані без необхідності прямої видимості, що спрощує процес ідентифікації виробів;
- швидкість зчитування дещо вища у порівнянні із попередньо розглянутими методами через те, що RFID-зчитувачі можуть зчитувати інформацію з багатьох міток одночасно, що прискорює процес ідентифікації;
- можливість перезапису RFID-міток в процесі виробництва, що дозволяє оновлювати інформацію про вироби.

Також метод радіочастотної ідентифікації має наступні недоліки:

- висока вартість RFID-обладнання та міток в порівнянні з іншими методами ідентифікації, такими як штрих-коди;
- чутливість до електромагнітних завад від інших пристроїв, що може впливати на надійність зчитування;

– необхідність кріплення міток до кожного виробу, що може бути незручним або неможливим для деяких типів плат.

Візуальний метод (машинний зір) – це метод, при якому для ідентифікації виробів використовуються камери та алгоритми комп'ютерного зору. Камери захоплюють зображення виробів на контрольних операціях, а спеціальне програмне забезпечення аналізує ці зображення для ідентифікації виробів на основі їх візуальних характеристик, таких як форма, розмір, колір, текстура тощо [19].

Даний метод має наступні переваги:

- ідентифікація виробів без необхідності фізичного контакту або нанесення додаткових міток;
- гнучкість налаштування для ідентифікації широкого спектру виробів з різними візуальними характеристиками;
- можливість не тільки ідентифікувати вироби, але й проводити їх візуальний контроль якості.

Обмеження візуального методу:

- необхідність забезпечення належного освітлення;
- чутливість до змін зовнішнього вигляду виробів;
- складність розробки алгоритмів для розпізнавання образів, виявлення дефектів та визначення характеристик виробів за зображеннями, що потребує спеціальних знань та інструментів.

Проаналізувавши переваги та обмеження кожного з методів, для розроблюваної системи ідентифікації виробів на контрольних операціях технологічних процесів обрано візуальний метод збору даних. Цей вибір обґрунтовується наступними факторами:

- універсальність, завдяки можливості ідентифікувати широкий спектр виробів незалежно від їх форми, розміру та матеріалу. Це особливо важливо в умовах, коли номенклатура виробів може змінюватися;

– відсутність необхідності модифікації виробів на відміну від методів з використанням штрих-кодів або RFID-міток. Це спрощує процес ідентифікації та зменшує витрати на додаткові матеріали та обладнання;

– можливість інтеграції з системами контролю якості, тобто проводити візуальний контроль якості виробів. Це дає можливість створити комплексну систему, яка поєднує ідентифікацію виробів з перевіркою їх відповідності заданим параметрам якості;

– гнучкість та адаптивність, бо алгоритми комп'ютерного зору, що використовуються у візуальному методі, можуть бути налаштовані та адаптовані під конкретні потреби виробництва. Це дозволяє враховувати специфіку виробів, умови освітлення та інші фактори, що впливають на якість ідентифікації;

– можливість повної автоматизації процесу ідентифікації виробів, що зменшує вплив людського фактору та підвищує швидкість і точність контролю.

Звісно, використання візуального методу також має певні обмеження та виклики, які необхідно враховувати при розробці системи. Зокрема, потрібно забезпечити належне освітлення та позиціонування виробів для отримання якісних зображень. Незважаючи на це, висновок зроблено, що візуальний метод збору даних є найбільш підходящим для нашої системи ідентифікації виробів на контрольних операціях технологічних процесів. Він забезпечує необхідну гнучкість, універсальність та можливість автоматизації процесу, що дозволить створити ефективну та надійну систему контролю виробництва.

2.4 Збір та аналіз даних

У рамках даної кваліфікаційної роботи важливим етапом є збір та аналіз даних. Метою цього етапу є відстеження конкретних моделей або партій виробів під час виробництва друкованих плат.

Виробництво друкованих плат складається з кількох ключових етапів, на яких можна ідентифікувати вироби та контролювати технологічний процес. Розглянемо ці етапи.

Почнемо з підготовки заготовки (базового матеріалу), що включає декілька важливих процесів:

– перевірка та ідентифікація типу матеріалу, з якого виготовляється друкована плата. Шляхом аналізу зображення можна визначити характерні ознаки різних типів матеріалів, таких як колір, текстура або маркування. Це дозволяє автоматично ідентифікувати тип матеріалу та перевірити його відповідність вимогам. Найчастіше використовуються склотекстоліт або композитні матеріали;

– перевірка товщини та розмірів заготовки. Камера захоплює зображення заготовки з різних ракурсів, а спеціальне програмне забезпечення аналізує отримані дані. Шляхом калібрування системи та застосування алгоритмів обробки зображень можна точно визначити товщину та розміри заготовки і порівняти їх з заданими специфікаціями.

Далі відбувається нанесення рисунка провідників, що може здійснюватися за допомогою фотолітографії або трафаретного друку. На даному етапі важливо контролювати точність та чіткість нанесеного рисунка, оскільки це безпосередньо впливає на функціональність друкованої плати. Камера захоплює детальні зображення поверхні друкованої плати, а програмне забезпечення аналізує їх на наявність дефектів, таких як переривання провідників, короткі замикання або відхилення від заданого рисунка. Автоматизована система оптичного контролю дозволяє швидко виявляти та локалізувати дефекти, що підвищує ефективність контролю якості.

Після відбувається травління, під час якого видаляється зайвий мідний шар, залишаючи лише провідники згідно з рисунком. Камера, встановлена над ванною з травильним розчином, дозволяє спостерігати за процесом в реальному часі. Програмне забезпечення аналізує зображення і може

виявляти нерівномірність травлення, зміни кольору розчину або інші відхилення. Це дозволяє оперативно реагувати на потенційні проблеми та коригувати параметри процесу.

Наступним етапом є свердління отворів, для контролю точності розташування та діаметру яких можна використовувати систему оптичного позиціонування. Камера, встановлена на свердлильному верстаті, захоплює зображення поверхні друкованої плати. Програмне забезпечення аналізує отримані дані і визначає чи правильні координати отворів. Це дозволяє автоматично коригувати положення свердла та забезпечувати точність свердління.

Невід'ємною частиною процесу є нанесення захисного покриття (паяльної маски), контролюючи якість нанесення камера захоплює зображення поверхні друкованої плати після нанесення маски. Програмне забезпечення аналізує зображення на наявність дефектів, таких як нерівномірність покриття, пухирці або відшарування. Такий підхід дозволяє швидко виявляти проблемні ділянки та забезпечувати якість захисного покриття.

Після чого відбувається процес металізації отворів, під час чого можуть виникнути дефекти, такі як відсутність металізації або її нерівномірність. Для контролю якості металізації отворів можна використовувати оптичний метод. Камера захоплює зображення поперечного перерізу отворів, в той час як програмне забезпечення аналізує отримані дані і визначає товщину та рівномірність металізації.

Необхідним також є етап нанесення контактних площадок для монтажу компонентів. Даний етап може здійснюватися за допомогою трафаретного друку або гальванічного осадження. Контроль цього етапу дозволяє виявляти відхилення від заданих параметрів та забезпечувати якість контактних площадок.

Останнім етапом є маркування та остаточний контроль, автоматизувати який можна використовуючи систему комп'ютерного зору. Камера захоплює

зображення готової друкованої плати з різних ракурсів. Програмне забезпечення аналізує отримані дані, перевіряючи наявність дефектів, таких як механічні пошкодження, забруднення або неправильне маркування.

Кожен з цих етапів має свої особливості та вимагає ретельного контролю якості. Для успішної ідентифікації виробів та відстеження технологічного процесу необхідно зібрати та проаналізувати дані на кожному з цих етапів.

У рамках даної кваліфікаційної роботи основна увага буде зосереджена на етапі перевірки товщини та розмірів заготовки. Цей етап є критично важливим, оскільки він визначає якість та точність кінцевого продукту.

Для збору даних на цьому етапі буде використовуватися візуальний метод. За допомогою спеціального обладнання, такого як камери, будуть отримані зображення заготовок друкованих плат. Ці зображення будуть проаналізовані для визначення розмірів заготовок.

Процес збору даних буде автоматизований, що дозволить швидко та ефективно отримувати необхідну інформацію. Аналіз зібраних даних буде проводитися за допомогою статистичних методів та алгоритмів машинного навчання.

Результати можуть бути використані для оптимізації технологічного процесу та підвищення якості кінцевого продукту. На основі отриманих даних також можуть бути розроблені рекомендації щодо вдосконалення процесу перевірки товщини та розмірів заготовок, а також запропоновані заходи для запобігання виникненню дефектів.

Отже, збір та аналіз даних на етапі перевірки розмірів заготовок є ключовим аспектом розробки програмного забезпечення для ідентифікації виробів на контрольних операціях технологічних процесів. Використання візуального методу збору даних та застосування сучасних технологій дозволить ефективно відстежувати та контролювати якість друкованих плат на цьому критичному етапі виробництва.

2.5 Розробка алгоритму ідентифікації виробів на основі нейронної мережі MobileNetV2

В сучасному світі, де технології стрімко розвиваються, периферійні обчислення набувають все більшої популярності. Завдяки зменшенню розмірів мікроконтролерів та зниженню їх вартості, стало можливим використовувати системи виявлення об'єктів у різноманітних сферах, таких як промисловість, робототехніка, системи безпеки тощо.

Периферійні обчислення – це модель, яка передбачає обробку даних безпосередньо на місці їх збору, тобто на "периферії" мережі. Це дозволяє зменшити навантаження на центральні обчислювальні потужності та прискорити процес обробки інформації. Використовуючи недорогі камери та мікроконтролери, можна створити систему, здатну розпізнавати певні об'єкти та реагувати на їх присутність у режимі реального часу [20].

У контексті розробки програмного забезпечення для ідентифікації виробів на контрольних операціях технологічних процесів, особливо актуальним є застосування технологій виявлення та класифікації об'єктів. Ці технології дозволяють автоматизувати процес контролю якості продукції, зменшити кількість помилок та підвищити ефективність виробництва.

Виявлення об'єктів – це галузь комп'ютерного зору, яка зосереджується на ідентифікації та визначенні місцезнаходження об'єктів на зображеннях або відеокадрах. Цей процес включає в себе два основні компоненти: класифікацію об'єктів та локалізацію об'єктів.

Класифікація об'єктів – це здатність комп'ютерної програми розпізнавати та категоризувати різні типи об'єктів на зображенні. Для цього програма "навчається" на великій кількості розмічених зображень, кожне з яких ідентифікується як певний об'єкт (наприклад, виріб, деталь, дефект тощо). Після навчання програма може аналізувати нові зображення та правильно класифікувати об'єкти на них.

Так як в рамках кваліфікаційної роботи збір та аналіз даних відбувається на етапі перевірки розмірів заготовок, звернено увагу на те, що різна основа друкованих плат має різний зовнішній вигляд. Зокрема, наприклад, виділено, що:

- склотекстоліт (FR4) зазвичай має зелений колір завдяки захисному лаку (солдермасці), який наноситься на плату. Сам матеріал без покриття може бути світло-жовтим або коричневим. Поверхня склотекстоліту гладка та рівномірна;

- композитні матеріали мають колір і текстуру, що можуть значно варіюватися залежно від складу композиту. Вони можуть бути коричневими, чорними, білими або навіть інших кольорів. Поверхня може бути менш рівною і гладкою, часто з видимими волокнами або текстурою композитного матеріалу.

Локалізація об'єктів – це процес визначення точного місцезнаходження цільового об'єкта на зображенні. Зазвичай, системи локалізації об'єктів малюють "обмежувальну рамку" навколо знайденого об'єкта та можуть розпізнавати декілька об'єктів одночасно.

Для реалізації алгоритму ідентифікації виробів на контрольних операціях технологічних процесів, використовуватиметься камера ESP32-CAM для збору візуальних даних. Ця камера є недорогим та ефективним рішенням для захоплення зображень у реальному часі.

Загальний алгоритм ідентифікації виробів наведено на рисунку 2.3.

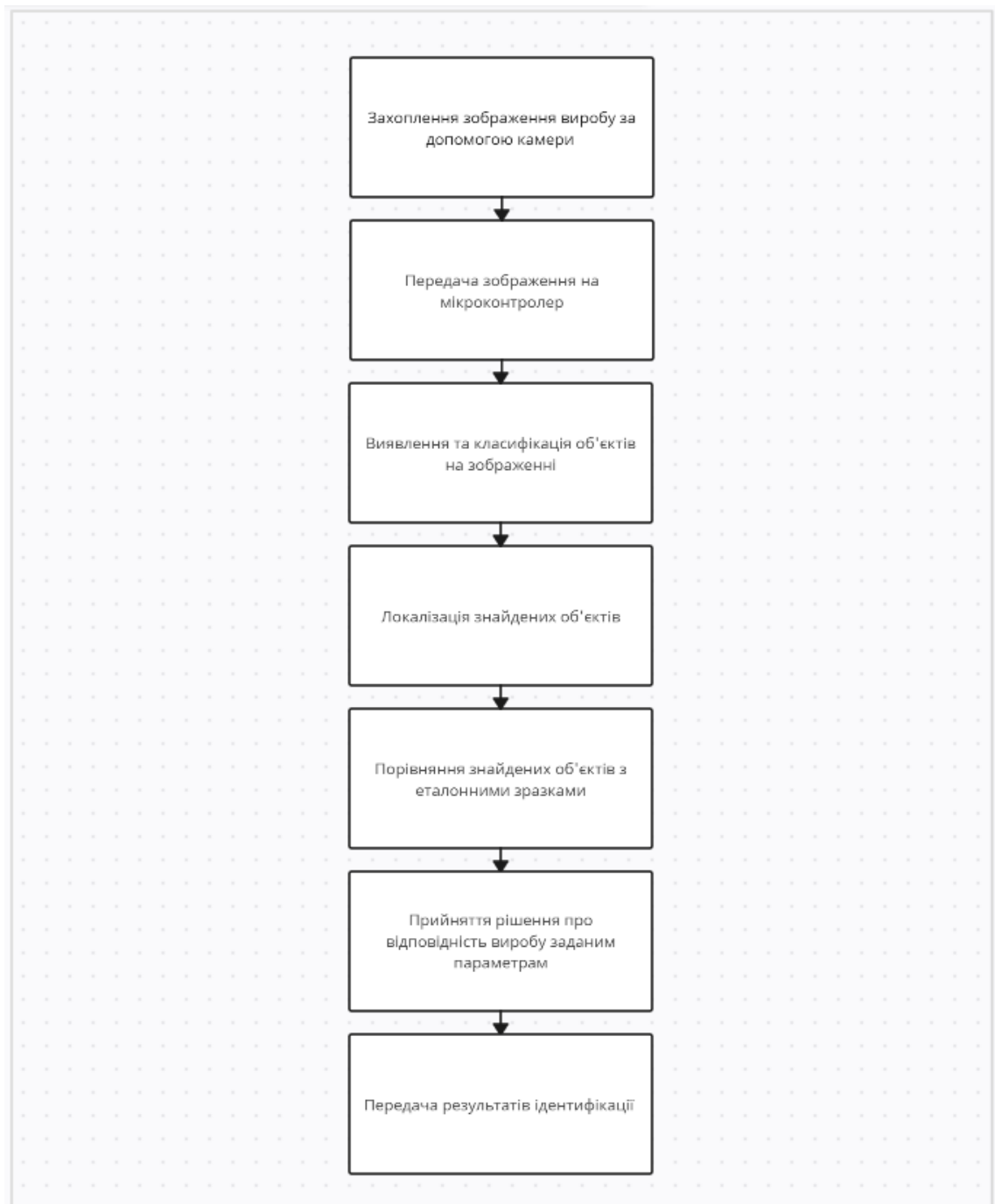


Рисунок 2.3 – Загальний алгоритм ідентифікації виробів

Для реалізації алгоритму класифікації та локалізації об'єктів у даній роботі використовується архітектура згорткової нейронної мережі MobileNetV2.

MobileNetV2 – це згорткова нейронна мережа (CNN), оптимізована для мобільних та вбудованих пристроїв [21]. Вона відрізняється високою

точністю при відносно низьких обчислювальних затратах, що робить її ідеальним вибором для використання з ESP32-CAM.

MobileNetV2 базується на архітектурі згорткових нейронних мереж (CNN) і використовує концепцію глибинно-роздільних згорток (Depthwise Separable Convolutions, DSC). Ця архітектура дозволяє значно зменшити кількість параметрів і обчислювальну складність мережі порівняно з традиційними згортковими мережами, зберігаючи при цьому високу точність розпізнавання.

Архітектура MobileNetV2 базується на наступних ключових елементах:

- Depthwise Separable Convolutions – цей тип згортки розділяє стандартну згортку на дві операції: глибинну згортку (depthwise convolution) та точкову згортку (pointwise convolution). Глибинна згортка застосовує окремий фільтр до кожного каналу вхідного тензора, тоді як точкова згортка використовує 1×1 згортку для комбінування виходів глибинної згортки. Такий підхід значно зменшує кількість обчислень та параметрів моделі без суттєвої втрати точності;

- використання інвертованих залишкових блоків (inverted residual blocks), на відміну від стандартних залишкових блоків, в яких спочатку відбувається зменшення розмірності, а потім її відновлення, в інвертованих блоках відбувається навпаки. Спочатку відбувається розширення розмірності за допомогою 1×1 згортки, потім застосовується глибинна згортка, а потім знову зменшується розмірність за допомогою 1×1 згортки. Така структура дозволяє зберегти інформацію та підвищити точність моделі;

- інвертовані залишкові блоки використовують лінійні вузькі місця (linear bottlenecks). Це означає, що остання 1×1 згортка в блоці не має нелінійної функції активації (наприклад, ReLU [22]). Такий підхід запобігає втраті інформації, яка може виникнути при застосуванні нелінійності до тензорів з малою розмірністю.

У даному випадку тензор – багатовимірний масив даних. Звичайне зображення можна представити як тензор з трьома вимірами: висота, ширина

та колірні канали (RGB). У випадку чорно-білого зображення тензор буде мати два виміри. Тензори використовуються для представлення даних в нейронних мережах, оскільки вони дозволяють ефективно обробляти великі обсяги інформації.

Згортка – це математична операція, яка застосовується до тензорів. В контексті Computer Vision згортка використовується для виявлення патернів на зображеннях. Якщо умовно взяти «фільтр», який рухається по зображенню, він виконує згортку, множачи значення пікселів зображення на відповідні значення у вікні. Результат множення сумується, формуючи один піксель на вихідному зображенні. Рухаючи «фільтр» по всьому зображенню, отримується нове зображення (feature map), на якому виділені певні патерни.

Глибинна згортка (Depthwise Convolution) застосовує окремий фільтр до кожного каналу вхідного тензора. Наприклад, якщо вхідне зображення має 3 канали (RGB), то глибинна згортка застосує 3 окремих фільтри – один для червоного каналу, один для зеленого та один для синього. Перевага глибинної згортки полягає в тому, що вона дозволяє виявляти патерни, специфічні для кожного каналу зображення.

Точкова згортка (Pointwise Convolution) використовує фільтр розміром 1×1 . Вона застосовується до результатів глибинної згортки, щоб скомбінувати інформацію з різних каналів. Точкова згортка дозволяє зменшити кількість каналів та зменшити обчислювальні затрати.

Однією з ключових переваг MobileNetV2 є її ефективність з точки зору обчислювальної складності та використання пам'яті. Завдяки використанню глибинно-роздільних згорток і bottleneck шарів, MobileNetV2 досягає високої точності розпізнавання при значно меншій кількості параметрів і операцій порівняно з іншими архітектурами CNN. Це робить її ідеальною для застосування на пристроях з обмеженими ресурсами, таких як мобільні телефони, вбудовані системи або платформи з низьким енергоспоживанням.

Окрім ефективності, MobileNetV2 також демонструє високу точність розпізнавання на різних наборах даних і задачах комп'ютерного зору. Вона

успішно застосовується для класифікації зображень, детекції об'єктів, семантичної сегментації та інших задач.

У контексті даної кваліфікаційної роботи, MobileNetV2 буде використовуватися для ідентифікації виробів на контрольних операціях технологічних процесів. Зображення виробів будуть отримуватися за допомогою камери ESP32-CAM, яка забезпечує візуальний метод збору даних.

Процес роботи MobileNetV2 наведено на рисунку 2.4.

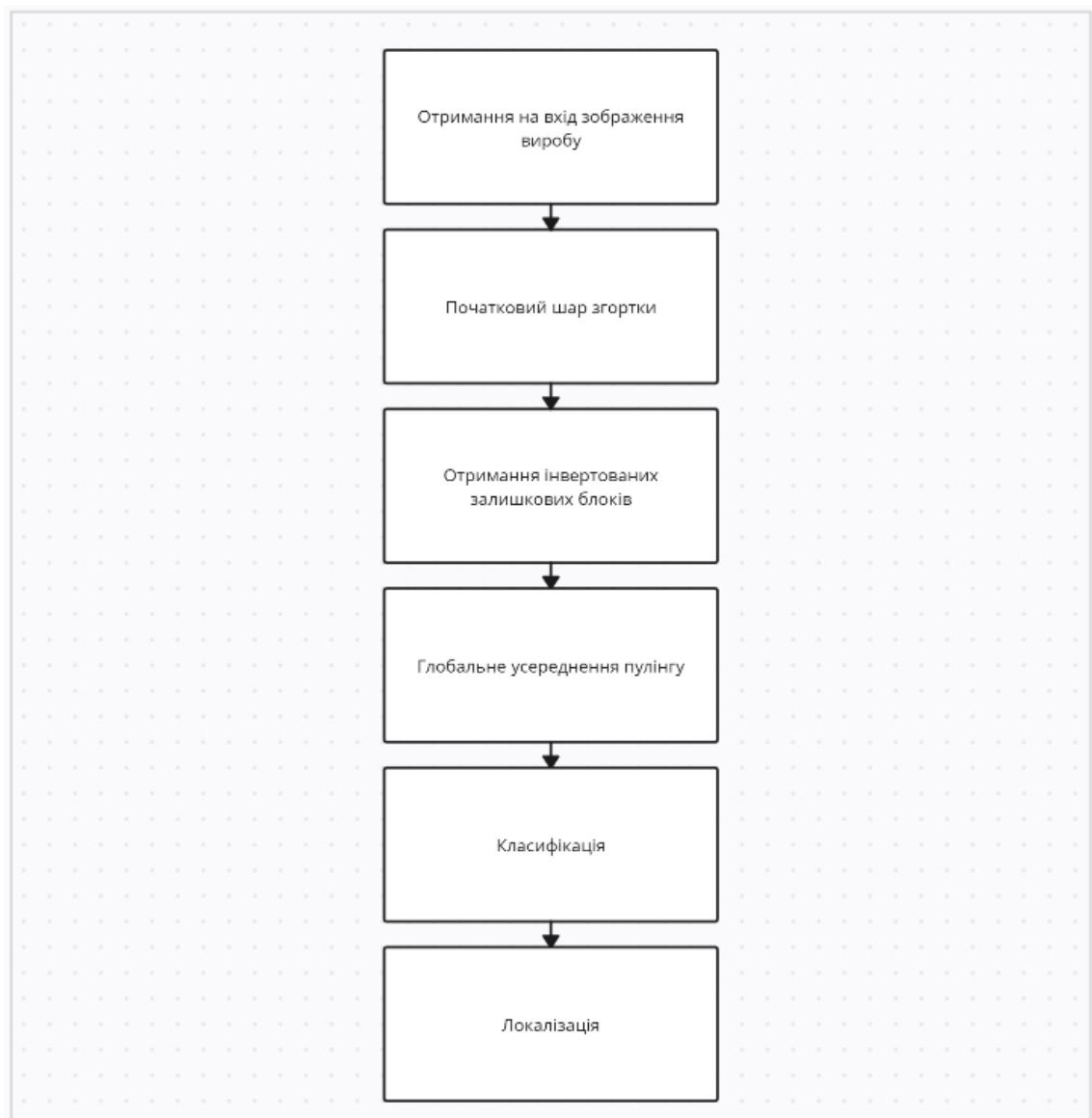


Рисунок 2.4 – Алгоритм роботи MobileNetV2

Розглянемо принцип роботи MobileNetV2.

По-перше, MobileNetV2 отримує на вхід зображення виробу, зроблене камерою ESP32- CAM. Це зображення представлено у вигляді тривимірного тензора (висота, ширина, колірні канали).

По-друге, початковий шар згортки – це стандартна згортка 3×3 з 32 фільтрами. Цей шар виявляє базові ознаки зображення, такі як краї, кути та текстури.

Після початкового шару, MobileNetV2 складається з послідовності інвертованих залишкових блоків (Inverted Residual Blocks). Кожен блок складається з трьох шарів:

- розширення (Expansion) – 1×1 згортка розширює кількість каналів, використовуючи невелику кількість фільтрів;

- глибинна згортка (Depthwise Convolution) – 3×3 глибинна згортка застосовується до кожного каналу окремо, виявляючи специфічні для каналу ознаки;

- зменшення (Projection) – 1×1 згортка зменшує кількість каналів до початкового значення, використовуючи лінійну активацію (без ReLU).

Інвертовані залишкові блоки дозволяють ефективно виявляти складні патерни, використовуючи обмежені обчислювальні ресурси. Кількість блоків та кількість каналів в кожному блоці визначаються конфігурацією MobileNetV2.

Глобальне усереднення пулінгу (Global Average Pooling) застосовується після останнього інвертованого залишкового блоку. Цей шар усереднює значення всіх пікселів в кожному каналі, формуючи вектор ознак.

Вектор ознак подається на повністю зв'язаний шар (fully connected layer), який виконує класифікацію. Цей шар визначає ймовірність приналежності зображення до кожного класу виробу.

Для локалізації виробів, MobileNetV2 може бути доповнена додатковими шарами, які визначають координати рамки навколо виробу. Ці шари можуть використовувати інформацію з попередніх шарів MobileNetV2 для точного визначення положення виробу.

Важливим аспектом розробки алгоритму є створення навчальної вибірки зображень виробів, яка буде використовуватися для тренування моделі класифікації та локалізації об'єктів. Ця вибірка повинна містити достатню кількість зразків виробів різних типів для забезпечення надійності та точності роботи алгоритму.

Таким чином, ідентифікація виробів на контрольних операціях технологічних процесів з використанням технологій комп'ютерного зору та периферійних обчислень дозволить автоматизувати процес контролю якості, підвищити ефективність виробництва та зменшити кількість помилок. Запропонований алгоритм, що базується на класифікації та локалізації об'єктів, може бути реалізований за допомогою недорогої камери ESP32-CAM та відповідного програмного забезпечення, що робить його доступним та привабливим рішенням для широкого кола промислових застосувань.

3 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ІДЕНТИФІКАЦІЙ ВИРОБІВ

3.1 Вибір апаратних модулів

Для розробки програмного забезпечення ідентифікації виробів на контрольних операціях технологічних процесів було обрано наступні апаратні модулі:

- ESP32-CAM – це потужна плата з вбудованим Wi-Fi та Bluetooth, яка ідеально підходить для проектів, пов'язаних з комп'ютерним зором;
- CMOS-камера OV2640 має роздільну здатність 2 мегапікселі та підтримує різні формати зображень та має кут огляду 66°;
- плата програмування на базі CH340G, яка є USB-UART перетворювачем.

Зовнішній вигляд перелічених компонентів наведено на рисунку 3.1.



Рисунок 3.1 – Зовнішній вигляд плати ESP32-CAM, CMOS-камери OV2640 та плати програмування [23]

ESP32-CAM – це компактний модуль, що наведений на рисунку 3.2, який об'єднує в собі функції мікроконтролера та камери.

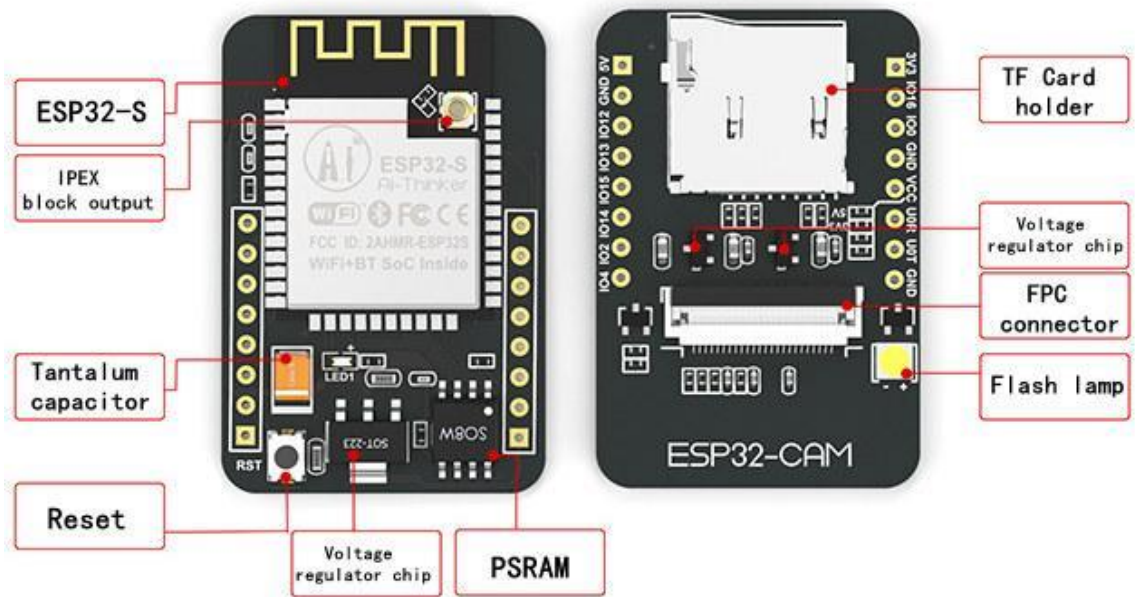


Рисунок 3.2 – Зовнішній вигляд ESP32-CAM [24]

Він забезпечує високу продуктивність при невеликому розмірі, що дозволяє використовувати його в різних додатках, включаючи системи ідентифікації та контролю. Основні характеристики ESP32-CAM представлені в таблиці 3.1.

Таблиця 3.1 – Основні характеристики ESP32-CAM

Характеристика	Значення
Процесор	Xtensa dual-core 32-bit LX 6
Частота процесора	До 240 МГц
Оперативна пам'ять	520 КБ SRAM
Флеш-пам'ять	4 МБ
Wi-Fi	IEEE 802.11 b/g/n
Bluetooth	Bluetooth 4.2 BR/EDR та BLE
Інтерфейси	UART, SPI, I2C, PWM, ADC, DAC
Вхідна напруга	5 В (через Micro USB) або 3.3 В (через пін)
Споживання енергії	До 240 мА при роботі з Wi-Fi
Габарити	27 x 40.5 мм

Також було виділено наступні переваги ESP32-CAM:

- висока продуктивність завдяки використанню двоядерного процесора з тактовою частотою до 240 МГц, що забезпечує швидку обробку зображень та виконання складних алгоритмів ідентифікації;
- підтримка Wi-Fi та Bluetooth «з коробки», що дозволяє інтегрувати його у складні системи автоматизації та віддаленого контролю без використання надлишкових модулів. Адже, як відомо, чим менше стороннього, тим краща надійність та безвідмовність;
- компактність ESP32-CAM дозволяє використовувати його у обмежених просторах та вбудовувати у різноманітні пристрої;
- багатofункціональність завдяки підтримці різноманітних інтерфейсів (UART, SPI, I2C, PWM, ADC, DAC), робить модуль універсальним для підключення додаткових сенсорів та пристроїв;

Проте виявлені наступні недоліки:

- високе енергоспоживання при роботі з Wi-Fi до 240 мА, що може вимагати додаткових заходів для забезпечення живлення в автономних системах;
- обмежена оперативна пам'ять 520 КБ SRAM, що може бути недостатньо для деяких дуже складних задач обробки зображень та зберігання великих масивів даних.

У контексті кваліфікаційної роботи ESP32-CAM виступає ключовим компонентом для реалізації візуального збору даних. З його допомогою буде здійснюватиметься фотофіксація виробів, обробка та аналіз зображення, ідентифікація та передача даних.

Камера OV2640, що наведена на рисунку 3.3, є одним із найпопулярніших сенсорів для зйомки зображень у різноманітних вбудованих системах та проектах Інтернету речей (IoT). Вона широко використовується у системах відеоспостереження, робототехніці, а також у контролі технологічних процесів, завдяки своїм відмінним характеристикам і надійності.

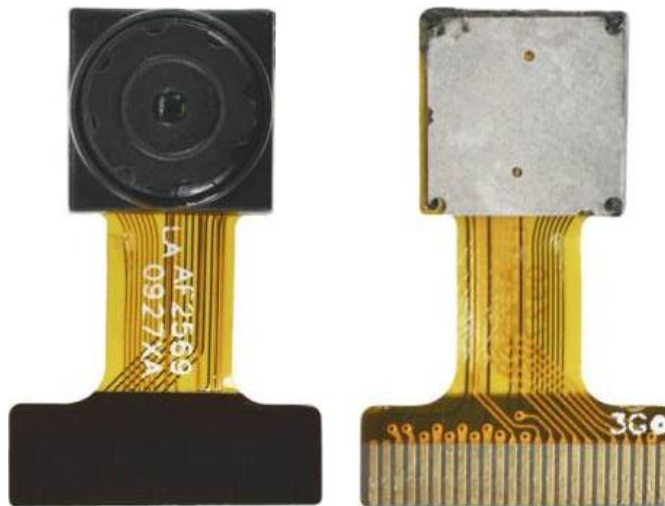


Рисунок 3.3 – Зовнішній вигляд камери OV2640 [25]

Докладні характеристики наведено у таблиці 3.2.

Таблиця 3.2 – Характеристики камери OV2640

Параметр		Значення
1		2
Розмір матриці		UXGA 1600 x 1200
Живлення сенсора		2,5 В ~ 3,0 В
Живлення логіки		1,7 В ~ 3,0 В
Споживана потужність	Активний режим	125 мВт (при 15 кадр/с, UXGA YUV режим) 140 мВт (при 15 кадр/с, UXGA стиснений режим)
	Режим очікування	600 мкА
Робочий діапазон температур		-30° С до 70° С
Діапазон температур для стабільних зображень		0° С до 50° С
Формати вихідних даних (8-біт)		YUV(422/420)/YCbCr422 RGB565/555 8-біт стиснені дані 8-/10-біт необроблені RGB дані
Розмір об'єктива		1/4"
Максимальна швидкість передачі зображень	UXGA/SXGA	15 кадр/с
	SVGA	30 кадр/с
	CIF	60 кадр/с
Відношення сигнал/шум		40 дБ

Продовження таблиці 3.2

1	2
Динамічний діапазон	50 дБ
Корекція гами	Програмована
Розмір пікселя	2,2 мкм x 2,2 мкм
Площа зображення	3590 мкм x 2684 мкм

Переваги камери OV2640:

– висока якість зображення завдяки високій роздільній здатності. Це дозволяє отримувати чіткі та деталізовані зображення, що є критично важливим для ідентифікації дефектів або невідповідностей у виробках на різних етапах технологічного процесу;

– компактні розміри, що дозволяє легко інтегрувати її у різні вбудовані системи та пристрої;

– висока чутливість сенсора, що надає змогу працювати у широкому діапазоні освітленості, а це, в свою чергу, забезпечує надійну роботу навіть в умовах недостатнього освітлення;

– енергоефективність, що дозволяє використовувати її у системах без необхідності великих джерел живлення;

– широкий діапазон налаштувань, включаючи експозицію, баланс білого, насиченість та контрастність, що дозволяє оптимізувати зображення під конкретні умови зйомки.

Проте, як і у будь-якого пристрою, камера OV2640 має свої недоліки:

– обмежена роздільна здатність. Хоча камера забезпечує високу якість зображення, її роздільна здатність може бути недостатньою для деяких високоточних завдань, де потрібна дуже висока деталізація;

– відсутність вбудованої стабілізації зображення, що може бути проблемою в умовах вібрацій або руху, часто присутніх на виробничих лініях;

– обмежений динамічний діапазон, що може призводити до втрати деталей у дуже світлих або дуже темних областях зображення. Це може ускладнити ідентифікацію дефектів у таких умовах;

– чутливість до умов навколишнього середовища, таких як температура і вологість, що може вплинути на стабільність роботи камери та якість зображень.

Таким чином, камера OV2640 виступає ключовим елементом, який забезпечує візуальний збір даних для подальшої обробки та аналізу.

Також плата програмування на базі чипа CH340G, яка наведена на рисунку 3.4, є невід'ємною частиною процесу розробки та відлагодження програмного забезпечення для ESP32. Ця плата забезпечує зв'язок між комп'ютером і мікроконтролером ESP32 через USB-інтерфейс, що дозволяє зручно завантажувати прошивки та здійснювати налагодження в реальному часі.

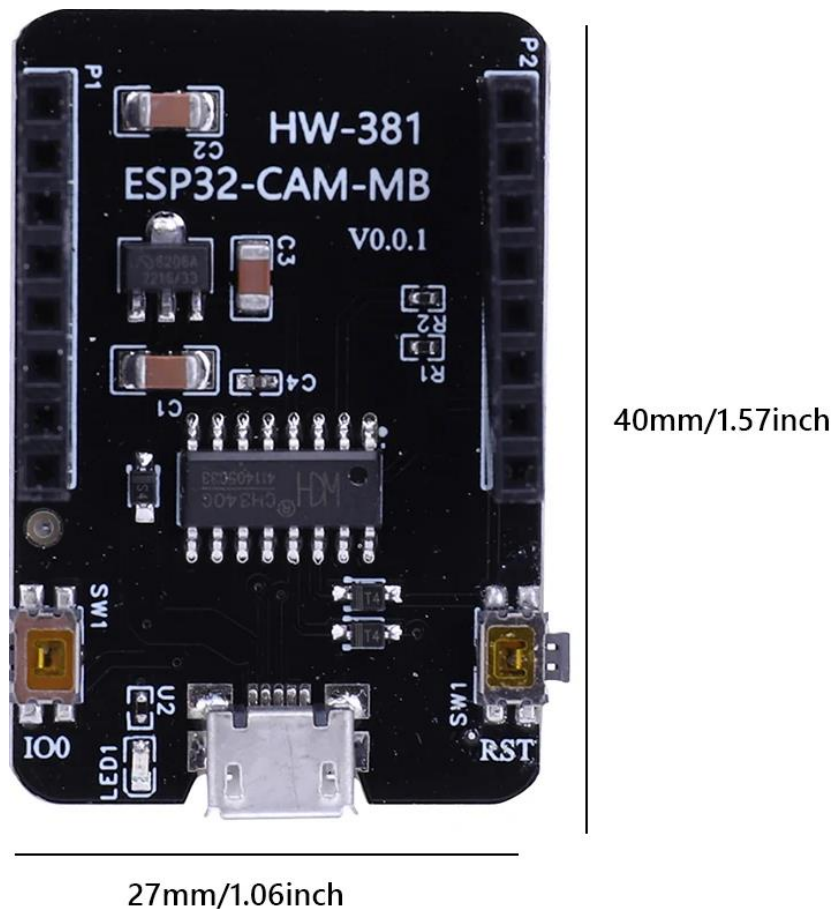


Рисунок 3.4 – Зовнішній вигляд плати програмування [26]

Переваги використання плати програмування на базі CH340G:

- простота використання, бо плата має простий інтерфейс та легко підключається до комп'ютера через USB;
- чип CH340G забезпечує стабільне та надійне з'єднання між комп'ютером і ESP32, що є важливим для коректного завантаження прошивок і передачі даних;
- підтримка високих швидкостей передачі даних, що дозволяє швидко передавати великі обсяги даних;
- сумісність з різними ОС, включаючи Windows, macOS та Linux, що робить її універсальною.

Попри перелічені переваги, дана плата має наступні недоліки:

- плата програмування на базі CH340G призначена виключно для програмування та налагодження і не має додаткових функцій, які могли б бути корисними в інших аспектах розробки;
- іноді можуть виникати проблеми зі сумісністю драйверів;
- плата не забезпечує додаткового живлення для ESP32, що може вимагати окремого джерела живлення для стабільної роботи мікроконтролера під час інтенсивного використання.

Отже, використання даної плати забезпечує зручний інтерфейс для завантаження прошивок на ESP32 та дозволяє проводити налагодження програмного забезпечення в реальному часі, що є критично важливим для створення надійної та ефективної системи ідентифікації виробів.

Взаємодія обраних апаратних модулів відбувається наступним чином:

- а) збір даних відбувається за допомогою камери OV2640, що фіксує зображення виробу на контрольній операції;
- б) зображення передається з OV2640 на ESP32-CAM через інтерфейс SCCB;
- в) обробка отриманого зображення з використанням алгоритмів машинного зору для ідентифікації виробу;

г) передача результатів ідентифікації через послідовний порт на комп'ютер або сервер для подальшої обробки або відображення;

д) плата програмування на базі CH340G використовується для завантаження програмного забезпечення на ESP32-CAM, налаштування параметрів камери та алгоритмів обробки зображень.

Розроблюване програмне забезпечення, використовуючи можливості ESP32-CAM, дозволить автоматизувати процес контролю якості, зменшити час на перевірку виробів та підвищити точність ідентифікації. Завдяки високій роздільній здатності камери та можливості бездротової передачі даних, система на базі ESP32-CAM буде ефективною та надійною у виконанні завдань контролю та ідентифікації.

Таким чином, ESP32-CAM є оптимальним вибором для реалізації проекту, завдяки своїм технічним характеристикам, функціональності та можливостям інтеграції у різноманітні системи автоматизації та контролю.

3.2 Вибір середовища розробки програмного забезпечення ідентифікації

Для програмування мікроконтролера ESP32, який буде використовуватися в розробленому програмному забезпеченні для ідентифікації виробів на контрольних операціях технологічних процесів, можна використовувати одну з декількох можливих мов програмування.

C та C++ є одними з найпопулярніших мов програмування для вбудованих систем та мікроконтролерів. Ці мови забезпечують низькорівневий контроль над пам'яттю та апаратними ресурсами, що є критично важливим для ефективної роботи на обмежених ресурсах мікроконтролера. C/C++ є потужними та гнучкими мовами, які дозволяють створювати ефективні та оптимізовані додатки для вбудованих систем.

Переваги виділені наступні:

– висока продуктивність та ефективність використання ресурсів;

- великий набір бібліотек, включаючи бібліотеки для роботи з камерою ESP32-CAM та машинного навчання;

- підтримка безлічі периферійних пристроїв та протоколів (Wi-Fi, Bluetooth, SPI, I2C).

Недоліки:

- вимагає більш глибокого розуміння мови та архітектури ESP32;

- відсутність вбудованого garbage collector.

Python є високорівневою мовою програмування, яка набула популярності завдяки своїй простоті та зрозумілості. Підтримується офіційно для ESP32 за допомогою спеціальної прошивки MicroPython, яка дозволяє виконувати Python-скрипти безпосередньо на мікроконтролері.

Python може бути зручним для прототипування та швидкої розробки, але його продуктивність може бути нижчою, ніж у C/C++. Також використання Python на ESP32 може бути обмежене наявною пам'яттю та обчислювальними ресурсами мікроконтролера.

Проте виділено наступні переваги:

- простота та читабельність коду, що дозволяє швидко розробляти прототипи;

- підтримка великої кількості бібліотек Python;

- гнучка інтеграція з периферією та протоколами через вбудовані модулі;

- MicroPython дозволяє виконувати код у реальному часі, що спрощує налагодження та тестування.

Недоліки:

- нижча продуктивність порівняно з C/C++;

- обмежений доступ до деяких функцій низького рівня;

- обмежена підтримка бібліотек, хоча кількість бібліотек для MicroPython зростає, вона все ще менша, ніж для C/C++.

JavaScript – популярна мова програмування, що широко використовується у веб-розробці. На платформі ESP32 JavaScript реалізується за допомогою фреймворку Espruino.

Espruino – це інтерпретатор JavaScript, оптимізований для мікроконтролерів, включаючи ESP32. Він дозволяє запускати JavaScript-код безпосередньо на мікроконтролері, забезпечуючи можливість швидкого та ефективного розроблення прототипів і кінцевих продуктів.

Переваги:

- гнучкість та простота JavaScript;
- асинхронне програмування, так як JavaScript спроектований для асинхронних завдань, що може бути корисним при обробці даних з датчиків;
- веб-орієнтована розробка, що передбачає легку інтеграцію з веб-технологіями, що може бути корисним для створення інтерфейсів керування.

Недоліки:

- менший набір бібліотек порівняно з іншими мовами;
- обмежена підтримка деяких специфічних функцій ESP32;
- відносно низька продуктивність JavaScript, як і інших інтерпретованих мов, у порівнянні з компільованими мовами;
- JavaScript менш розповсюджений у світі мікроконтролерів, що обмежує кількість доступних ресурсів і підтримки.

Go (TinyGo) – це реалізація мови Go, оптимізована для мікроконтролерів. Підтримка ESP32 реалізована через фреймворк TinyGo.

Використання Go може мати наступні переваги:

- типізована мова програмування з високим рівнем безпеки;
- простота та читабельність коду.

Недоліки:

- обмежена кількість бібліотек;
- порівняно низька продуктивність.

Після ретельного аналізу та врахування вимог до програмного забезпечення для ідентифікації виробів, найбільш доцільним вибором для

розробки є мова програмування C++. C++ забезпечує необхідний рівень контролю над апаратними ресурсами, продуктивністю та пам'яттю, що є критично важливим для ефективної роботи на обмеженій платформі ESP32. Крім того, C++ має потужну екосистему бібліотек та інструментів для розробки вбудованих систем, що значно полегшить процес розробки.

Далі для успішної розробки програмного забезпечення для ідентифікації виробів на контрольних операціях технологічних процесів важливо вибрати оптимальне середовище розробки. У даному випадку, оскільки буде використовуватися мікроконтролер ESP32-CAM для збору даних, розглянуто кілька середовищ програмування, які підтримують ESP32, і порівняно їх. Середовище має забезпечувати зручний інструментарій для написання, компіляції, завантаження та налагодження програмного коду для ESP32-CAM.

Arduino IDE – одне з найпопулярніших середовищ розробки для мікроконтролерів, включаючи ESP32. Воно підтримує мови програмування C та C++ і має простий інтерфейс, приклад якого наведено на рисунку 3.5.

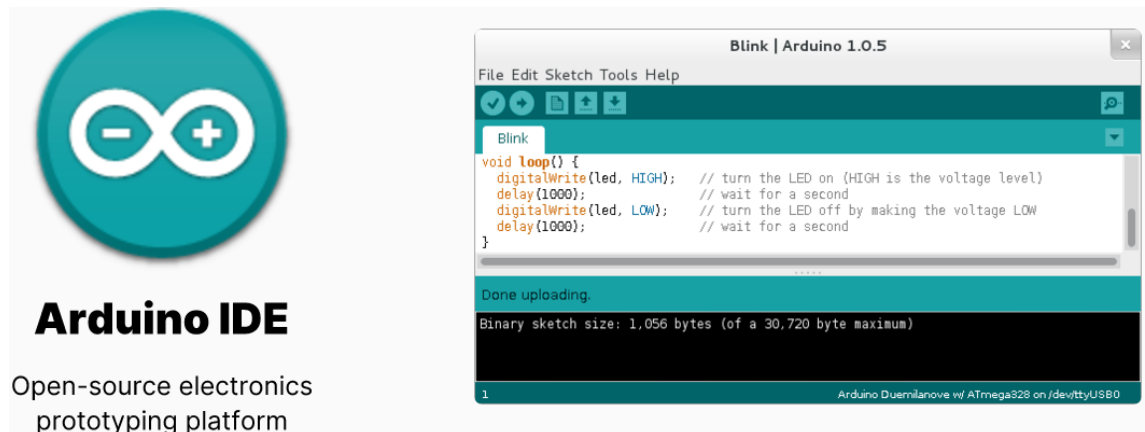


Рисунок 3.5 – Arduino IDE [27]

Використання Arduino IDE надає наступні переваги:

- простота використання та інтуїтивно зрозумілий інтерфейс;
- велика кількість готових бібліотек для роботи з різними модулями та сенсорами;

– активна спільнота користувачів і розробників, що забезпечує швидке вирішення проблем та підтримку;

– вбудована підтримка ESP32 через менеджер плат.

Проте виявлені наступні недоліки:

– обмежені можливості налагодження, що може бути недостатньо для складних проектів;

– відсутність складних інструментів для управління великими проектами та залежностями.

Наступне середовище розробки PlatformIO. Це сучасна середовище розробки для вбудованих систем, яка підтримує велику кількість платформ, включаючи ESP32. Скріншот інтерфейсу PlatformIO у MVSC наведено на рисунку 3.6.

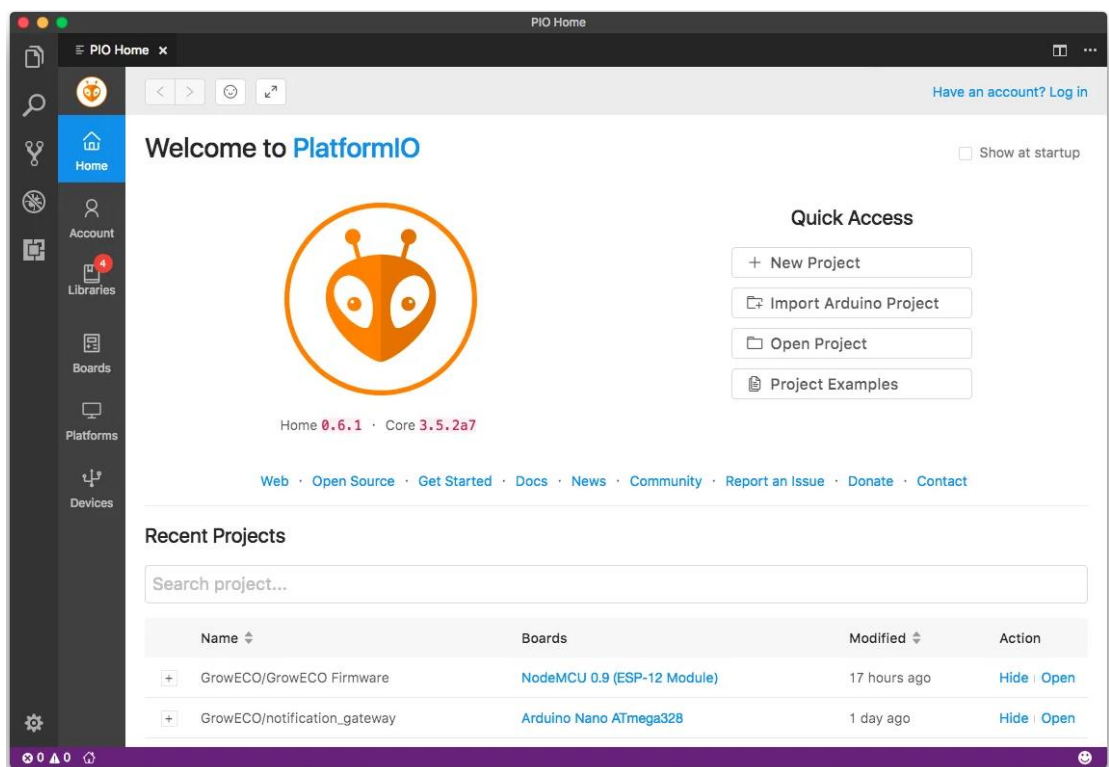


Рисунок 3.6 – PlatformIO [28]

Переваги PlatformIO полягають у:

– підтримці багатьох платформ, що надає можливість працювати з різними мікроконтролерами та платами в одному середовищі;

- інтеграції з популярними IDE, такими як Visual Studio Code, Atom, що забезпечує потужні можливості редагування коду та налагодження;
- наявності зручних інструментів для управління бібліотеками та залежностями;
- покращених можливості налагодження, що забезпечує детальне відстеження помилок;
- розширених можливостях, такі як автодоповнення коду, інтеграція з системами контролю версій, віддалене налагодження та управління бібліотеками.

Недоліки PlatformIO:

- складніша конфігурація, більше налаштувань та опцій, що може бути складнішим та незручним для великих проектів;
- більш складний інтерфейс у порівнянні з Arduino IDE, що може потребувати більше часу на виконання тих же самих операцій.

ESP-IDF – офіційне середовище розробки від компанії Espressif для мікроконтролерів ESP32, що підтримує мови програмування C та C++. Він надає низькорівневий доступ до функцій та периферії ESP32, що дозволяє розробникам створювати високопродуктивні та оптимізовані програми.

На рисунку 3.7 наведено використання одного з шаблонів коду для блимання світлодіодом ESP32.

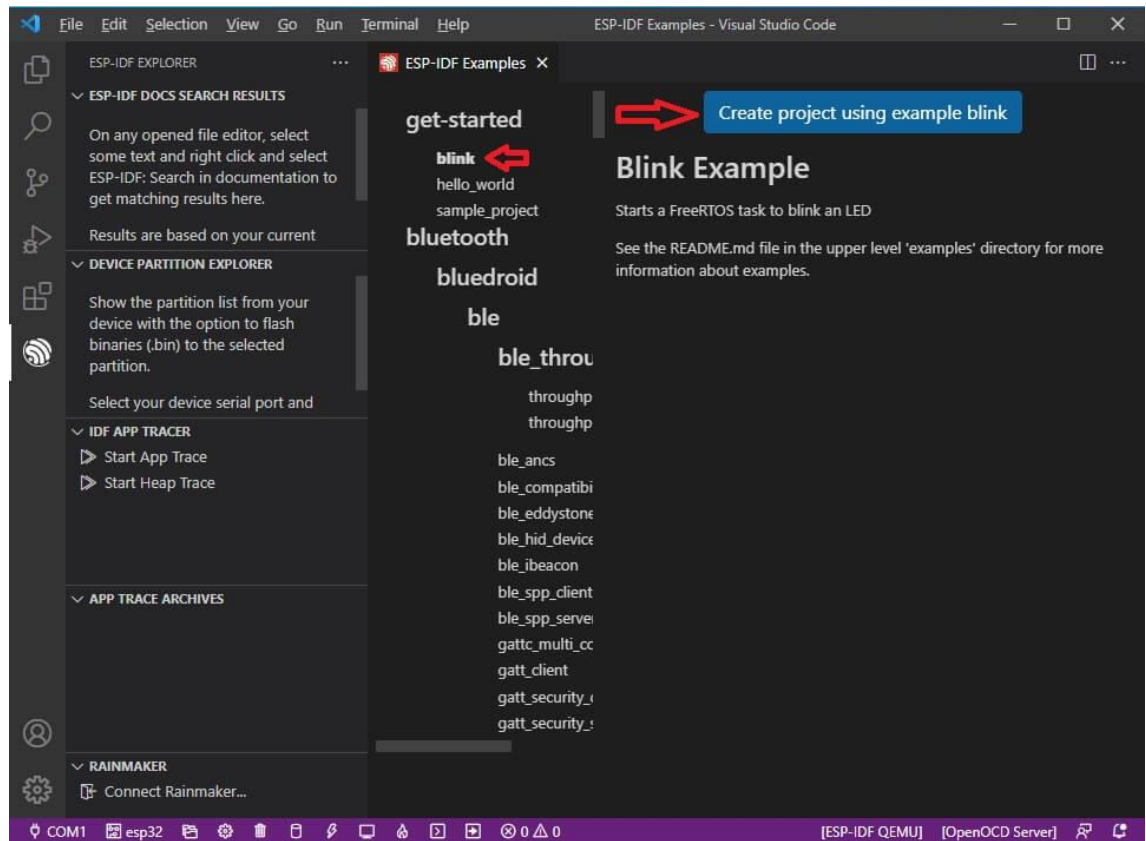


Рисунок 3.7 – ESP-IDF [29]

Переваги використання ESP-IDF полягають у:

- повному контролю над апаратним забезпеченням, доступу до всіх функцій і можливостей ESP32;
- потужних інструментах розробки включаючи інструменти для налаштування, компіляції та налагодження;
- повній та чіткій документації і великій кількості прикладів коду;
- вбудованій підтримці операційної системи реального часу FreeRTOS, що дозволяє створювати багатозадачні програми.

Також були виявлені наступні недоліки:

- складність початкового налаштування та використання;
- інтенсивне використання ресурсів системи для налаштування та використання.

MicroPython – це інтерпретатор мови програмування Python, оптимізований для мікроконтролерів, включаючи ESP32. Існують різні IDE, що підтримують розробку на MicroPython, такі як Thonny, uPyCraft та інші.

На рисунку 3.8 наведено використання MicroPython у поєднанні з IDE PyCharm.

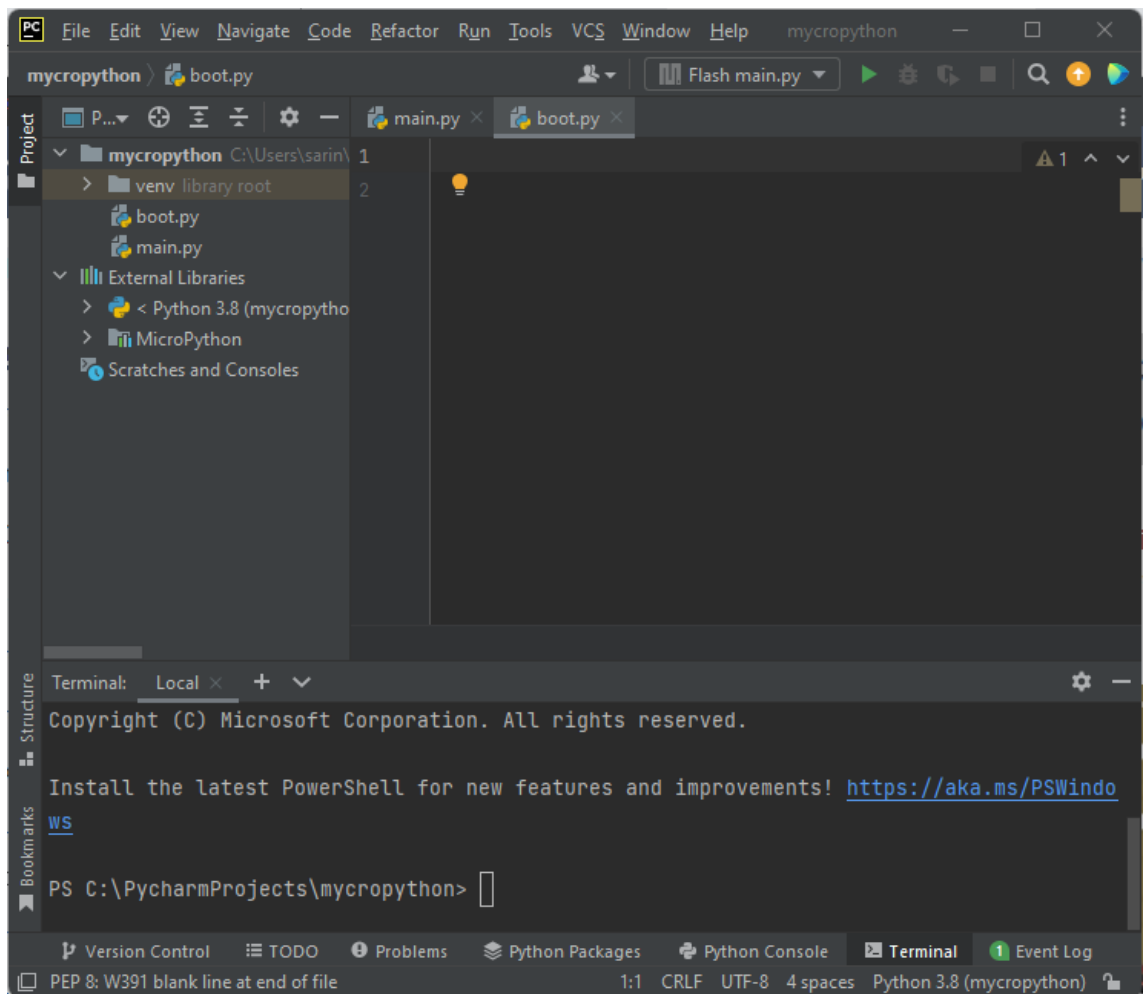


Рисунок 3.8 – MicroPython з PyCharm [30]

Переваги даного варіанту:

- простота синтаксису Python;
- інтерактивне програмування, яке надає можливість виконання коду в реальному часі, що спрощує налагодження та тестування;
- динамічна природа Python дозволяє швидко створювати і тестувати прототипи.

Недолік – обмежена кількість бібліотек для MicroPython, ніж для C/C++.

Espruino – це середовище розробки для мови програмування JavaScript, оптимізоване для мікроконтролерів, включаючи ESP32. Espruino Web IDE – це веб-інтерфейс, що наведений на рисунку 3.9, для розробки та налагодження коду на Espruino.

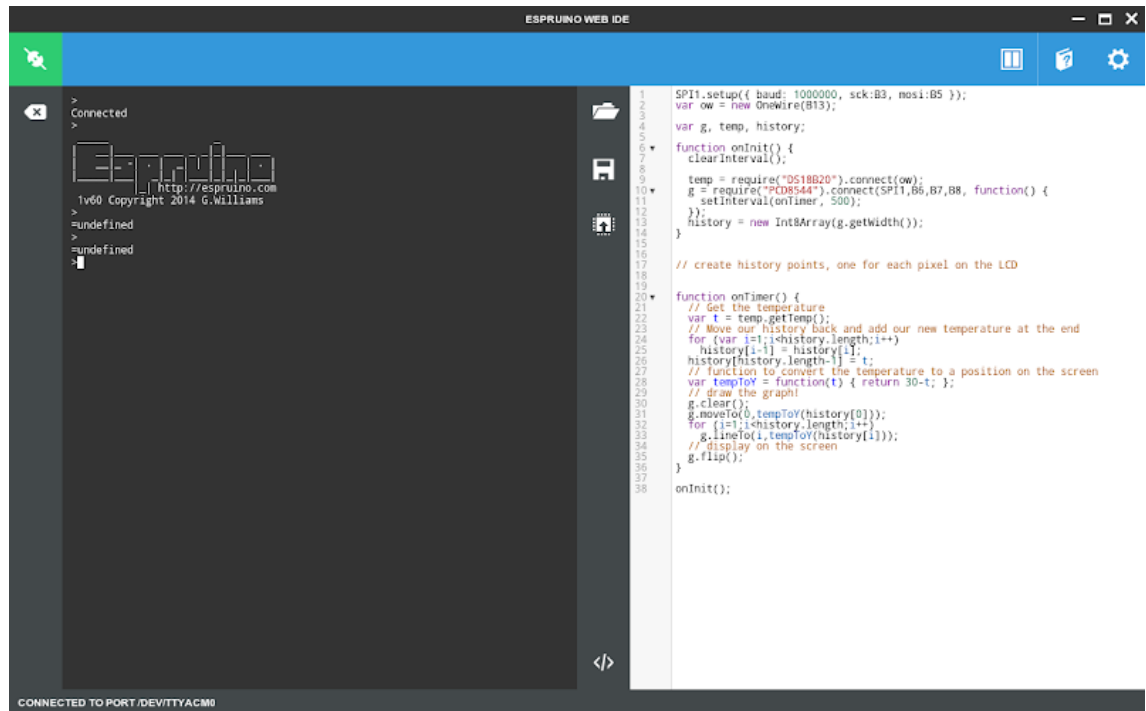


Рисунок 3.9 – Espruino Web IDE [31]

Переваги використання Espruino:

- підтримує асинхронне програмування, що дозволяє ефективно обробляти події;
- зручна інтеграція з веб-технологіями для створення інтерфейсів та комунікаційних систем.

Недолік – обмежений доступ до апаратного забезпечення, деякі функції мікроконтролера можуть бути менш доступними або складнішими у використанні.

Таким чином після ретельного аналізу можливих середовищ розробки для ESP32 було прийнято рішення використовувати Arduino IDE для

розробки програмного забезпечення ідентифікації виробів на основі ESP32-CAM. Цей вибір обумовлений наступними факторами:

- широкий вибір бібліотек для роботи з ESP32-CAM та іншими компонентами;
- велика спільнота розробників, що забезпечує швидке вирішення проблем та підтримку;
- просте налаштування та використання платформи ESP32 завдяки вбудованій підтримці.

Тому для реалізації програмного забезпечення ідентифікації виробів на контрольних операціях технологічних процесів буде використано мову програмування C++ у поєднанні з середовищем розробки Arduino IDE, яке забезпечує зручну інтеграцію з платформою ESP32 та її периферійними пристроями, такими як камера ESP32-CAM.

3.3 Розробка макету програмного забезпечення

Для реалізації програмного забезпечення, яке здатне виконувати ідентифікацію виробів на контрольних операціях технологічних процесів, необхідно спочатку зібрати дані для тренування моделі машинного навчання. Початковий етап – збір даних за допомогою модуля ESP32-CAM, з використанням світлодіодного спалаху, для збереження зображень з камери.

На рисунку 3.10 наведено фрагмент коду імпорту необхідних бібліотек та оголошення об'єктів.

```

1 #define MAX_RESOLUTION_XGA 1
2 #define FLASH_LED_PIN 4
3
4 #include "esp32cam.h"
5 #include "esp32cam/http/ImageCollectionServer.h"
6
7 Cam cam;
8 Http::CollectImagesServer http(cam);

```

Рисунок 3.10 – Фрагмент коду імпорту бібліотеки та оголошення об'єктів

Бібліотеки `esp32cam.h` та `esp32cam/http/ImageCollectionServer.h` використовуються для роботи з камерою ESP32-CAM і сервером збору зображень відповідно. Об'єкти `Cam` та `CollectImagesServer` забезпечують доступ до функцій камери та управління HTTP сервером для збору зображень.

Фрагмент коду ініціалізації системи наведено на рисунку 3.11.

```

10 void setup() {
11   Serial.begin(115200);
12   delay(3000);
13   Serial.println("Init");
14   pinMode(FLASH_LED_PIN, OUTPUT);
15   digitalWrite(FLASH_LED_PIN, HIGH);
16   cam.aithinker();
17   cam.highQuality();
18   cam.highestSaturation();
19   cam.xga();

```

Рисунок 3.11 – Фрагмент коду ініціалізації системи

Функція `setup()` виконується один раз при запуску мікроконтролера. Вона налаштовує послідовний порт для відладки, встановлює пін для керування світлодіодним спалахом як вихідний і вмикає спалах. Також налаштовується камера для роботи в режимі високої якості із роздільною здатністю XGA.

Підключення до Wi-Fi мережі відбувається за допомогою фрагменту коду на рисунку 3.12.

```
21 while (!cam.begin())
22     Serial.println(cam.getErrorMessage());
23
24 while (!cam.connect("1", "09090909"))
25     Serial.println(cam.getErrorMessage());
26
27 while (!http.begin())
28     Serial.println(http.getErrorMessage());
29
30 Serial.println(http.getWelcomeMessage());
31 }
```

Рисунок 3.12 – Фрагмент коду підключення до Wi-Fi

Цей блок коду забезпечує підключення камери до Wi-Fi мережі. Він містить цикли для повторних спроб підключення у разі невдачі, з відповідним виведенням повідомлень про помилки.

Обробка HTTP запитів відбувається за допомогою метода `http.handle()`, що розміщений у функції `loop()`. Функція `loop()` виконується безперервно і відповідає за обробку HTTP запитів від клієнтів. Цей код дозволяє камері ESP32-CAM збирати зображення для подальшого використання в навчанні моделі машинного навчання, а світлодіодний спалах використовується для покращення якості зображень в умовах низького освітлення.

Цей етап є важливим для збору якісних даних, необхідних для тренування моделі машинного навчання, яка буде використовуватись для ідентифікації виробів на контрольних операціях технологічних процесів.

Таким чином було зібрано 5 архівів із загальною кількістю понад 1200 зображень, на яких зафіксовано фон та плати dell, Hуnіx, phone, Samsung, steuergerat. Приклад отриманих фото наведено на рисунку 3.13.

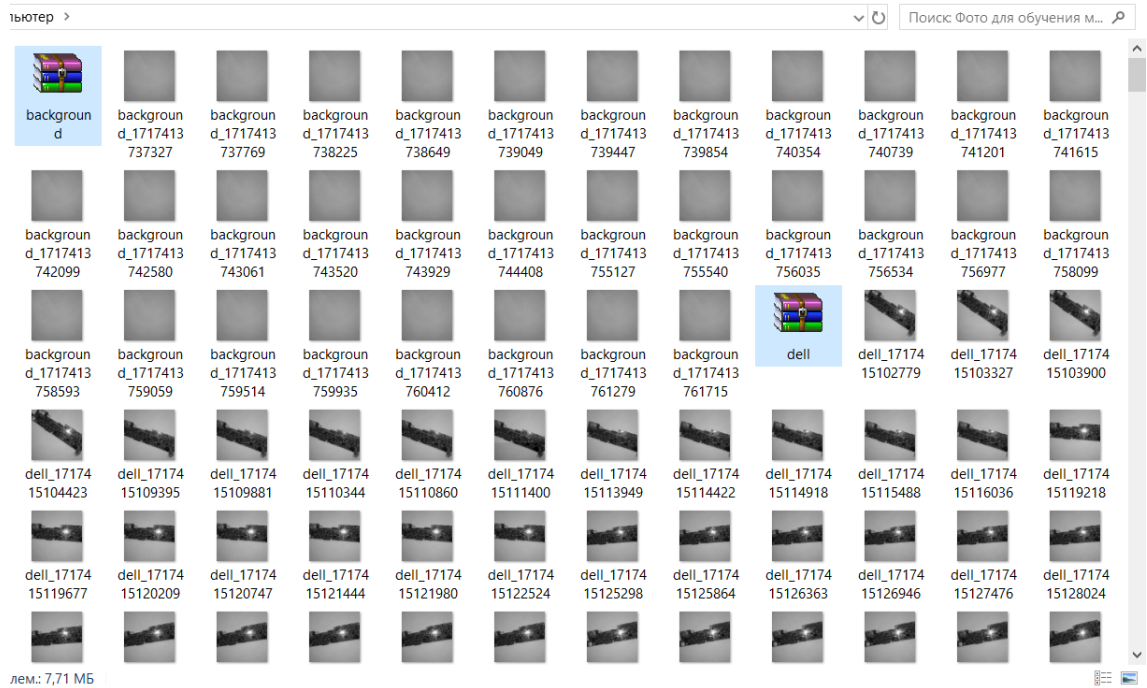


Рисунок 3.13 – Приклад отриманих зображень

Наступним кроком було створення моделі машинного навчання використовуючи TensorFlow Lite. Результатом було отримання набору даних, що наведено на рисунку 3.14.

```
tflite_learn.h
ALIGN(16) const unsigned char tflite_learn[] = {
0x20, 0x00, 0x00, 0x00, 0x00, 0x54, 0x46, 0x4c, 0x33, 0x00, 0x00, 0x00, 0x00,
0x14, 0x00, 0x20, 0x00, 0x1c, 0x00, 0x18, 0x00, 0x14, 0x00, 0x10, 0x00,
0x0c, 0x00, 0x00, 0x00, 0x08, 0x00, 0x04, 0x00, 0x14, 0x00, 0x00, 0x00,
0x1c, 0x00, 0x00, 0x00, 0x8c, 0x00, 0x00, 0x00, 0xe4, 0x00, 0x00, 0x00,
0x84, 0x2f, 0x00, 0x00, 0x94, 0x2f, 0x00, 0x00, 0x18, 0x9c, 0x00, 0x00,
0x03, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x10, 0x00, 0x00, 0x04, 0x00,
0x00, 0x00, 0x0a, 0x00, 0x10, 0x00, 0x0c, 0x00, 0x08, 0x00, 0x04, 0x00,
0x0a, 0x00, 0x00, 0x00, 0x0c, 0x00, 0x00, 0x1c, 0x00, 0x00, 0x00,
0x3c, 0x00, 0x00, 0x00, 0x0f, 0x00, 0x00, 0x00, 0x73, 0x65, 0x72, 0x76,
0x69, 0x6e, 0x67, 0x5f, 0x64, 0x65, 0x66, 0x61, 0x75, 0x6c, 0x74, 0x00,
0x01, 0x00, 0x00, 0x00, 0x04, 0x00, 0x00, 0x00, 0x9c, 0xff, 0xff, 0xff,
0x47, 0x00, 0x00, 0x00, 0x04, 0x00, 0x00, 0x00, 0x08, 0x00, 0x00, 0x00,
0x6f, 0x75, 0x74, 0x70, 0x75, 0x74, 0x5f, 0x30, 0x00, 0x00, 0x00, 0x00,
0x01, 0x00, 0x00, 0x00, 0x04, 0x00, 0x00, 0x00, 0x12, 0xd0, 0xff, 0xff,
0x04, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00, 0x78, 0x00, 0x00, 0x00,
0x02, 0x00, 0x00, 0x00, 0x34, 0x00, 0x00, 0x00, 0x04, 0x00, 0x00, 0x00,
0xdc, 0xff, 0xff, 0xff, 0x4a, 0x00, 0x00, 0x04, 0x00, 0x00, 0x00, 0x00,
0x13, 0x00, 0x00, 0x00, 0x43, 0x4f, 0x4e, 0x56, 0x45, 0x52, 0x53, 0x49,
0x4f, 0x4e, 0x5f, 0x4d, 0x45, 0x54, 0x41, 0x44, 0x41, 0x54, 0x41, 0x00,
0x08, 0x00, 0x0c, 0x00, 0x08, 0x00, 0x04, 0x00, 0x08, 0x00, 0x00, 0x00,
0x49, 0x00, 0x00, 0x00, 0x04, 0x00, 0x00, 0x13, 0x00, 0x00, 0x00,
0x6d, 0x69, 0x6e, 0x5f, 0x72, 0x75, 0x6e, 0x74, 0x69, 0x6d, 0x65, 0x5f,
0x76, 0x65, 0x72, 0x73, 0x69, 0x6f, 0x6e, 0x00, 0x4b, 0x00, 0x00, 0x00,
0x9c, 0x2e, 0x00, 0x00, 0x94, 0x2e, 0x00, 0x00, 0x64, 0x2e, 0x00, 0x00,
0x3c, 0x2e, 0x00, 0x00, 0x6c, 0x2d, 0x00, 0x00, 0xdc, 0x2c, 0x00, 0x00,
0xcc, 0x26, 0x00, 0x00, 0xfc, 0x25, 0x00, 0x00, 0x6c, 0x24, 0x00, 0x00,
0x3c, 0x24, 0x00, 0x00, 0xac, 0x22, 0x00, 0x00, 0xdc, 0x21, 0x00, 0x00,
0x1c, 0x20, 0x00, 0x00, 0x4c, 0x1f, 0x00, 0x00, 0xbc, 0x1d, 0x00, 0x00,
0x8c, 0x1d, 0x00, 0x00, 0xfc, 0x1b, 0x00, 0x00, 0x2c, 0x1b, 0x00, 0x00,
0x6c, 0x19, 0x00, 0x00, 0x9c, 0x18, 0x00, 0x00, 0x0c, 0x17, 0x00, 0x00,
0xdc, 0x16, 0x00, 0x00, 0x4c, 0x15, 0x00, 0x00, 0x7c, 0x14, 0x00, 0x00,
0xb, 0x12, 0x00, 0x00, 0xec, 0x11, 0x00, 0x00, 0x5c, 0x10, 0x00, 0x00,
0x2c, 0x10, 0x00, 0x00, 0x9c, 0x0e, 0x00, 0x00, 0xcc, 0x0d, 0x00, 0x00,
0x0c, 0x0c, 0x00, 0x00, 0x3c, 0x0b, 0x00, 0x00, 0xac, 0x09, 0x00, 0x00,
0x7c, 0x09, 0x00, 0x00, 0xec, 0x07, 0x00, 0x00, 0x1c, 0x07, 0x00, 0x00,
```

Рисунок 3.14 – Набір даних для ідентифікації

Для використання вищенаведеного набору даних використовується наступний фрагмент коду, що наведений на рисунку 3.15, в якому визначено категорії класифікації об'єктів. Тобто, в даному випадку, це або виробники плат, такі як dell, Samsung, Hynix, або тип плат, наприклад phone або steuergerat.

```
#include <stdint.h>
#include "model_metadata.h"

#include "tflite-model/tflite_learn.h"
#include "lib/classifier/esp32_model_types.h"
#include "lib/classifier/inferencing_engines/engines.h"

const char* esp32_classifier_inferencing_categories[] = { "dell", "hynix", "phone", "samsung", "steuergerat" };

esp32_dsp_named_axis_t esp32_dsp_config_4_named_axes[] = {
    { .name = "Image", .axis = 0 }
};

size_t esp32_dsp_config_4_named_axes_size = 1;
uint8_t esp32_dsp_config_4_axes[] = { 0 };
const uint32_t esp32_dsp_config_4_axes_size = 1;
esp32_dsp_config_image_t esp32_dsp_config_4 = {
    4, // uint32_t blockId
    1, // int implementationVersion
    1, // int length of axes
    esp32_dsp_config_4_named_axes, // named axes
    esp32_dsp_config_4_named_axes_size, // size of the named axes array
    "Grayscale" // select channels
};

const size_t esp32_dsp_blocks_size = 1;
esp32_model_dsp_t esp32_dsp_blocks[esp32_dsp_blocks_size] = {
    { // DSP block 4
        4,
        9216, // output size
        &extract_image_features, // DSP function pointer
        (void*)&esp32_dsp_config_4, // pointer to config struct
        esp32_dsp_config_4_axes, // array of offsets into the input stream, one for each axis
        esp32_dsp_config_4_axes_size, // number of axes
    }
};
```

Рисунок 3.15 – Використання набору даних моделей

Основний файл скетчу починається з ініціалізації та конфігурації камери. На рисунку 3.16 фрагмент коду визначає конфігурацію для підключення камери до ESP32. Таким чином задаються порти для підключення сигнальних ліній камери, а також параметри захоплення зображень, такі як роздільна здатність, якість JPEG і частота XCLK. Ці налаштування дають змогу камері працювати коректно відповідно до вимог системи.

```

esp32_camera
6 #define PWDN_GPIO_NUM      32
7 #define RESET_GPIO_NUM    -1
8 #define XCLK_GPIO_NUM     0
9 #define SIOD_GPIO_NUM     26
10 #define SIOC_GPIO_NUM     27
11
12 #define Y9_GPIO_NUM        35
13 #define Y8_GPIO_NUM        34
14 #define Y7_GPIO_NUM        39
15 #define Y6_GPIO_NUM        36
16 #define Y5_GPIO_NUM        21
17 #define Y4_GPIO_NUM        19
18 #define Y3_GPIO_NUM        18
19 #define Y2_GPIO_NUM         5
20 #define VSYNC_GPIO_NUM     25
21 #define HREF_GPIO_NUM      23
22 #define PCLK_GPIO_NUM      22
23
24 /* Constant */
25 #define ESP32_CAMERA_RAW_FRAME_BUFFER_COLS      320
26 #define ESP32_CAMERA_RAW_FRAME_BUFFER_ROWS     240
27 #define ESP32_CAMERA_FRAME_BYTE_SIZE          3
28
29 static bool debug_nn = false;
30 static bool is_initialised = false;
31 uint8_t *snapshot_buf; //points to the output of the capture
32
33 static camera_config_t camera_config = {
34     .pin_pwdn = PWDN_GPIO_NUM,
35     .pin_reset = RESET_GPIO_NUM,
36     .pin_xclk = XCLK_GPIO_NUM,
37     .pin_sscb_sda = SIOD_GPIO_NUM,
38     .pin_sscb_scl = SIOC_GPIO_NUM,
39
40     .pin_d7 = Y9_GPIO_NUM,

```

Рисунок 3.16 – Ініціалізація та конфігурація камери

Функція `esp32_camera_init()`, що наведена на рисунку 3.17, ініціалізує камеру з використанням заданої конфігурації. Вона перевіряє, чи була камера вже ініціалізована, і якщо ні, то ініціалізує її за допомогою функції `esp_camera_init()`. Після успішної ініціалізації камера налаштовується для

оптимального відображення (переворот зображення і налаштування яскравості).

```

148 // Setup image sensor & start streaming
149 bool esp32_camera_init(void) {
150
151     if (is_initialised) return true;
152
153     //initialize the camera
154     esp_err_t err = esp_camera_init(&camera_config);
155     if (err != ESP_OK) {
156         Serial.printf("Camera init failed with error 0x%x\n", err);
157         return false;
158     }
159
160     sensor_t * s = esp_camera_sensor_get();
161     // initial sensors are flipped vertically and colors are a bit saturated
162     if (s->id.PID == OV3660_PID) {
163         s->set_vflip(s, 1); // flip it back
164         s->set_brightness(s, 1); // up the brightness just a bit
165         s->set_saturation(s, 0); // lower the saturation
166     }
167
168     is_initialised = true;
169     return true;
170 }

```

Рисунок 3.17 – Налаштування параметрів камери

Функція `esp32_camera_capture()`, що наведена на рисунку 3.18, виконує захоплення зображення з камери. Спочатку вона перевіряє, чи була камера ініціалізована. Потім захоплюється зображення у форматі JPEG, яке конвертується у формат RGB888 і зберігається в буфер `snapshot_buf`. Якщо розмір захопленого зображення відрізняється від необхідного, зображення буде рескальовано.

```

202 bool esp32_camera_capture(uint32_t img_width, uint32_t img_height, uint8_t *out_buf) {
203     bool do_resize = false;
204
205     if (!is_initialised) {
206         esp32_printf("ERR: Camera is not initialized\r\n");
207         return false;
208     }
209
210     camera_fb_t *fb = esp_camera_fb_get();
211
212     if (!fb) {
213         esp32_printf("Camera capture failed\n");
214         return false;
215     }
216
217     bool converted = fmt2rgb888(fb->buf, fb->len, PIXFORMAT_JPEG, snapshot_buf);
218
219     esp_camera_fb_return(fb);
220
221     if(!converted){
222         esp32_printf("Conversion failed\n");
223         return false;
224     }
225
226     if ((img_width != ESP32_CAMERA_RAW_FRAME_BUFFER_COLS)
227         || (img_height != ESP32_CAMERA_RAW_FRAME_BUFFER_ROWS)) {
228         do_resize = true;
229     }
230
231     if (do_resize) {
232         id::image::processing::crop_and_interpolate_rgb888(
233         out_buf,
234         ESP32_CAMERA_RAW_FRAME_BUFFER_COLS,
235         ESP32_CAMERA_RAW_FRAME_BUFFER_ROWS,
236         out_buf,

```

Рисунок 3.18 – Захват зображення

Функція `loop()`, що наведена на рисунку 3.19, запускає основний цикл програми. У цьому циклі відбувається захоплення зображення, обробка даних і запуск класифікатора. Спочатку виділяється пам'ять для буфера `snapshot_buf`, потім відбувається захоплення зображення і його обробка. Після цього запускається класифікатор, і результати виводяться на консоль. Якщо класифікатор виявляє об'єкт, інформація про нього виводиться, інакше виводиться повідомлення про відсутність об'єктів.

```

88 void loop()
89 {
90
91     // instead of wait_ms, wait on the signal, this allows threads to cancel
92     if (esp32_sleep(5) != ESP32_IDENTIFICATION_OK) {
93         return;
94     }
95
96     snapshot_buf = (uint8_t*)malloc(ESP32_CAMERA_RAW_FRAME_BUFFER_COLS * ESP32_CAMERA_RAW_FRAME_BUFFER_ROWS * ESP32_CAMERA_FRAME_BYTE_SIZE);
97
98     // check if allocation was successful
99     if (snapshot_buf == nullptr) {
100         esp32_printf("ERR: Failed to allocate snapshot buffer!\n");
101         return;
102     }
103
104     id::signal_t signal;
105     signal.total_length = ESP32_CLASSIFIER_INPUT_WIDTH * ESP32_CLASSIFIER_INPUT_HEIGHT;
106     signal.get_data = &esp32_camera_get_data;
107
108     if (esp32_camera_capture((size_t)ESP32_CLASSIFIER_INPUT_WIDTH, (size_t)ESP32_CLASSIFIER_INPUT_HEIGHT, snapshot_buf) == false) {
109         esp32_printf("Failed to capture image\r\n");
110         free(snapshot_buf);
111         return;
112     }
113
114     // Run the classifier
115     esp32_identification_result_t result = { 0 };
116
117     ESP32_IDENTIFICATION_ERROR err = run_classifier(&signal, &result, debug_nn);
118     if (err != ESP32_IDENTIFICATION_OK) {
119         esp32_printf("ERR: Failed to run classifier (%d)\n", err);
120         return;
121     }

```

Рисунок 3.19 – Основний цикл програми

Наведені на рисунках 3.10 - 3.19 фрагменти коду відображають ключові моменти налаштування, ініціалізації та виконання завдань ідентифікації на платформі ESP32. Таким чином відбувається конфігурація камери, обробка зображень, а також інтеграція алгоритмів машинного навчання, які відповідають за розпізнавання та класифікацію об'єктів. Наприклад, початок роботи системи включає налаштування різноманітних параметрів камери, таких як частота, якість JPEG зображень, а також пінів для з'єднання з сенсором. Далі, завдяки функціям захоплення зображень та їх обробки, система виконує класифікацію, використовуючи навчені моделі.

На рисунку 3.20 продемонстровано ESP32- CAM та плату для ідентифікації.

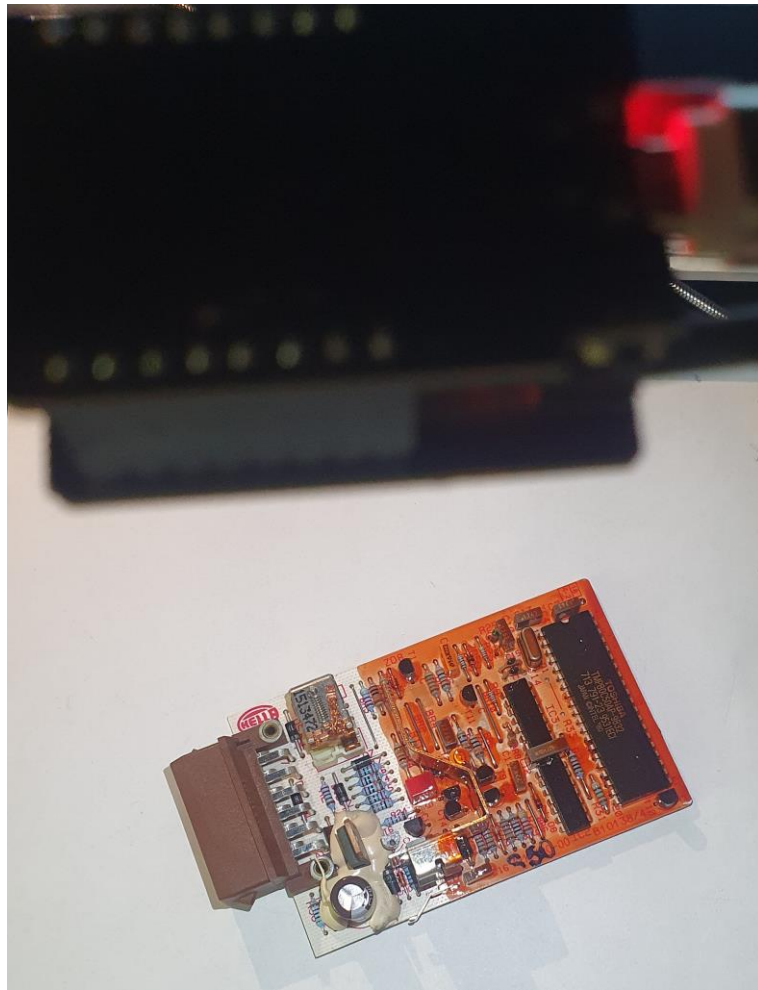


Рисунок 3.20 – Процес ідентифікації плати steuergerat

Результат ідентифікації виводиться до послідовного порта, що дозволяє легко відстежувати процес ідентифікації та аналізувати отримані дані. На рисунку 3.21 наведено приклад виводу результатів до послідовного порта. Цей вивід містить інформацію про розпізнаний об'єкт та рівень довіри до класифікації. Такий підхід дозволяє оперативно реагувати на зміни в середовищі та приймати необхідні коригувальні дії.

```

COM4
12:18:28.838 -> The steuergerat was found (confidence is 99).
12:18:29.578 -> The steuergerat was found (confidence is 99).
12:18:30.288 -> The steuergerat was found (confidence is 99).

```

Рисунок 3.21 – Результат ідентифікації плати steuergerat

На рисунку 3.22 продемонстровано процес ідентифікації плати phone, який здійснюється за допомогою ESP32- CAM. Як і у випадку з платою steuergerat, система виконує класифікацію об'єктів, використовуючи попередньо навчені моделі машинного навчання.

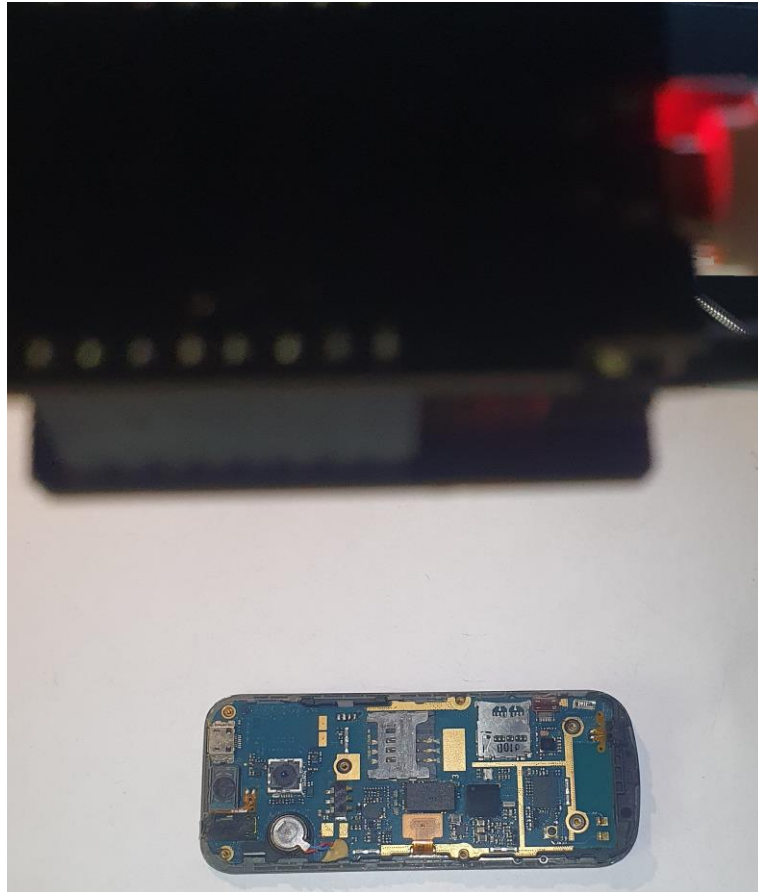
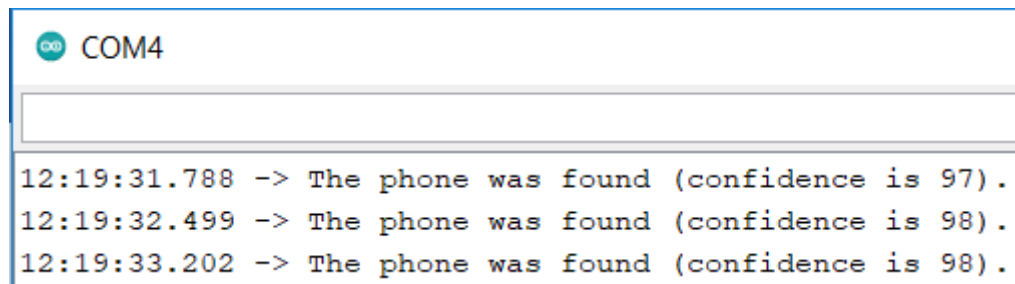


Рисунок 3.22 – Процес ідентифікації плати phone

Після завершення ідентифікації, результати виводяться до послідовного порта, що дозволяє зручно відстежувати та аналізувати процес ідентифікації. На рисунку 3.23 наведено приклад виводу результатів ідентифікації плати phone до послідовного порта. Цей вивід містить детальну інформацію про розпізнаний об'єкт та рівень довіри до класифікації.



```
COM4  
12:19:31.788 -> The phone was found (confidence is 97).  
12:19:32.499 -> The phone was found (confidence is 98).  
12:19:33.202 -> The phone was found (confidence is 98).
```

Рисунок 3.23 – Результат ідентифікації плати phone

Таким чином, створене програмне забезпечення демонструє високий рівень інтеграції сучасних технологій машинного навчання та обробки зображень, що дозволяє ефективно вирішувати завдання ідентифікації виробів на контрольних операціях.

3.4 Охорона праці

Охорона праці є невід'ємною частиною будь-якого виробничого процесу. При розробці програмного забезпечення для ідентифікації виробів на контрольних операціях технологічних процесів необхідно враховувати вимоги охорони праці для забезпечення безпечних умов роботи персоналу та запобігання виникненню нещасних випадків і професійних захворювань.

Шум і вібрація є одними з найпоширеніших шкідливих факторів на виробництві, де здійснюється контроль якості продукції. Джерелами шуму та вібрації можуть бути різноманітні технологічні процеси та обладнання, такі як робота верстатів, конвеєрів, пресів, компресорів, вентиляційних систем тощо.

Тривалий вплив підвищеного рівня шуму може призвести до розвитку професійної приглухуватості, а в окремих випадках – до повної втрати слуху. Крім того, шум може викликати порушення нервової та серцево-судинної систем, знижувати працездатність і концентрацію уваги, що підвищує ризик виникнення нещасних випадків на виробництві.

Вібрація, в свою чергу, може спричиняти вібраційну хворобу, яка характеризується ураженням нервової, судинної та кістково-суглобової систем організму. Локальна вібрація, що передається через руки працівника від ручного інструменту або деталей обладнання, може призвести до порушення чутливості пальців, болю та деформації суглобів.

Для забезпечення безпечних умов праці необхідно контролювати рівень шуму та вібрації відповідно до встановлених санітарних норм. Згідно з ДСН 3.3.6.037-99 "Санітарні норми виробничого шуму, ультразвуку та інфразвуку", гранично допустимі рівні (ГДР) шуму на робочих місцях не повинні перевищувати 80 дБА. Для окремих видів робіт, що потребують підвищеної уваги та зосередженості, ГДР шуму становлять 50-60 дБА [32].

Щодо вібрації, ДСН 3.3.6.039-99 "Державні санітарні норми виробничої загальної та локальної вібрації" встановлюють гранично допустимі значення віброшвидкості та віброприскорення для різних частотних діапазонів. Наприклад, для загальної вібрації в діапазоні частот 8-16 Гц ГДР віброшвидкості становить 0,56-0,28 м/с, а віброприскорення – 50-25 м/с²[33].

Для зниження рівня шуму та вібрації на виробництві можуть застосовуватися такі заходи:

- використання сучасного обладнання з низьким рівнем шуму та вібрації;
- застосування звукоізоляційних і віброізоляційних матеріалів для облицювання стін, стель і підлог виробничих приміщень;
- встановлення шумопоглинальних екранів і кожухів на шумних агрегатах;
- використання віброізоляційних опор і амортизаторів для обладнання;
- застосування засобів індивідуального захисту (ЗІЗ), таких як протишумні навушники та віброзахисні рукавиці;
- проведення регулярного технічного обслуговування та ремонту обладнання для запобігання підвищенню рівня шуму та вібрації;

– організація раціональних режимів праці та відпочинку, обмеження часу роботи в умовах підвищеного шуму та вібрації.

Крім того, необхідно проводити періодичний контроль рівня шуму та вібрації на робочих місцях, а також забезпечувати своєчасне проходження працівниками медичних оглядів для виявлення ранніх ознак професійних захворювань, пов'язаних з впливом цих шкідливих факторів.

В свою чергу освітлення відіграє ключову роль у забезпеченні безпечних та комфортних умов праці. Недостатнє або неправильно організоване освітлення може призвести до зорового дискомфорту, втоми очей, зниження продуктивності праці та підвищення ризику виникнення нещасних випадків. Згідно з ДБН В.2.5-28:2018 "Природне і штучне освітлення", норми освітленості для виробничих приміщень, де здійснюється контроль якості продукції, становлять 500-1000 лк залежно від характеру виконуваних робіт [34].

Окрім впливу на здоров'я та працездатність персоналу, освітлення також відіграє важливу роль у процесі ідентифікації виробів за допомогою ESP32-CAM. Недостатнє або нерівномірне освітлення може призвести до помилок розпізнавання, спотворення кольорів і зниження точності ідентифікації. Для забезпечення надійної роботи системи ідентифікації необхідно створити оптимальні умови освітлення, які включають:

- достатній рівень освітленості робочої зони;
- рівномірний розподіл світла без різких тіней і відблисків;
- правильний вибір типу світильників і ламп з урахуванням спектральних характеристик і кольоропередачі;
- регулярне очищення світильників і заміну ламп для підтримання необхідного рівня освітленості.

Мікроклімат виробничих приміщень також має значний вплив на здоров'я працівників і якість виконуваних робіт. Основними параметрами мікроклімату є температура, відносна вологість і швидкість руху повітря. Відповідно до ДСН 3.3.6.042-99 "Санітарні норми мікроклімату виробничих

приміщень", оптимальні значення температури повітря для категорії робіт середньої важкості становлять 18-20°C у холодний період року та 21-23°C у теплий період. Відносна вологість повітря повинна бути в межах 40-60%, а швидкість руху повітря – не більше 0,2 м/с [35].

Недотримання санітарних норм мікроклімату може призвести до погіршення самопочуття працівників, зниження працездатності, розвитку респіраторних захворювань та інших проблем зі здоров'ям. Крім того, несприятливі мікрокліматичні умови можуть негативно впливати на роботу ESP32-CAM та інших електронних компонентів системи ідентифікації виробів.

Надмірна вологість повітря може призвести до конденсації вологи на електронних платах і контактах, що спричиняє корозію, окислення та погіршення електричних характеристик. Це, у свою чергу, може викликати збої в роботі обладнання, спотворення сигналів і зниження надійності системи в цілому. Висока температура повітря також може призвести до перегріву електронних компонентів, скорочення терміну їх служби та виникнення помилок у процесі ідентифікації виробів.

Для забезпечення оптимальних мікрокліматичних умов у виробничих приміщеннях необхідно застосовувати такі заходи:

- встановлення ефективних систем вентиляції та кондиціонування повітря;
- регулярне провітрювання приміщень;
- використання зволожувачів або осушувачів повітря для підтримання оптимальної вологості;
- теплоізоляція стін, стель і підлог для зменшення впливу зовнішніх температурних коливань;
- раціональне розміщення робочих місць з урахуванням особливостей технологічного процесу та джерел тепла;
- забезпечення працівників спеціальним одягом і засобами індивідуального захисту відповідно до умов праці.

Для контролю параметрів мікроклімату необхідно використовувати спеціальні прилади, такі як термометри, психрометри та анемометри. Регулярний моніторинг мікрокліматичних умов дозволяє своєчасно виявляти відхилення від санітарних норм і вживати необхідних заходів для їх усунення.

Окрім забезпечення оптимальних умов освітлення та мікроклімату для персоналу, необхідно також враховувати вимоги до експлуатації ESP32-CAM та інших електронних компонентів системи ідентифікації виробів. Зокрема, слід дотримуватися таких рекомендацій:

- забезпечити надійний захист електронних пристроїв від пилу, вологи та інших забруднень;

- уникати встановлення обладнання в місцях з підвищеною температурою або вологістю;

- використовувати спеціальні корпуси або бокси з вентиляцією для запобігання перегріву електронних компонентів;

- регулярно проводити очищення та технічне обслуговування обладнання для підтримання його працездатності;

- забезпечити надійне заземлення та електромагнітну сумісність електронних пристроїв для запобігання завадам і збоям у роботі.

Дотримання санітарних норм і правил, а також реалізація комплексу технічних та організаційних заходів дозволять мінімізувати негативний вплив навколишніх факторів на здоров'я працівників і забезпечити безпечні умови праці на виробництві, де здійснюється контроль якості продукції.

ВИСНОВКИ

У ході виконання кваліфікаційної роботи було досягнуто поставленої мети – розроблено програмне забезпечення для ідентифікації виробів на контрольних операціях технологічних процесів, що дозволяє підвищити точність та надійність контролю якості плат.

Для досягнення поставленої мети було вирішено наступні завдання:

- проаналізовано типове виробництво процесу виготовлення друкованих плат, що дало змогу зрозуміти особливості та вимоги до системи ідентифікації виробів на різних етапах виробничого процесу;

- проведено аналіз процесу збору даних про вироби на різних етапах виробництва, що дозволило визначити необхідні параметри та характеристики для ефективною ідентифікації;

- розроблено структуру програмного забезпечення для ідентифікації виробів на контрольних операціях технологічних процесів, що забезпечує модульність та гнучкість системи;

- обґрунтовано метод збору даних, що дозволяє отримувати якісні та інформативні зображення виробів для подальшої обробки та аналізу;

- проведено збір та аналіз даних, що стали основою для навчання та тестування розробленої системи ідентифікації;

- розроблено алгоритм ідентифікації виробів на основі нейронної мережі MobileNetV2, що забезпечує високу точність та швидкість розпізнавання об'єктів.

Розроблене програмне забезпечення є доцільним та актуальним рішенням для сучасних виробничих підприємств, оскільки воно дозволяє автоматизувати процес ідентифікації виробів, зменшити кількість помилок та підвищити ефективність контролю якості. Система здатна працювати в режимі реального часу, що є важливим фактором для оптимізації виробничих процесів.

Впровадження розробленого програмного забезпечення може принести значні економічні вигоди для підприємств, скорочуючи витрати на ручний контроль та зменшуючи кількість браку. Крім того, точна ідентифікація виробів дозволяє покращити відстеження виробничих потоків та управління запасами.

Перспективами подальшого розвитку даної роботи є інтеграція системи ідентифікації з іншими виробничими системами, такими як ERP та MES, для створення комплексного рішення з управління виробництвом. Також, можливе вдосконалення алгоритмів розпізнавання для підвищення точності ідентифікації в умовах різноманітності виробів та змінних умов освітлення.

Отримані результати дослідження можуть бути використані не лише в галузі виробництва друкованих плат, але й в інших сферах промисловості, де необхідна точна та швидка ідентифікація об'єктів. Розроблене програмне забезпечення є гнучким та адаптивним, що дозволяє налаштувати його під специфічні потреби різних виробничих процесів.

Таким чином, розроблене програмне забезпечення для ідентифікації виробів на контрольних операціях технологічних процесів є важливим кроком на шляху до створення "розумних" виробництв та реалізації концепції "Індустрії 4.0" в Україні та світі. Його впровадження сприятиме підвищенню конкурентоспроможності вітчизняних підприємств та зміцненню їх позицій на глобальному ринку.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. ДСТУ 3008-15. Документація. Звіти у сфері науки та техніки. структура та правила оформлення. – Введ. 2015-06-22. – К. Держстандарт України, 2017 – 29 с.
2. Методичні вказівки з підготовки кваліфікаційної роботи для здобувачів першого (бакалаврського) рівня вищої освіти денної і заочної форми навчання спеціальності 151 «Автоматизація та комп'ютерно-інтегровані технології» освітньої програми «Автоматизація та комп'ютерно-інтегровані технології» / Упоряд.: І.Ш. Невлюдов, О.І. Филипенко, О.В. Токарева, С.П. Новоселов, О.В. Сичова. Харків: ХНУРЕ, 2023. – 64 с.
3. Невлюдов І.Ш. Комп'ютерно-інтегровані технології виробництва технічних засобів автоматизації. Частина 1: підручник. Харків: ФОП Панов А.М., 2021. – 604 с.
4. Sotnik S. V. Analysis of design process of automated fire protection system / S. V. Sotnik, Y. R. Vasylchenko // Automation, electronics and robotics (AERT-2023). – 2023. – p. 59-62.
5. Невлюдов І.Ш. Механізми технічних засобів автоматизації (довідкові матеріали з курсового і дипломного проектування): навчальний посібник. / І.Ш. Невлюдов, В.І. Роменський, І.О. Яшков. – Харків: ХНУРЕ, 2021. – 292 с.
6. Sotnik S. Modern Integrated Software Development Environments // International Journal of Academic and Applied Research (IJAAR) / S. Sotnik, V. Lyashenko, T. Schakurova. – 2021. – Vol. 5, Issue 10. – pp. 157-161.
7. SMTmax. QM1100 Automatic Pick and Place Machine with Vision system, 2015. YouTube. URL: <https://www.youtube.com/watch?v=ZngMoV325Sw> (date of access: 10.04.2024).
8. Невлюдов І. Ш. Технічні засоби автоматизації: Підручник / І.Ш. Невлюдов, А.О. Андрусевич, О.І. Филипенко, Н.П. Демська, С.П. Новоселов. – Кривий Ріг : Криворізький коледж НАУ, 2019. – 366 с.

9. Sotnik S. V. Design features of control panels and consoles in automation systems // 9th International scientific and practical conference “Science and innovation of modern world” (May 18-20, 2023) Cognum Publishing House, London, United Kingdom / S. V. Sotnik, K. S. Redkin. – 2023, pp. 201-205.

10. Jonathan C. Automatic identification and data collection systems. London : McGraw-Hill, 2024. 245 p.

11. Yechevskiy A.D. System of monitoring and control of microclimate parameters in office premises / A.D. Yechevskiy // Автоматизація та Приладобудування («Automation and Development of Electronic Devices» ADED-2024) [Електронний ресурс]: збірник студентських наукових статей / Харківський національний університет радіоелектроніки; [редкол.: І.Ш. Невлюдов та ін.]. – Харків: ХНУРЕ, 2024. – Вип. 1. – С. 122-126.

12. Основні методи ідентифікації об'єктів. ni.biz.ua - сайт навчальної інформації для українських студентів. URL: http://ni.biz.ua/4/4_6/4_60904_osnovnie-metodi-identifikatsii-ob-ektov.html (дата звернення: 10.04.2024).

13. What is the Canny edge detection algorithm? Knowledge Columns. URL: <https://zhuanlan.zhihu.com/p/398008427> (date of access: 10.04.2024).

14. Розробка безконтактного методу ідентифікацій виробів на промисловій лінії. ElAr :: Головна. URL: <https://openarchive.nure.ua/entities/publication/cf25a926-af7e-4dad-8756-fd324d7b0d9b> (дата звернення: 12.04.2024).

15. Придятько Д.Р. Огляд методів розпізнавання об'єктів за допомогою систем технічного зору. Автоматизація та Приладобудування («Automation and Development of Electronic Devices» ADED-2023) [Електронний ресурс] : збірник студентських наукових статей / Харківський національний університет радіоелектроніки ; [редкол.: І.Ш. Невлюдов та ін.]. – Харків : ХНУРЕ, 2023. – Вип. 2. – с. 7-10.

16. Contributors to Wikimedia projects. Convolutional neural network - Wikipedia. Wikipedia, the free encyclopedia. URL:

https://en.wikipedia.org/wiki/Convolutional_neural_network (date of access: 20.04.2024).

17. Єчевський А. Д. Система моніторингу та управління параметрами мікроклімату в офісних приміщеннях. Автоматизація та Приладобудування («Automation and Development of Electronic Devices» ADED-2023) [Електронний ресурс] : збірник студентських наукових статей / Харківський національний університет радіоелектроніки ; [редкол.: І.Ш. Невлюдов та ін.]. – Харків : ХНУРЕ, 2023. – Вип. 2. – с. 159-163.

18. Токарева О. В. Теорія автоматичного управління : підручник. Харків: ФОП Панов А. М., 2020. – 346 с.

19. Yechevskiy A. D. Analysis of collecting data process on products at different stages of production. Стан, досягнення та перспективи інформаційних систем і технологій /Матеріали XXIV Всеукраїнської науково-технічної конференції молодих вчених,аспірантів та студентів. Одеса, 18-19 квітня 2024 р. – Одеса, Видавництво ОНТУ,2024 р. – с. 84-86.

20. Учасники проектів Вікімедіа. Кордонні обчислення – Вікіпедія. Вікіпедія. URL: https://uk.wikipedia.org/wiki/Кордонні_обчислення (дата звернення: 03.05.2024).

21. MobileNetV2: Inverted Residuals and Linear Bottlenecks. arXiv.org. URL: <https://arxiv.org/abs/1801.04381> (date of access: 07.05.2024).

22. Учасники проектів Вікімедіа. ReLU – Вікіпедія. Вікіпедія. URL: <https://uk.wikipedia.org/wiki/ReLU> (дата звернення: 15.05.2024).

23. ESP32-CAM модуль з WiFi та Bluetooth, камерою OV2640 та материнською платою. prom.ua. URL: <https://prom.ua/ua/p1805655226-esp32-cam-modul.html> (дата звернення: 21.05.2024).

24. ESP32-Cam Wi-Fi, Bluetooth, модуль камери OV2640, плата розробника. "Ardu.prom.ua". URL: <https://ardu.prom.ua/ua/p1224387627-esp32-cam-bluetooth.html> (дата звернення: 21.05.2024).

25. Модуль камери 2Мп OV2640 для ESP32-CAM (FOV70). Arduino в Україні. URL: <https://arduino.ua/ru/prod4514-modyl-kameri-2mp-ov2640-dlya-esp32-cam-fov70> (дата звернення: 22.05.2024).

26. ESP32 CAM MB programmierung adapter karte ch340 seriell zu usb wifi + bluetooth kompatibel 4,7 v 5,3 v für ESP32 CAM modul – AliExpress. aliexpress.com. URL: <https://www.aliexpress.com/item/1005006835516026.html> (дата звернення: 22.05.2024).

27. Arduino IDE. Flathub. URL: <https://flathub.org/uk/apps/cc.arduino.arduinoide> (дата звернення: 23.05.2024).

28. PlatformIO: Your Gateway to Embedded Software Development Excellence. PlatformIO. URL: <https://platformio.org/platformio-ide> (date of access: 23.05.2024).

29. Get started with ESP-IDF (Windows 10 & 11) - EmbeddedTutorials. EmbeddedTutorials. URL: <https://embeddedtutorials.com/eps32/get-started-with-esp-idf-windows/> (date of access: 23.05.2024).

30. MicroPython IDEs for ESP32 and ESP8266 | Random Nerd Tutorials. Random Nerd Tutorials. URL: <https://randomnerdtutorials.com/micropython-ides-esp32-esp8266/> (date of access: 23.05.2024).

31. Espruino Web IDE. Chrome Web Store. URL: <https://chromewebstore.google.com/detail/espruino-web-ide/bleoifhkdbjfbobjackfdifdneehpo?hl=zh-TW> (date of access: 23.05.2024).

32. ДСН 3.3.6.037-99 Санітарні норми виробничого шуму, ультразвуку та інфразвуку. БУДСТАНДАРТ Online - нормативні документи будівельної галузі України. URL: https://online.budstandart.com/ua/catalog/doc-page?id_doc=48147 (дата звернення: 01.06.2024).

33. ДСН 3.3.6.039-99 Державні санітарні норми виробничої загальної та локальної вібрації. БУДСТАНДАРТ Online - нормативні документи будівельної галузі України. URL: https://online.budstandart.com/ua/catalog/doc-page.html?id_doc=6372 (дата звернення: 01.06.2024).

34. ДБН В.2.5-28:2018 Природне і штучне освітлення. БУДСТАНДАРТ Online - нормативні документи будівельної галузі України. URL: https://online.budstandart.com/ua/catalog/doc-page.html?id_doc=79885 (дата звернення: 01.06.2024).

35. ДСН 3.3.6.042-99 Санітарні норми мікроклімату виробничих приміщень. БУДСТАНДАРТ Online - нормативні документи будівельної галузі України. URL: https://online.budstandart.com/ua/catalog/doc-page.html?id_doc=14283 (дата звернення: 01.06.2024).