

Факультет комп'ютерно-інформаційних систем і програмної інженерії

(повна назва факультету)

Кафедра комп'ютерних систем та мереж

(повна назва кафедри)

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

Магістр

(назва освітнього ступеня)

на тему: Методи та засоби адаптивного керування параметрами 3D принтера
на основі аналізу процесу друку.

Виконав: студент(ка) 6 курсу, групи СІм-61

спеціальності 123 «Комп'ютерна інженерія»

(шифр і назва спеціальності)

	<u>Кобрин О.В.</u> (підпис)	<u>Кобрин О.В.</u> (прізвище та ініціали)
Керівник	<u>Стадник Н.Б.</u> (підпис)	<u>Стадник Н.Б.</u> (прізвище та ініціали)
Нормоконтроль	<u>Тиш Є.В.</u> (підпис)	<u>Тиш Є.В.</u> (прізвище та ініціали)
Завідувач кафедри	<u>Осухівська Г.М.</u> (підпис)	<u>Осухівська Г.М.</u> (прізвище та ініціали)
Рецензент	<u></u> (підпис)	<u></u> (прізвище та ініціали)

ЗАТВЕРДЖУЮ

Завідувач кафедри

Осухівська Г.М

(підпис)

(прізвище та ініціали)

«17» листопада 2025р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

на здобуття освітнього ступеня магістр

(назва освітнього ступеня)

за спеціальністю 123 «Комп'ютерна інженерія»

(шифр і назва спеціальності)

студенту Кобрин Олександр Віталійович

(прізвище, ім'я, по батькові)

1. Тема роботи Методи та засоби адаптивного керування параметрами 3D принтера на основі аналізу процесу друку.

Керівник роботи Стадник Наталія Богданівна

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

Затверджені наказом ректора від «14» листопада 2025 року № 4/7-987

2. Термін подання студентом завершеної роботи 16 грудня 2025 р.

3. Вихідні дані до роботи Розроблений програмний застосунок PrintGuard для моніторингу процесу 3D-друку на основі моделі YOLOv8, який аналізує відеопотік з камери.

4. Зміст роботи (перелік питань, які потрібно розробити)
- Вступ
- 1 Аналітичний огляд предметної області та постановка задачі
- 2 Теоретичні та методичні основи аналізу процесу друку
- 3 Практична частина
- 4 Охорона праці та безпека в надзвичайних ситуаціях
- Висновки
5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, слайдів)
- 1 Актуальність дослідження
- 2 Завдання об'єкт і предмет наукова новизна і практична цінність дослідження
- 3 Модель функціонування системи
- 4 Структурна схема системи
- 5 Схема взаємодії модулів програмного забезпечення
- 6 Блок-схема алгоритму роботи системи
- 7 Результат роботи системи
- 8 Висновки

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
<i>Охорона праці</i>	<i>Осухівська Г.М., зав.каф. КС</i>	<i>04.12.2025</i>	
<i>Безпека в надзвичайних ситуаціях</i>			

7. Дата видачі завдання 17.11.2025

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1.	<i>Збір та розмітка фото/відео по класах і папках.</i>	<i>17.11.2025 – 20.11.2025</i>	<i>Викон.</i>
2.	<i>Навчання моделі YOLOv8 на основі підготовлених кадрів.</i>	<i>21.11.2025 – 25.11.2025</i>	<i>Викон.</i>
3.	<i>Тестування навченої моделі на тестових кадрах/відео.</i>	<i>25.11.2025 – 26.11.2025</i>	<i>Викон.</i>
4.	<i>Перенавчання моделі з доповненням кадрів; зміна кількості epoch і batch size.</i>	<i>26.11.2025 – 28.11.2025</i>	<i>Викон.</i>
5.	<i>Повторне тестування моделі.</i>	<i>28.11.2025 – 30.11.2025</i>	<i>Викон.</i>
6.	<i>Збір результатів навчання для кваліфікаційної роботи.</i>	<i>30.11.2025</i>	<i>Викон.</i>
7.	<i>Розробка програми PrintGuard для використання готової моделі.</i>	<i>01.12.2025 – 02.12.2025</i>	<i>Викон.</i>
8.	<i>Додавання сповіщень та підключення до принтера.</i>	<i>02.12.2025 – 04.12.2025</i>	<i>Викон.</i>
9.	<i>Тестування програми PrintGuard.</i>	<i>04.12.2025 – 06.12.2025</i>	<i>Викон.</i>
10.	<i>Доповнення кваліфікаційної роботи .</i>	<i>06.12.2025 – 12.12.2025</i>	<i>Викон.</i>
11.	<i>Попередній захист кваліфікаційної роботи магістра</i>	<i>16.12.2025</i>	<i>Викон.</i>
12.	<i>Захист кваліфікаційної роботи магістра</i>	<i>26.12.2025</i>	<i>Викон.</i>

Студент

(підпис)

Кобрин О.В

(прізвище та ініціали)

Керівник роботи

Стадник Н.Б

АНОТАЦІЯ

Кобрин О.В. Методи та засоби адаптивного керування параметрами 3D-принтера на основі аналізу процесу друку: робота на здобуття кваліфікаційного ступеня магістра: спец. 123 - комп'ютерна інженерія / наук. кер. Стадник Н.Б. Тернопіль: Тернопільський національний технічний університет імені Івана Пулюя, 2025.

Ключові слова: 3D-друк, FDM, комп'ютерний зір, моніторинг процесу, нейромережеве розпізнавання, детекція, датасет, precision, recall, mAP.

Кваліфікаційна робота магістра присвячена дослідженню методів і засобів адаптивного керування параметрами 3D-принтера на основі аналізу процесу друку за даними комп'ютерного зору. Актуальність теми зумовлена потребою підвищення стабільності FDM-друку та зменшення частки браку шляхом оперативного виявлення візуальних відхилень без втручання у конструкцію принтера. У роботі виконано огляд підходів до контролю якості 3D-друку та обґрунтовано доцільність використання нейромережевої детекції для автоматизованого аналізу кадрів з камери.

Для реалізації підходу підготовлено набір розмічених зображень, зібраний з відкритих джерел та приведений до узгоджених умов зйомки (ракурс, відстань, освітлення). Налаштовано процес навчання моделі детекції та проведено експериментальну оцінку якості за показниками precision, recall та mAP. Отримані результати підтверджують можливість виявлення візуальних відхилень у процесі FDM-друку та придатність запропонованого підходу для практичного застосування у вигляді програмного модуля моніторингу, що може працювати з потоковим відео або послідовністю кадрів і формувати обґрунтовані рекомендації щодо налаштувань, які впливають на якість виробу.

ANNOTATION

Kobryn O.V. Methods and Tools for Adaptive Control of 3D Printer Parameters Based on Printing Process Analysis: thesis for obtaining the Master's degree: specialty 123 - Computer Engineering / academic supervisor Stadnyk N.B. Ternopil: Ivan Puluj Ternopil National Technical University, 2025.

Keywords: 3D printing, FDM, computer vision, process monitoring, neural network recognition, object detection, dataset, precision, recall, mAP.

The master's comprehensive thesis is devoted to the study of methods and tools for adaptive control of 3D printer parameters based on computer-vision analysis of the printing process. The relevance of the topic is determined by the need to improve FDM printing stability and reduce the reject rate through timely detection of visual deviations without interfering with the printer design. The thesis reviews approaches to 3D printing quality control and substantiates the feasibility of using neural-network-based object detection for automated analysis of camera frames.

To implement the proposed approach, a labeled image dataset was prepared from open sources and brought to consistent acquisition conditions (camera angle, distance, and lighting). The training process for the detection model was configured, and an experimental evaluation was carried out using precision, recall, and mAP metrics. The obtained results confirm the feasibility of detecting visual deviations in the FDM printing process and the suitability of the proposed approach for practical application as a software monitoring module that can operate on a video stream or a sequence of frames and provide substantiated recommendations regarding settings that affect the final print quality.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ	9
ВСТУП	10
РОЗДІЛ 1 АНАЛІТИЧНИЙ ОГЛЯД ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ	12
1.1. Особливості процесу FDM-друку як об'єкта моніторингу.....	12
1.2. Причини виникнення відхилень у процесі друку та їх вплив на якість виробу.....	15
1.3. Підходи до контролю процесу FDM-друку та виявлення відхилень.....	19
1.4. Комп'ютерний зір у задачах моніторингу адитивного виробництва.....	23
РОЗДІЛ 2 ТЕОРЕТИЧНІ ТА МЕТОДИЧНІ ОСНОВИ АНАЛІЗУ ПРОЦЕСУ ДРУКУ.....	25
2.1. Формалізація задачі виявлення відхилень у процесі FDM-друку за зображеннями.....	25
2.2. Вимоги до вхідних даних та формування зони інтересу.....	27
2.3. Попередня обробка зображень та приведення до узгодженого формату.....	28
2.4. Формування датасету та принципи збирання даних з різних ресурсів.....	30
2.5. Розмітка даних для задачі детекції та узгодження правил анотації.....	32
2.6. Аугментації та підвищення узагальнюваності моделі.....	34
2.7. Обґрунтування вибору підходу детекції об'єктів.....	36
2.8. Загальні принципи нейромережевої детекції та особливості підходу YOLO....	37
2.9. Навчання моделі детекції та постановка оптимізаційної задачі та компоненти втрат.....	38
2.10. Метрики оцінки якості та принципи інтерпретації результатів.....	40
РОЗДІЛ 3 ПРАКТИЧНА ЧАСТИНА.....	42
3.1. Мета та постановка експериментальних досліджень.....	42
3.2. Сценарії використання системи та вимоги до практичного прототипу.....	43

3.3. Формування набору даних та структура проєкту для навчання.....	43
3.4. Попередня обробка кадрів та узгодження розміру входу детектора.....	45
3.5. Налаштування навчання та параметри експериментів.....	45
3.6. Результати навчання №1.....	46
3.7. Якісний аналіз прикладів детекції для навчання №1.....	48
3.8. Результати навчання №2 криві навчання, узагальнені метрики.....	49
3.9. Матриця плутанини для навчання №2 та інтерпретація помилок.....	50
3.10. Аналіз залежностей precision, recall і F1 від порога confidence та вибір робочого порога.....	52
3.11. Крива precision-recall та порівняння класів за складністю детекції.....	54
3.12. Якісний аналіз прикладів детекції для навчання №2 та порівняння з навчанням №1.....	55
3.13. Аналіз навчальних батчів та контроль якості даних.....	57
3.14. Прототип прикладного використання: перевірка папки кадрів та формування візуалізованих результатів.....	59
3.15. Прототип для відеопотоку та часова фільтрація спрацювань.....	60
3.16. Обмеження, фактори впливу та напрями удосконалення.....	60
3.17. Настільний застосунок PrintGuardian як приклад інтеграції детектора у систему моніторингу.....	61
3.18. Візуалізація результатів детекції та інтерфейс користувача прототипу PrintGuardian.....	62
3.19. Отримання повідомлення про відхилення та передача керувальної команди через OctoPrint.....	63
РОЗДІЛ 4 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ...	66
4.1. Охорона праці	66
4.2. Безпека в надзвичайних ситуаціях.....	68
4.2.1. Пожежна безпека та запобігання аваріям.....	69
4.2.2. Ураження електричним струмом.....	70

4.2.3.	Травмування	персоналу	та	перша	
	допомога.....				70
4.2.4.	Організаційні заходи та планування дій у НС.....				71
	ВИСНОВКИ.....				72
	СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ.....				74
	Додаток А	Тези конференції			
	Додаток Б	Блок схема алгоритму роботи системи			
	Додаток В	Схема взаємодії модулів програмного забезпечення			
	Додаток Г	Структурна схема системи			
	Додаток Д	Лістинг програми			

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

FDM (англ. Fused Deposition Modeling) технологія пошарового нанесення розплавленого термопластику.

3D-друк - адитивне виготовлення виробів шляхом пошарового формування.

G-code - керувальний код, що описує траєкторії руху та режими друку.

CAD (англ. Computer-Aided Design) комп'ютерне проектування моделі.

CNN - (англ. Convolutional Neural Network) згорткова нейронна мережа.

YOLO (англ. You Only Look Once) сімейство моделей детекції об'єктів у зображенні.

IoU (англ. Intersection Over Union) міра перекриття прогнозованого та еталонного прямокутника.

mAP (англ. Mean Average Precision) усереднений показник точності детекції.

Precision - частка правильних спрацьовувань серед усіх спрацьовувань моделі.

Recall - частка виявлених об'єктів серед усіх еталонних об'єктів.

FPS (англ. Frames Per Second) кількість оброблених кадрів за секунду.

CPU (англ. Central Processing Unit) центральний процесор.

GPU (англ. Graphics Processing Unit), графічний процесор.

Dataset - набір даних, що містить зображення та розмітку для навчання та перевірки.

ВСТУП

Актуальність роботи. Сучасні настільні 3D-принтери класу FDM широко застосовуються для прототипування та виготовлення функціональних деталей, однак процес друку є чутливим до параметрів режиму та умов виконання. Відхилення, які проявляються у вигляді візуальних дефектів, часто виявляються із запізненням, що призводить до браку, втрат матеріалу та часу. Тому актуальним є використання комп'ютерного зору для оперативного моніторингу процесу друку та підтримки коригування параметрів без втручання у конструкцію принтера.

Мета кваліфікаційної роботи. Розробити та експериментально оцінити програмно-алгоритмічне рішення на основі комп'ютерного зору для моніторингу процесу FDM-друку з виявленням візуальних відхилень і формуванням обґрунтованих рекомендацій щодо коригування параметрів друку.

Завдання кваліфікаційної роботи:

- виконати аналіз сучасних підходів до моніторингу процесу 3D-друку та визначити вимоги до комп'ютерно-зорової системи;
- сформувати та підготувати набір даних із зображеннями процесу FDM-друку, зібраними з відкритих джерел, і привести їх до узгоджених умов зйомки;
- визначити вимоги до вхідних даних і структури розмітки для навчання моделі детекції;
- реалізувати й налаштувати процес навчання нейромережевої моделі детекції для аналізу кадрів;
- провести експериментальну оцінку якості моделі за метриками precision, recall та mAP і проаналізувати результати;
- сформувати рекомендації щодо практичного використання результатів моніторингу для підтримки коригування параметрів друку.

Об'єкт дослідження: комп'ютерно-зорова система моніторингу процесу FDM-друку як засіб виявлення візуальних відхилень та підтримки коригування параметрів 3D-принтера.

Предмет дослідження: методи та програмно-алгоритмічні засоби аналізу зображень процесу FDM-друку на основі нейромережевої детекції для ідентифікації візуальних відхилень і формування рекомендацій щодо налаштувань друку.

Методи дослідження: аналіз і узагальнення науково-технічних джерел для обґрунтування підходу та вимог; методи підготовки даних (відбір, структуризація, узгодження умов зйомки, розмітка); методи цифрової обробки зображень; методи машинного навчання і глибинного навчання для навчання моделі детекції; експериментальні методи для навчання, валідації та порівняння налаштувань; статистичні методи оцінювання якості за метриками precision, recall, mAP.

Наукова новизна дослідження: удосконалено підхід до моніторингу процесу FDM-друку шляхом використання комп'ютерного зору та нейромережевої детекції на уніфікованому наборі даних, сформованому з різних відкритих джерел і приведеному до узгоджених умов зйомки; набули подальшого розвитку підходи до оцінювання придатності моделі для практичного застосування через аналіз впливу структури датасету та умов зйомки на метрики precision, recall і mAP.

Практичне значення результатів: результати роботи можуть бути використані як програмний модуль моніторингу процесу FDM-друку, що аналізує потокове відео або послідовність кадрів та забезпечує виявлення візуальних відхилень і надання рекомендацій щодо коригування параметрів, які впливають на якість виробу; запропонована методика підготовки даних і оцінювання якості може застосовуватися під час розроблення подібних систем контролю якості в 3D-друці.

Публікації. Результати дослідження апробовано у вигляді тез доповідей на XIII науково-технічній конференції Тернопільського національного технічного університету імені Івана Пулюя "Інформаційні моделі, системи та технології" та XIV міжнародній науково-практичній конференції молодих учених та студентів "Актуальні задачі сучасних технологій".

Структура роботи. Робота складається з пояснювальної записки та графічної частини. Пояснювальна записка складається із вступу, 4 розділів, висновків, списку використаних джерел та додатків.

РОЗДІЛ 1

АНАЛІТИЧНИЙ ОГЛЯД ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ

1.1. Особливості процесу FDM-друку як об'єкта моніторингу

Процес FDM-друку базується на пошаровому формуванні виробу шляхом екструзії розплавленого термопластичного матеріалу через сопло друкувального вузла. Технологія FDM-друку належить до адитивних методів виготовлення, у яких виріб формується шляхом послідовного нанесення розплавленого термопластику у вигляді доріжок, що створюють шар заданої товщини. Основою процесу є керований перенос матеріалу через сопло екструдера з подальшим охолодженням та зчепленням з попереднім шаром. На відміну від традиційних методів механічної обробки, де геометрія формується шляхом видалення матеріалу, у FDM геометрія створюється шляхом додавання матеріалу, що робить стабільність формування кожного шару критичною для якості всього виробу.

З технологічної точки зору, FDM є системою, де одночасно взаємодіють теплові процеси, механіка переміщень і реологія розплаву. Термопластик у зоні хотенду нагрівається до робочої температури, в'язкість матеріалу змінюється, і розплав під тиском подається через сопло. Далі доріжка укладається на поверхню шару, формується контактна зона між шарами, і матеріал охолоджується. На цьому етапі важливими стають швидкість переміщення, фактичний об'єм поданого матеріалу, температура поверхні, на яку укладається доріжка, і умови охолодження. Навіть незначні відхилення в одному з цих параметрів можуть призвести до зміни ширини доріжки, зменшення або збільшення її товщини, порушення адгезії або появи дефектів поверхні.

На рисунку 1.1 наведено схематичне подання процесу FDM-друку, яке відображає основні елементи друкувального вузла та принцип пошарового формування виробу.

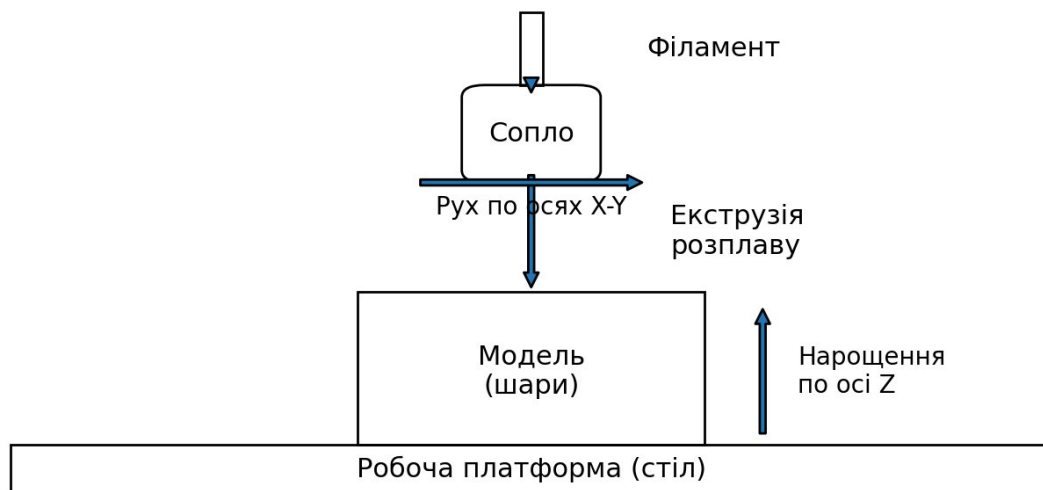


Рис. 1.1. Схематичне подання процесу FDM-друку

Як об'єкт моніторингу FDM-друк має кілька характерних особливостей. По-перше, процес є довготривалим, а тривалість друку може вимірюватися годинами або десятками годин. Це створює ризик того, що дефект виникне в середині процесу і буде виявлений із запізненням. По-друге, процес є багатofакторним: результати залежать не лише від параметрів у G-code, але й від фактичного стану системи, включаючи зношення механіки, стан сопла, чистоту платформи, вологість філаменту, стабільність живлення, потоки повітря та інші чинники. По-третє, значна частина відхилень проявляється саме у візуальній формі, тому камера спостереження може слугувати простим та інформативним каналом отримання даних про стан процесу.

Важливо також враховувати роль програмної підготовки друку. Під час нарізки моделі у слайсері формуються траєкторії, швидкості, режими подачі, ретракції, охолодження, параметри заповнення та інші установки. Формально процес друку є виконанням цих команд, однак реальний друк завжди відрізняється через затримки нагріву, інерцію рухомих вузлів, зміни навантаження на подачу та фізичні властивості матеріалу. Саме тому моніторинг має сенс навіть при коректно складеному G-code, оскільки він дозволяє фіксувати фактичний стан формування доріжок і шарів.

Під час використання комп'ютерного зору слід враховувати, що сцена друку є динамічною. У кадрі одночасно можуть бути присутні сопло, рухомі елементи, блискучі поверхні платформи, відблиски від освітлення, зміни кольору через нагрів або тіні від руху каретки. Такі фактори створюють шум для алгоритмів аналізу зображень. Разом із тим багато дефектів мають характерні структурні ознаки, що зберігаються у різних умовах, наприклад хаотична ниткоподібна структура "spaghetti" або тонкі перемички "stringing". Це робить задачу комп'ютерного зору практично реалізовною, якщо навчання виконано на достатньо різноманітних прикладах.

Ще одна ознака процесу як об'єкта моніторингу полягає у тому, що частина дефектів не виникає миттєво. Деякі відхилення накопичуються, наприклад поступове погіршення якості через вологий філамент, нестабільність температури або часткове забруднення сопла. У таких випадках окремий кадр може не бути достатньо інформативним, але послідовність кадрів дозволяє побачити тенденцію. Це важливо для формування правильної логіки прийняття рішень у системі моніторингу, де значення мають повторювані спрацювання та їх тривалість.

Таким чином, FDM-друк як об'єкт моніторингу поєднує технічну складність і високу інформативність візуального каналу. Це обґрунтовує вибір візійного підходу як основи для аналізу стану процесу та створює передумови для подальшого використання результатів аналізу в рамках адаптивного налаштування параметрів друку.

Практичний інтерес до FDM-друку пов'язаний із тим, що він доступний для малого виробництва, навчальних лабораторій і домашнього використання, однак стабільність процесу суттєво залежить від дисципліни налаштувань і умов експлуатації. На етапі запуску друку користувач має забезпечити коректний перший шар, адже саме він створює механічну «базу» для всіх наступних шарів. Типові помилки першого шару (надто мала/велика відстань сопла до столу, недостатня адгезія поверхні, неузгоджена температура столу і матеріалу) часто проявляються одразу у вигляді нерівномірних доріжок, «пропусків» або підриву країв. Якщо на цьому етапі процес не стабілізувати, то дефект зазвичай не зникає самотійно, а

переходить у більш критичні стани (зсув, відрив, намотування нитки на сопло). Саме тому моніторинг для FDM доцільно розглядати як механізм раннього попередження, який дає сигнал, коли ще можливо мінімізувати втрати часу та матеріалу.

На відміну від багатьох інших технологій адитивного виробництва, FDM має виражену «геометрію процесу»: траєкторії сопла та доріжки матеріалу є видимими, а отже можуть виступати інформаційними ознаками для комп'ютерного зору. Візуально спостерігаються: стабільність товщини та ширини доріжки, рівномірність заповнення, характерні деформації на дрібних елементах і містках, поява ниток на переходах, а також загальна поведінка деталі (підняття кутів, вібрації, відрив). При цьому процес є не просто «знімком», а безперервною послідовністю станів. У більшості реальних сценаріїв одиничний кадр не дає повної картини: короткочасне перекриття головкою або випадковий блиск може спотворити спостереження. Тому коректна система моніторингу повинна враховувати часовий контекст: підтверджувати дефект на серії кадрів, відкидати одиничні спрацювання та застосовувати паузи між повідомленнями.

Додатковою особливістю є чутливість FDM до умов навколишнього середовища. Протяги, різкі перепади температури, нестабільна вентиляція та навіть незначні коливання вологості можуть змінювати термічний режим шару і впливати на адгезію. З точки зору комп'ютерного зору це означає, що набір даних та модель повинні «бачити» друк у різних типових умовах: різне освітлення, фон, колір матеріалу, різна відстань камери. В іншому випадку модель може працювати лише у «лабораторному» сценарії й давати нестабільні результати при зміні умов.

1.2. Причини виникнення відхилень у процесі друку та їх вплив на якість виробу

Відхилення у процесі FDM-друку можна розглядати як наслідок порушення балансу між подачею матеріалу, рухом сопла та умовами формування шару. Якість друку визначається тим, наскільки доріжка пластику повторює задану траєкторію та наскільки стабільно відбувається зчеплення доріжок між собою і з попереднім

шаром. Коли один або кілька чинників виходять за допустимі межі, з'являються візуальні ознаки дефекту, а у готовому виробі це проявляється у вигляді шорсткості, порожнин, слабких шарових з'єднань або втрати геометричної точності.

Узагальнену логіку взаємозв'язку між причинами відхилень, візуальними проявами та наслідками наведено на рисунку 1.2.

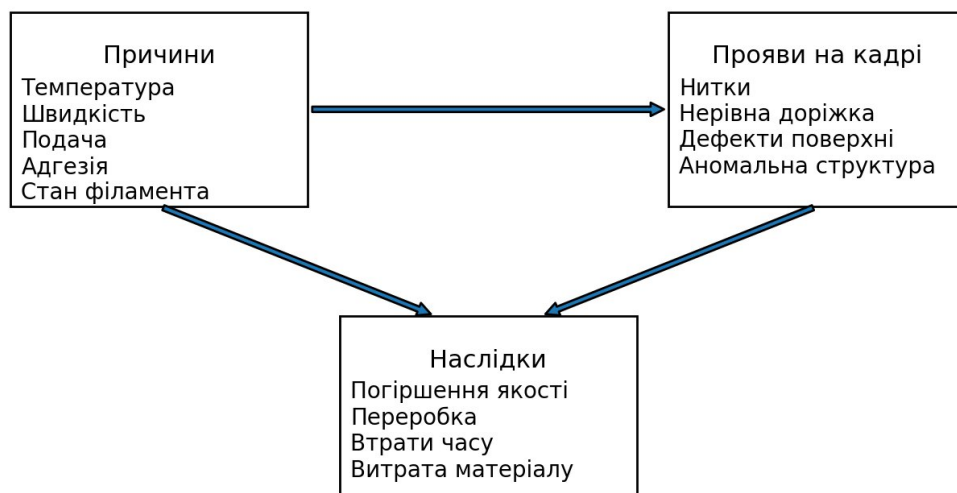


Рис. 1.2. Узагальнена логіка: причини відхилень, візуальні прояви та наслідки

Однією з ключових причин проблем є стан першого шару. Якщо перший шар не забезпечує стабільного зчеплення з платформою, подальше формування виробу стає нестійким. Причинами можуть бути неправильний зазор між соплом і столом, нерівність поверхні, забруднення, невідповідний режим температури платформи або надто висока стартова швидкість. Візуально недостатня адгезія може виглядати як відшарування кутів, "підняття" країв, нерівні доріжки першого шару або локальні розриви. Навіть якщо друк продовжується, такі явища створюють умови для подальших дефектів.

Температурний режим сопла визначає в'язкість розплаву та його здатність формувати однорідні доріжки. При недостатній температурі матеріал стає в'язким, і може з'явитися неоекструзія, пропуски або погане зчеплення між шарами. При надмірній температурі зростає текучість, що сприяє витіканню під час переміщень

без екструзії, збільшенню ниток і ризику появи поверхневих дефектів. Вплив температури не завжди лінійний і залежить від типу філаменту, швидкості друку та охолодження. Також важливо, що температура на дисплеї не гарантує однакову теплопередачу у зону екструзії через стан термістора, тепловий контакт і інші фактори.

Швидкісні режими та прискорення визначають, наскільки точно механіка відпрацьовує траєкторії. При надто високих швидкостях можливе погіршення геометрії, виникнення вібрацій, зміна фактичної подачі через обмеження екструдера або прослизання філаменту. Для багатьох настільних систем характерно, що стабільний друк можливий у певному діапазоні швидкостей, а вихід за межі цього діапазону призводить до збільшення браку. У візуальному плані це може проявлятися як нестабільна товщина доріжки, нерівність заповнення або локальні дефекти поверхні на переходах траєкторій.

Ретракція є важливим параметром для зменшення витікання матеріалу під час переміщень сопла між ділянками друку. Недостатня ретракція або неправильна швидкість ретракції призводить до утворення ниток "stringing". Надмірна ретракція може спричинити інші проблеми, пов'язані з нестабільністю подачі, коливаннями тиску у хотенді та появою пропусків. Оскільки нитки можуть бути тонкими і напівпрозорими, їх виявлення є складнішим завданням для системи комп'ютерного зору. Разом із тим на практиці "stringing" є вагомим індикатором некоректно підібраних параметрів температури і ретракції.

Дефекти типу "zits" зазвичай пов'язані з локальними надлишками матеріалу, що формуються через короткочасні зміни тиску в екструдері, переходи траєкторій, особливості шва, коливання швидкості або мікроколивання температури. Вони проявляються як дрібні напливи на поверхні. Якщо камера розташована далеко або має низьку деталізацію, дефект може бути малопомітним, тому точність детекції залежить від роздільної здатності кадру, якості фокусування та освітлення.

"Spaghetti" є одним з найбільш критичних проявів відхилень, оскільки зазвичай означає втрату нормального формування виробу. Причиною може бути відрив деталі, руйнування частини геометрії, зіткнення сопла з виробом,

неправильна адгезія або інші ситуації, коли сопло екструдує пластик у повітрі. Візуально дефект має виражену структуру, що робить його зручним для автоматичного виявлення. При цьому важливо, що ранні ознаки можуть бути слабшими, наприклад локальне утворення ниток до моменту повного зриву, тому система моніторингу має реагувати на послідовні спрацювання, а не лише на одиничний кадр.

Окремо слід виділити вплив властивостей філаменту. Вологий філамент може давати бульбашки, нерівномірний вихід матеріалу, шум у доріжці та погіршення поверхні. Нерівномірність діаметра, домішки або забруднення можуть спричиняти нестабільну подачу. Такі фактори часто важко діагностувати лише за параметрами принтера, але вони проявляються у вигляді змін на поверхні виробу, що робить їх релевантними для візуального аналізу.

Вплив навколишнього середовища також є значущим. Потоки повітря, коливання температури, вібрації поверхні, на якій стоїть принтер, можуть призводити до змін умов охолодження і поведінки матеріалу. У лабораторних умовах доцільно організовувати стабільні умови, однак на практиці користувачі часто друкують у побутових приміщеннях, де умови змінюються. Це означає, що система моніторингу повинна бути здатною працювати при певній варіативності сцени.

Таким чином, причини відхилень у процесі FDM-друку мають різну природу, але спільним є те, що значна частина з них проявляється у візуальній формі. Це дозволяє сформулювати задачу як задачу детекції характерних ознак на кадрах друку та пов'язати результати виявлення з рекомендаціями щодо налаштувань, які впливають на стабільність процесу.

Відхилення у процесі друку доцільно аналізувати не лише як перелік причин, а як послідовність: умова → зміна режиму формування доріжки → візуальний прояв → наслідок. Такий причинно-наслідковий підхід важливий, оскільки практична система моніторингу здебільшого не «лікує» першопричину автоматично, а забезпечує виявлення прояву, після чого користувач або система керування приймає рішення (призупинити друк, змінити режим, повторно підготувати поверхню столу

тощо). Саме тому у роботі акцент робиться на візуальних симптомах, які можна надійно розпізнавати з камери.

Частина причин породжує дефекти, що проявляються дуже швидко. Наприклад, некоректний перший шар або погана адгезія часто приводять до відриву деталі: спочатку видно «підняті» краї або зсув контуру, потім — часткове відривання, а далі — повний зрив і перехід у хаотичне намотування нитки. Інша група причин формує дефекти поступово: забруднення сопла, деградація подачі матеріалу, термічний дисбаланс на дрібних деталях. У такому випадку перші прояви можуть бути малопомітними, але з часом приводять до втрати геометрії, порожнин, ослаблення деталі або до зупинки екструзії.

Практично важливо, що однаковий візуальний прояв може мати різні першопричини. Наприклад, «нитки» на переходах можуть виникати як через підвищену температуру, так і через невдалу ретракцію або вологий матеріал. Система моніторингу в такому випадку не намагається визначити, який саме параметр є коренем проблеми, а фіксує сам факт появи стійкої ознаки, яка корелює з втратою якості. Після цього користувач має можливість корегувати режим друку або зупинити процес, щоб уникнути подальших втрат.

З точки зору якості виробу дефекти можна інтерпретувати за впливом на функціональні властивості. Геометричні відхилення (деформації, зсуви шарів) безпосередньо змінюють розміри та посадки, що критично для технічних деталей. Дефекти адгезії між шарами знижують міцність і роблять виріб крихким у напрямку нормалі до шарів. Поверхневі артефакти (нерівності, «шрами», нитки) погіршують зовнішній вигляд і можуть вимагати додаткової постобробки. Найбільш затратними є «катастрофічні» дефекти, коли друк уже не може повернутися у штатний режим без втручання: такі випадки створюють прямі втрати матеріалу та часу.

1.3. Підходи до контролю процесу FDM-друку та виявлення відхилень

У практиці контролю FDM-друку можна виділити кілька загальних підходів, які відрізняються доступністю, складністю впровадження та інформативністю. У

побутових та навчальних умовах найчастіше застосовується періодичний візуальний контроль оператором. Цей підхід простий, але не забезпечує гарантованого виявлення проблем у часі. Оператор може не помітити ранні ознаки дефекту, особливо якщо друк відбувається довго або якщо дефект формується на ділянці, яку важко побачити. Крім того, візуальна оцінка є суб'єктивною, а рішення часто залежить від досвіду користувача.

Контроль за даними принтера може включати спостереження за показами температури, статусами завдань, а також аналіз журналів. Проте вбудовані параметри не завжди відображають реальність формування доріжок. Наприклад, у випадку часткового засмічення сопла температура може бути нормальною, але фактичний потік матеріалу зменшиться. У випадку проблем з ретракцією параметри також можуть бути коректними з точки зору команд, але фізичний процес буде давати нитки. Це означає, що контроль за параметрами без візуального спостереження має обмежену ефективність для задачі виявлення дефектів поверхні та аномальних утворень.

Сенсорні підходи, пов'язані з додатковими датчиками, можуть бути інформативними, але часто є складними для впровадження у настільних системах. Такі рішення потребують встановлення, підключення до контролера, обробки сигналів та узгодження даних з подіями друку. У магістерській роботі, орієнтованій на практичність, доцільно обирати рішення, які забезпечують достатню інформативність при мінімальних вимогах до модифікації обладнання.

Саме тому візуальний контроль з використанням камери є перспективним компромісом. Камера може бути зовнішньою або вбудованою, а дані можуть бути отримані у вигляді послідовності кадрів. Для користувача це доступний спосіб отримати "погляд" на процес, а для системи аналізу - джерело ознак дефектів. При цьому підхід не вимагає втручання у механіку принтера, а розгортання може бути виконане на звичайній обчислювальній платформі.

Загальну структуру комп'ютерно-зорового моніторингу процесу друку з використанням нейромережевої детекції наведено на рисунку 1.3.

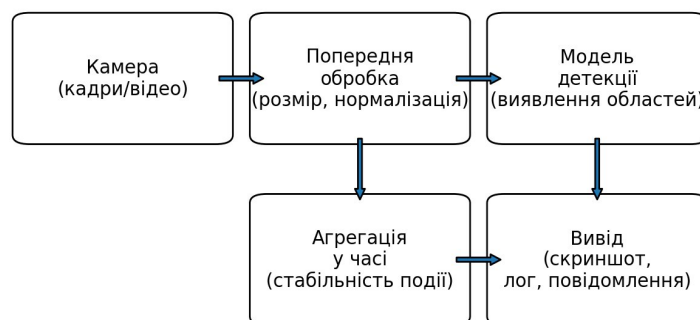


Рис. 1.3. Структура комп'ютерно-зорового моніторингу процесу друку

Алгоритмічно візуальний контроль може будуватися на класичних методах обробки зображень або на методах машинного навчання. Класичні методи часто спираються на фіксовані пороги, контури, зміни кольору або статистичні оцінки. Їх слабким місцем є чутливість до змін освітлення, відблисків і фону. Для FDM-друку це є критичним, оскільки сцена може мати блискучу платформу, рухомі елементи та зміни експозиції. Тому більш практичним для задачі загального застосування стає глибинне навчання, яке здатне узагальнювати ознаки дефекту на основі великої кількості прикладів.

У межах задач моніторингу особливу роль відіграє вибір типу моделі. Класифікація кадру як "норма" або "дефект" є простішою, але не забезпечує локалізації. Сегментація дає точні контури дефекту, але потребує більш детальної розмітки і більших ресурсів. Детекція об'єктів забезпечує локалізацію прямокутником і класифікацію типу дефекту, що є достатнім для практичного моніторингу, збереження скриншотів та ведення журналів подій. Додатково детекція дозволяє оцінювати масштаб прояву дефекту за розміром області та аналізувати динаміку у часі.

Таким чином, у даній роботі контроль процесу розглядається як поєднання візуального спостереження з автоматизованим аналізом кадрів. Результати аналізу є основою для подальшого прийняття рішень щодо налаштування режимів друку, тобто для підтримки адаптивного керування параметрами на основі фактичного стану процесу.

Існуючі підходи контролю процесу друку можна розглядати за рівнем інтеграції із принтером. Найпростіший рівень — це спостереження без керування: користувач бачить відео з камери та сам приймає рішення. Наступний рівень — автоматизоване сповіщення: система аналізує відео, виявляє відхилення і надсилає повідомлення (наприклад, на телефон). Найбільш функціональний рівень — інтеграція з керуванням, коли при підтвердженому дефекті система може призупинити друк або виконати іншу дію через доступний інтерфейс керування.

На практиці для домашніх і навчальних умов особливо цінним є саме другий рівень, оскільки він не вимагає глибокого втручання у прошивку принтера, але різко зменшує ймовірність ситуації, коли користувач не помічає проблему десятки хвилин або години. При цьому додавання керування (пауза/стоп) є логічним розширенням: багато платформ керування друком мають API або механізми віддалених команд, що дозволяє реалізувати реакцію без фізичної присутності.

Водночас візуальний контроль має природні обмеження, які необхідно враховувати у вимогах до системи. По-перше, область друку частково перекривається рухомою головою, що створює «сліпі» моменти. По-друге, умови освітлення можуть змінюватися, а колір філамента впливає на контраст і видимість дрібних ознак. По-третє, дефекти можуть бути схожими за формою (наприклад, локальні нерівності шару та тимчасове прилипання нитки), що підвищує ризик хибних спрацювань.

Саме тому обґрунтованим є поєднання детекції з правилами часової стабілізації: сповіщення формується не при одиничному спрацюванні, а при підтвердженні на серії кадрів; додатково використовується пауза між повідомленнями, щоб не «засмічувати» канал оповіщення під час одного й того ж дефекту. Така логіка є практичною і підвищує довіру користувача до системи, оскільки зменшує кількість випадкових повідомлень.

1.4. Комп'ютерний зір у задачах моніторингу адитивного виробництва

Комп'ютерний зір у контексті адитивного виробництва використовується як засіб перетворення зображень у формалізовані ознаки та рішення. Для FDM-друку камера дозволяє отримувати дані про форму доріжок, контури виробу, присутність ниток та локальні дефекти поверхні. Це створює можливість переходу від суб'єктивного спостереження до систематичного контролю з вимірюваними показниками.

При побудові візуальної системи моніторингу важливим є вибір розташування камери. Ракурс визначає, які елементи сцени потрапляють у кадр, і наскільки добре видно поверхню, на якій формуються дефекти. Надто низький кут може призводити до перекриття зони друку кареткою або соплом, а надто високий може зменшувати деталізацію дрібних дефектів. Також важливо, щоб камера була зафіксована жорстко, оскільки мікроколивання створюють додатковий шум та знижують стабільність детекції.

Умови освітлення є одним з ключових факторів. Для стабільної роботи системи доцільним є рівномірне освітлення зони друку без різких відблисків. У реальності часто зустрічаються ситуації, коли освітлення змінюється протягом доби або залежить від зовнішніх джерел. Це означає, що модель має бути навчена на даних, які включають різні умови освітлення, або ж сцена має бути стандартизована, наприклад шляхом використання стабільного світла у робочій зоні.

Якість зображення також залежить від параметрів камери. Компресія відео може створювати артефакти, які впливають на виявлення тонких структур. Автофокус і автоматична експозиція можуть змінювати картинку і створювати нестабільність ознак. З практичної точки зору, для систем моніторингу важливо мати достатню роздільну здатність і стабільні налаштування, щоб модель бачила дефект в однаковому масштабі.

Для задачі, орієнтованої на виявлення дефектів, доцільно розглядати модель як частину загального пайплайна аналізу. Кадр спочатку приводиться до визначеного розміру, виконується нормалізація, після чого модель видає координати

імовірних областей дефектів і їх класи. Далі може бути застосована логіка фільтрації за порогом впевненості та агрегації у часі. Важливим є те, що одиничне спрацювання на одному кадрі не завжди є підставою для висновку про критичне відхилення. Більш надійним є підхід, коли рішення про наявність проблеми приймається на основі серії кадрів, у яких дефект повторюється.

Комп'ютерний зір також дозволяє формувати дані для подальшого аналізу. Наприклад, можна оцінювати частоту появи дефектів певного типу при різних параметрах друку або порівнювати стабільність процесу для різних матеріалів. У контексті магістерської роботи це важливо тим, що результати детекції можуть використовуватися не лише як сповіщення, але і як об'єктивна база для оцінки ефективності налаштувань та подальшого вдосконалення процесу друку.

Таким чином, комп'ютерний зір у даній роботі розглядається як практичний інструмент аналізу процесу FDM-друку, який забезпечує можливість автоматизованого виявлення відхилень і створює передумови для формування рекомендацій з корекції параметрів друку на основі фактичного стану процесу.

У задачах моніторингу адитивного виробництва комп'ютерний зір використовується не лише для «факту дефекту», а й для пояснюваності результату. На відміну від телеметрії, яка часто відображає лише числові параметри, візуальний канал дозволяє показати користувачу, що саме система вважає проблемою: виділити прямокутником область прояву та додати текстовий підпис класу. Це важливо для практичного застосування, оскільки користувач швидко оцінює, чи є спрацювання обґрунтованим, і чи потрібно реагувати.

Реалістичний моніторинг повинен враховувати, що дефекти мають різний характер у часі. «Катастрофічні» дефекти, як-от хаотичне намотування нитки, зазвичай мають високу візуальну помітність і добре піддаються детекції. Водночас «м'які» дефекти, як-от локальна нерівномірність шару, можуть проявлятися слабше і вимагати більшої кількості даних та точнішої розмітки. Звідси випливає методичний висновок: для практичної системи доцільно починати з найбільш виражених дефектів, поступово розширюючи перелік класів і підвищуючи здатність моделі узагальнювати.

РОЗДІЛ 2

ТЕОРЕТИЧНІ ТА МЕТОДИЧНІ ОСНОВИ АНАЛІЗУ ПРОЦЕСУ ДРУКУ

2.1. Формалізація задачі виявлення відхилень у процесі FDM-друку за зображеннями

У межах даної кваліфікаційної роботи магістра задача контролю процесу FDM-друку розглядається як задача комп'ютерного зору, в якій вихідне рішення формується на основі аналізу візуальних даних із камери спостереження. Фактичний стан друку оцінюється за зображенням або послідовністю зображень, що відображають зону друку, геометрію поточного шару, характер екструзії та появу візуальних ознак, які пов'язані з відхиленнями. Такий підхід дозволяє перейти від суб'єктивного контролю оператором до відтворюваного алгоритмічного аналізу, який можна оцінювати кількісними метриками.

Вхідними даними є кадр, отриманий з камери, що фіксує робочу область 3D-принтера. Кадр являє собою цифрове зображення з роздільною здатністю, визначеною параметрами камери та режимом зйомки. У реальних умовах кадр може містити не лише друковану модель, але й елементи конструкції принтера, ділянки платформи, рухомі вузли, відблиски від поверхні, тіні та візуальні шуми. Таким чином, задача виявлення відхилень у друці є задачею аналізу сцени з великою кількістю неконтрольованих факторів, де цільові ознаки можуть бути як великими і контрастними, так і дрібними та слабо вираженими.

Вихідними даними системи є набір виявлень, кожне з яких включає тип відхилення, координати області прояву на зображенні та оцінку впевненості. У випадку застосування детекції об'єктів область прояву подається у вигляді обмежувального прямокутника, який локалізує дефект у межах кадру. Тип відхилення визначається як клас з наперед заданого набору класів. Оцінка впевненості є числовим показником, який використовується для фільтрації слабких спрацювань та для налаштування компромісу між чутливістю і кількістю хибних тривог.

Для коректної постановки задачі необхідно узгодити, що вважається "відхиленням" у контексті візуального аналізу. У цій роботі під відхиленнями розуміються характерні візуальні прояви процесу друку, що свідчать про небажаний стан екструзії або формування поверхні. При цьому важливо підкреслити, що візуальні прояви не завжди є прямим підтвердженням конкретної фізичної причини, але вони є інформативними індикаторами проблеми, які можна виявляти автоматично. З практичної точки зору системі достатньо стабільно ідентифікувати прояв і класифікувати його тип, а інтерпретація щодо параметрів друку може бути сформована на рівні рекомендацій для оператора.

Особливість моніторингу FDM-друку полягає в часовій природі процесу. Відхилення можуть виникати поступово, з наростанням ознак у часі, або проявлятися різко. Тому важливо розглядати не лише одиничний кадр, а послідовність кадрів. У методичному плані це означає, що система може працювати у двох режимах: режим покадрової детекції, де рішення приймається для кожного кадру окремо, і режим подієвого аналізу, де рішення про наявність проблеми приймається на основі повторюваності спрацювань на серії кадрів. Другий режим є більш стійким до випадкових шумів, але потребує додаткових правил агрегації результатів у часі. У межах теоретичної постановки задачі базовою одиницею аналізу є кадр, а часові механізми розглядаються як надбудова, що підвищує практичну надійність системи.

Після отримання детекцій на одному кадрі важливо перейти від «кадрової» оцінки до «процесної», тобто такої, що описує стан друку в часі. У прикладних системах моніторингу часто застосовується правило підтвердження дефекту: подія вважається істинною, якщо детекція одного й того ж класу повторюється на декількох кадрах підряд або з певною частотою протягом короткого інтервалу. Такий підхід дає дві переваги. По-перше, він зменшує вплив одиничних хибних детекцій, які можуть виникати через шум зображення або перекриття області друку. По-друге, він узгоджується з природою дефектів: більшість критичних відхилень не виникають на один кадр, а тривають певний час.

Крім підтвердження серією кадрів, практично корисним є введення «вікна спостереження» і «cooldown» (пауза між подіями). Вікно спостереження означає, що рішення приймається з урахуванням останніх N кадрів, а cooldown обмежує частоту повідомлень при одному й тому ж дефекті. Таким чином, загальна постановка задачі містить дві частини: детектор як інструмент отримання первинних ознак, та логіка агрегації у часі як інструмент підвищення надійності системи.

2.2. Вимоги до вхідних даних та формування зони інтересу

Ефективність візуального аналізу значною мірою залежить від того, наскільки стабільними є умови зйомки та наскільки інформативною є зона, що потрапляє у кадр. У випадку 3D-друку камера може бути розташована по-різному, але у будь-якому варіанті бажано, щоб у кадрі була видима робоча платформа та зона формування шару. Якщо камера фіксує занадто велику область, зростає частка нерелевантних деталей, що ускладнює навчання. Якщо камера фіксує занадто малу область, може бути втрачено контекст, наприклад край моделі або характерні ознаки ниткоподібних утворень.

Методично доцільно вводити поняття зони інтересу, тобто області кадру, у межах якої відбувається основний аналіз. Зона інтересу може визначатися як фіксована рамка кадру, що містить платформу і друковану модель. У разі стабільної установки камери така зона може бути однаковою для всіх кадрів. Це дозволяє зменшити вплив фону і пришвидшити обробку, оскільки модель отримує більш концентровані дані. Разом з тим, у загальному випадку зона інтересу може змінюватися залежно від того, як змінюється масштаб об'єкта у кадрі, але у настільних умовах найчастіше достатньо фіксованої зони.

Ще одним аспектом вхідних даних є роздільна здатність та деталізація. Дрібні дефекти поверхні сприймаються камерою лише тоді, коли вони займають достатню кількість пікселів. Якщо дефект є надто малим у кадрі, модель фактично не має інформації, щоб відрізнити його від шуму або текстури. Це означає, що для цілей

детекції дрібних відхилень важливо, щоб камера забезпечувала достатній масштаб зйомки, а також щоб зображення не втрачало дрібні деталі через сильну компресію.

Освітлення є фактором, що впливає на контрастність, наявність відблисків і стабільність кольорових ознак. У робочих умовах бажаним є рівномірне освітлення, за якого зменшується кількість різких тіней і відблисків. У теоретичній частині це важливо тому, що будь-які неконтрольовані зміни освітлення створюють домішкові ознаки, які модель може помилково вважати корисними. Для зменшення цього ефекту під час підготовки даних доцільно включати приклади з різними умовами освітлення або виконувати нормалізацію яскравості та контрасту в рамках попередньої обробки.

Для практичної реалізації моніторингу важливим є підбір параметрів зйомки, які забезпечують стабільність ознак. До таких параметрів належать кут огляду, дистанція камери, частота кадрів та спосіб стиснення відео. Надто широкий кут огляду зменшує деталізацію області друку, тоді як надто близький план може втрачати контекст і збільшувати вплив перекриттів рухомою головою. Компромісом є ракурс, при якому видно ключову область друку та прилеглу частину столу, де можуть проявлятися ранні ознаки відриву.

Використання ROI як методичного прийому також полегшує навчання моделі, оскільки зменшує різноманітність фону. Це особливо важливо у випадках, коли датасет формувався з різних джерел: різні принтери, різні столи, різні кольори корпусу. Обмеження аналізу зоною друку приводить до того, що модель більше «зосереджується» на релевантних ознаках і менше реагує на зовнішні елементи.

2.3. Попередня обробка зображень та приведення до узгодженого формату

Нейромережеві моделі детекції, які застосовуються на практиці, зазвичай працюють з фіксованим розміром вхідних даних. Це означає, що зображення різних розмірів необхідно привести до стандартного формату. Найпоширеніший спосіб полягає у масштабуванні зображення зі збереженням пропорцій та заповненні вільних областей фоновими пікселями. Такий підхід дозволяє уникнути спотворення

геометрії об'єктів, оскільки пряме розтягування кадру до квадратного розміру змінює співвідношення сторін і може погіршити здатність моделі узагальнювати ознаки. Приклад приведення кадру до стандартного розміру методом letterbox наведено на рисунку 2.1.

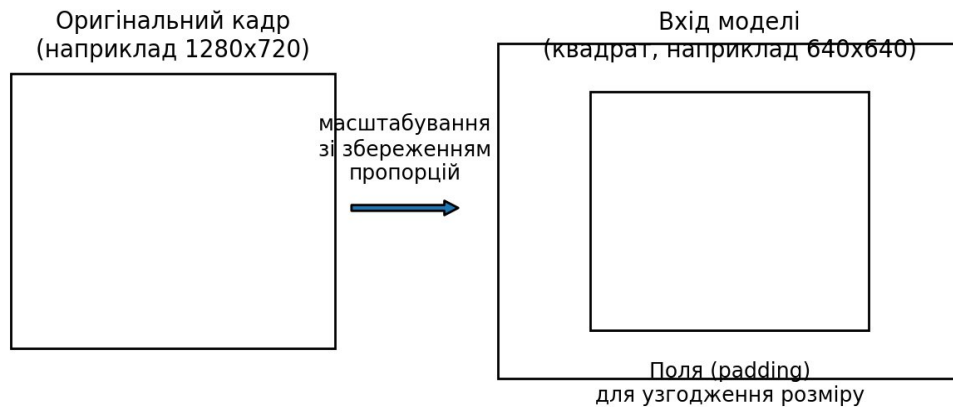


Рис. 2.1. Приведення кадру до стандартного розміру методом letterbox

У контексті задачі детекції дефектів FDM-друку важливо розуміти, що приведення зображення до фіксованого розміру впливає на видимість дрібних деталей. Якщо початковий кадр має високу роздільну здатність, а вхідний розмір моделі є значно меншим, дрібні дефекти можуть бути "згладжені" під час масштабування. Тому вибір розміру вхідного зображення є методичним компромісом між швидкодією та деталізацією. При цьому корисно враховувати, що для дрібних дефектів вигреш у точності часто досягається за рахунок більшого вхідного розміру, але це підвищує час обробки і вимоги до обчислювальних ресурсів.

Окрім масштабування, попередня обробка може включати перетворення кольорового простору та нормалізацію. Для багатьох моделей прийнято подавати пікселі у стандартизованому вигляді, де значення каналів нормалізовані у певному діапазоні. Така операція підвищує стабільність навчання і зменшує залежність від абсолютних значень яскравості. Для сцени друку це може бути важливо у випадках,

коли камера автоматично змінює експозицію, і без нормалізації одна і та сама сцена може мати різні середні значення яскравості.

Важливим методичним питанням є збереження відповідності між перетвореннями зображення та розміткою. Якщо зображення масштабується і доповнюється фоном, координати обмежувальних прямокутників у розмітці також мають бути коректно перераховані. У практичній реалізації це зазвичай виконується автоматично у процесі навчання, оскільки сучасні фреймворки детекції застосовують однакові перетворення до зображення і міток. Проте на рівні методики важливо розуміти, що саме від правильності цього узгодження залежить коректність навчання і достовірність метрик.

Операція letterbox є критичною для узгодження даних, оскільки при навчанні та інференсі модель очікує однаковий формат входу. Методично важливо, що letterbox виконує масштабування зі збереженням пропорцій, тобто не «розтягує» зображення, а отже не змінює відносні форми проявів дефектів. Це знижує ризик того, що модель навчиться на спотворених ознаках, які не відповідають реальному вигляду дефекту.

Окрім узгодження розміру, у задачах реального моніторингу доцільно враховувати стабільність освітлення. Навіть при правильному налаштуванні камери світло може змінюватися протягом дня або через відблиски. Тому під час підготовки даних бажано мати приклади як у «нормальному» освітленні, так і в умовах зниженого контрасту, щоб модель не була надто чутливою до одного сценарію.

2.4. Формування датасету та принципи збирання даних з різних ресурсів

Задачі комп'ютерного зору є залежними від даних, тому якість і різноманітність датасету визначає практичну придатність моделі. Для задачі виявлення відхилень у процесі FDM-друку характерною є відсутність одного універсального датасету, який би охоплював усі типи принтерів, камер, матеріалів і умов освітлення. Тому на практиці датасет формується шляхом збирання прикладів з різних доступних джерел. У межах цієї роботи дані розглядаються як сукупність

зображень, отриманих з різних ресурсів, які потім приводяться до узгодженого формату та розмічаються за узгодженими правилами. Методичну послідовність підготовки даних та навчання моделі детекції, використану в роботі, наведено на рисунку 2.2.

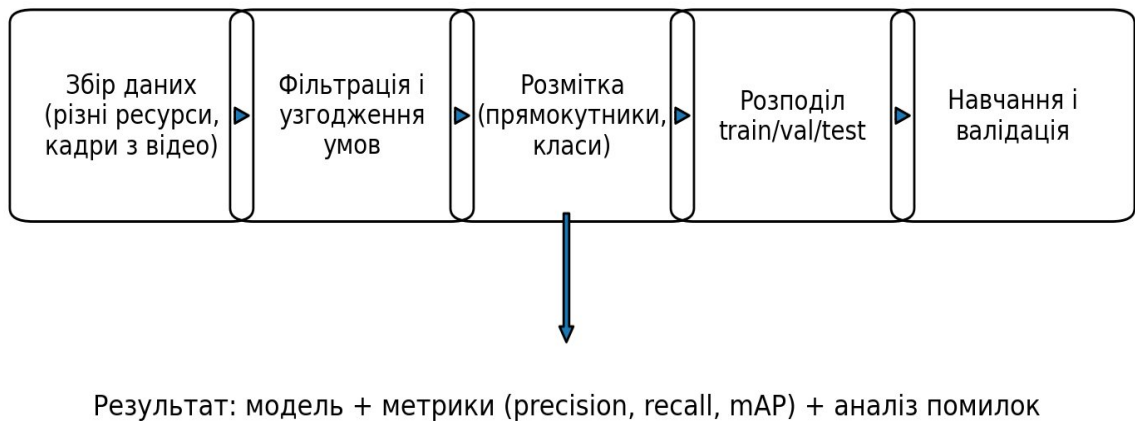


Рис. 2.2. Методичний конвеєр підготовки даних та навчання моделі

Збирання даних з різних ресурсів має як переваги, так і ризики. Перевага полягає у різноманітності сцен, що підвищує узагальнювальну здатність моделі і зменшує ризик того, що модель "пристосується" до однієї камери або одного типу друку. Ризик полягає у тому, що дані можуть бути неоднорідними за якістю, масштабом, компресією та за тим, як саме проявляються дефекти. Якщо не виконати узгодження, модель може навчитися на випадкових ознаках, наприклад на особливостях кольоропередачі або фону, а не на суті дефекту. Тому методика формування датасету має включати етап аналізу якості, фільтрації непридатних прикладів та приведення до більш однорідних умов.

Ще одним важливим питанням є баланс класів. На практиці деякі типи дефектів зустрічаються частіше, або їх легше знайти у відкритих джерелах, тому датасет може бути незбалансованим. У задачах детекції це проявляється в тому, що модель може краще навчитися на класі, який представлений великою кількістю прикладів, і гірше на класі з малою кількістю прикладів. У методичній частині важливо закласти механізми, які зменшують негативний вплив дисбалансу,

наприклад шляхом розширення менш представлених класів, застосування аугментацій або коректного налаштування процесу навчання.

Дані для детекції можуть бути отримані як зі статичних зображень, так і з відео. У випадку відео доцільно виконувати вибір кадрів з певним кроком, щоб зменшити надмірну схожість сусідніх кадрів. Якщо взяти багато послідовних кадрів з одного відео, модель може отримати занадто багато майже однакових прикладів, і це погіршує узагальнення. Тому методично доцільно відбирати кадри таким чином, щоб вони представляли різні стадії процесу або різні ситуації дефекту. Окремо важливо уникати ситуації, коли майже однакові кадри потрапляють і в навчальну, і в тестову вибірку, оскільки це завищує оцінку метрик і не відображає реальну здатність моделі працювати на нових даних.

Окремою проблемою при формуванні датасету є дисбаланс класів: у реальному друці більшість часу процес іде штатно, а дефекти є відносно рідкісними. Якщо переносити цей дисбаланс у навчальні дані без корекції, модель може «звикнути» переважно бачити норму і гірше розпізнавати дефекти. Методично доцільно забезпечити достатню кількість прикладів дефектних ситуацій та різних проявів одного й того ж дефекту. При цьому важливо не «перекосити» дані в інший бік, коли дефекти представлені надмірно однотипно (наприклад, лише один друк або одна деталь), адже тоді модель запам'ятовує сцену, а не ознаку.

Також доцільно враховувати різні масштаби проявів: інколи дефект займає малу частину кадру (локальний підрив або невелике намотування нитки), а інколи — значну частину (масивне «спагеті»). Наявність прикладів різного масштабу вчить модель бути стійкою до віддалення камери та до різних розмірів друкованих моделей.

2.5. Розмітка даних для задачі детекції та узгодження правил анотації

Розмітка є ключовим етапом підготовки датасету, оскільки саме вона визначає, що модель вважає "правильним" під час навчання. Для детекції розмітка зазвичай полягає у нанесенні обмежувальних прямокутників на області, де проявляється

дефект, та у присвоєнні цим областям класів. У задачі FDM-друку важливо, щоб правила розмітки були достатньо однозначними, інакше різні приклади одного і того ж дефекту будуть розмічені по-різному, що ускладнює навчання.

Одне з методичних питань полягає у визначенні меж дефекту. Наприклад, ниткоподібні утворення можуть бути довгими і тонкими, а "клубок" пластику може мати складну форму. Прямокутник є спрощеною формою локалізації, тому необхідно визначити, чи покриває прямокутник увесь дефект, чи лише його найбільш інформативну частину. У практиці детекції найчастіше прагнуть охопити дефект цілком, але з певним допуском, оскільки надто точне обведення не завжди можливе і не завжди корисне. При цьому важливо, щоб прямокутники не включали надто багато фону, оскільки це знижує точність локалізації та ускладнює навчання.

Друге методичне питання полягає у відмежуванні дефекту від нормальних елементів друку. Наприклад, тонкі лінії можуть виникати не лише як дефект, але й як частина геометрії моделі або як нормальні "перемички" в процесі формування елементів. Якщо розмітка не враховує ці ситуації, модель може отримати суперечливі сигнали і почати помилково позначати нормальні елементи як дефекти. Тому під час розмітки необхідно спиратися на контекст кадру, а також на знання про типові прояви конкретного відхилення.

Третє питання полягає у маркуванні дрібних дефектів. Дефекти поверхні, які мають малий розмір, складно розмічати на зображеннях низької якості. Якщо дефект займає кілька пікселів, прямокутник стає майже точкою, і модель може не мати достатньої інформації для навчання. У таких випадках методично доцільно відбирати лише ті приклади, де дефект справді видимий, або використовувати зображення з більшою деталізацією. Також важливо узгодити, чи розмічається кожен дрібний дефект окремо, чи вони розглядаються як груповий прояв у межах певної області. Для задачі, орієнтованої на моніторинг, груповий підхід може бути більш практичним, оскільки оператору важливо знати факт прояву проблеми, а не точну кількість дрібних точок.

Розмітка має бути узгоджена з форматами, які використовуються у навчанні. Формат YOLO передбачає, що для кожного зображення існує файл з координатами

прямокутників у нормалізованому вигляді відносно ширини і висоти кадру. Це зручно тим, що розмітка не залежить від абсолютних розмірів, але вимагає коректного перерахунку при будь-яких змінах роздільної здатності. У методичному плані це означає, що зображення і мітки мають завжди розглядатися як пара, а будь-які перетворення мають бути узгодженими.

Якість розмітки безпосередньо визначає верхню межу якості моделі. Якщо координати прямокутників неточні або класи вибрані суперечливо, модель отримує неоднозначний навчальний сигнал. Для задачі моніторингу важливо, щоб bounding box охоплював саме прояв дефекту, а не надто великий фон. Надмірно великі прямокутники «розмивають» ознаки, а надто малі можуть не включати ключові елементи прояву.

У випадку відеоданих додатковою складністю є те, що прояв дефекту може змінюватися від кадру до кадру: зростати або переміщуватися. Методично це означає, що розмітку краще формувати на представницьких кадрах, де прояв дефекту є чітким, а для підтвердження у часі використовувати вже логіку системи (серії кадрів). Таким чином, навіть якщо окремий кадр має граничний прояв, рішення про подію приймається на рівні часової агрегації.

2.6. Аугментації та підвищення узагальнюваності моделі

У задачах комп'ютерного зору аугментації використовуються як спосіб штучно розширити різноманітність даних без збирання нових прикладів. Для моніторингу процесу друку це особливо актуально, оскільки умови освітлення, ракурс і якість камери можуть змінюватися, а датасет, зібраний з обмеженої кількості джерел, може не охоплювати всі варіанти. Аугментації дозволяють моделі навчитися бути стійкою до таких змін і зменшують ризик перенавчання на конкретні умови.

Аугментації, які є корисними для сцени друку, включають зміни яскравості, контрасту, насиченості, додавання шуму, легке розмиття, а також геометричні перетворення. Зміни яскравості і контрасту імітують різні умови освітлення, що

корисно при роботі з камерами, які автоматично змінюють експозицію. Додавання шуму та розмиття імітує якість зображення при слабкому освітленні або при компресії відео. Геометричні перетворення, такі як невеликі повороти або зсуви, імітують невеликі зміни позиції камери або вібрації.

Разом з тим, методика аугментацій має враховувати, що занадто сильні перетворення можуть створити нереалістичні приклади. Наприклад, сильне розмиття може "стерти" дрібні дефекти, а надто великі повороти можуть змінити контекст сцени. Тому аугментації слід застосовувати помірно, зберігаючи реалістичність. У задачі детекції також важливо, щоб геометричні перетворення коректно застосовувалися не лише до зображення, але й до координат прямокутників у розмітці.

Для даної задачі корисним є також підхід, коли частина аугментацій застосовується випадково під час навчання, а не створюється як окремі збережені файли. Це дозволяє кожного разу показувати моделі дещо інший варіант кадру і таким чином розширювати ефективний обсяг даних. У методичній частині важливо зазначити, що аугментації не замінюють реальні дані, але дозволяють отримати більшу стійкість моделі при обмеженому датасеті.

Аугментації слід підбирати так, щоб вони імітували реальні варіації сцени, а не створювали штучні спотворення. Для моніторингу друку корисними є варіації яскравості, контрасту, невеликі повороти та масштабування, а також легкий шум, який наближує дані до реального відео з компресією. Натомість агресивні перетворення (велике обертання, сильне розтягування, надмірне блюрення) можуть зробити дефект «неприродним» і знизити здатність моделі узагальнювати.

Ще один методичний аспект — перевірка того, чи не змінюють аугментації зміст розмітки. Будь-яке геометричне перетворення повинно узгоджено застосовуватися як до зображення, так і до координат bounding box. У сучасних фреймворках ці операції автоматизовані, однак контроль навчальних батчів (візуалізація розмітки) залишається важливою процедурою якості.

2.7. Обґрунтування вибору підходу детекції об'єктів

Задача візуального контролю друку може бути сформульована по-різному. Один варіант полягає у класифікації кадру як "нормальний" або "аномальний" або у багатокласовій класифікації, де кадр відноситься до певного стану. Такий підхід простий з точки зору розмітки, оскільки достатньо позначити кадр загальним класом. Проте він не дає локалізації дефекту, а також може бути нестійким, якщо дефект займає невелику частину кадру. У такому випадку модель може не "побачити" дефект, оскільки глобальні ознаки кадру домінують [3].

Сегментація є більш точним підходом, оскільки дозволяє виділити форму дефекту піксель за пікселем. Це особливо корисно для тонких структур або для складних форм. Однак сегментація вимагає значно більш трудомісткої розмітки, оскільки потрібно обводити контур дефекту або формувати маску. Для задачі магістерської роботи, орієнтованої на практичне впровадження, сегментація може бути надмірно складною на етапі підготовки даних, особливо якщо дефекти різноманітні і дрібні [12].

Детекція об'єктів є компромісним підходом, який поєднує можливість визначити тип відхилення і локалізувати його приблизну область. Для моніторингу процесу друку цього зазвичай достатньо, оскільки оператору важливо розуміти, де саме виникла проблема, а точна форма дефекту у багатьох випадках не є критичною. Також детекція добре інтегрується у практичні сценарії, де потрібно зберігати скріншоти з виділеними областями, вести журнал спрацювань і аналізувати частоту відхилень.

З огляду на це у даній роботі обрано підхід детекції об'єктів, що дозволяє отримати структурований вихід у вигляді набору виявлень з координатами і класами [3, 4]. Далі в розділі розглядаються принципи роботи нейромережових детекторів, процес навчання та метрики оцінки, які дозволяють кількісно оцінити якість такого підходу.

2.8. Загальні принципи нейромережевої детекції та особливості підходу YOLO

Нейромережеві детектори об'єктів формують прогноз шляхом перетворення зображення у набір ознак, які потім використовуються для регресії координат прямокутників та класифікації цих прямокутників за класами. На початкових шарах мережа виявляє базові ознаки, такі як контури та текстури. На глибших шарах формуються більш абстрактні ознаки, які відповідають формам і структурам. Для задачі дефектів друку це означає, що модель може виявляти як грубі аномальні структури, так і дрібні патерни, якщо вони достатньо представлені у даних.

Підхід YOLO відноситься до однопрохідних детекторів, які формують прогноз за один прямий прохід мережі. Практична перевага полягає у швидкодії, що важливо для моніторингу. При цьому сучасні реалізації YOLO використовують багатомасштабні ознаки, що дозволяє працювати з об'єктами різного розміру. Для дефектів друку це критично, оскільки різні типи відхилень мають різний масштаб у кадрі. Великі ниткоподібні утворення можуть займати значну частину кадру, а дрібні дефекти поверхні можуть бути дуже локальними.

Методично важливо розуміти, що швидкодія YOLO досягається за рахунок певної структури прогнозу, де мережа оцінює можливі об'єкти на регулярній сітці ознак. Потім застосовується механізм відбору найкращих виявлень. У результаті система видає множину прямокутників з оцінками впевненості. Оскільки мережа може видати багато кандидатів, на практиці застосовується процедура придушення дублюючих виявлень, яка залишає найбільш впевнені прямокутники і прибирає ті, що сильно перекриваються та описують один і той самий дефект.

Вибір порога впевненості є важливим для практичного застосування. Якщо поріг низький, модель буде показувати багато кандидатів, серед яких буде більше хибних. Якщо поріг високий, модель може пропускати дефекти, особливо складні або дрібні. Методично правильним є вибір порога на основі метрик на валідаційній вибірці, а також з урахуванням того, яка помилка є більш критичною для задачі моніторингу. У багатьох прикладних системах моніторингу більш небажаним є пропуск дефекту, ніж одиничне хибне спрацювання, але це залежить від умов

використання. У даній роботі поріг розглядається як налаштування, що вибирається на основі аналізу кривих метрик і практичних тестів на кадрах.

Підхід YOLO є практичним для задач, де важлива швидкодія, оскільки детекція виконується за один прохід нейромережі без окремого етапу формування пропозицій областей. Архітектурно сучасні модифікації містять три логічні частини: блок виділення ознак (backbone), блок об'єднання ознак різних масштабів (neck) та вихідну голову детекції (head). Таке компонування дозволяє моделі працювати з проявами різного розміру: від дрібних локальних артефактів до великих дефектів, що займають значну частину кадру.

Для моніторингу друку важливо, що дефекти можуть бути як тонкими і малоконтрастними, так і різко вираженими. Тому здатність моделі працювати з багатомасштабними ознаками є ключовою. Крім того, практична цінність YOLO полягає у доступному інструментарії навчання та валідації, який дозволяє швидко проводити повторні експерименти після доповнення датасету, а також отримувати стандартні метрики та візуалізації.

2.9. Навчання моделі детекції та постановка оптимізаційної задачі та компоненти втрат

Навчання нейромережевої моделі детекції можна розглядати як процес оптимізації параметрів мережі, при якому модель наближає свою функцію прогнозу до еталонної розмітки. Для кожного зображення модель формує набір прогнозованих прямокутників і класів, а еталонна розмітка задає, де на зображенні є дефекти і які вони. Функція втрат вимірює, наскільки відрізняються прогноз і еталон, і навчання полягає у мінімізації цієї функції за рахунок зміни параметрів мережі.

У задачі детекції функція втрат зазвичай складається з кількох компонентів. Одна компонента відповідає за точність локалізації, тобто за те, наскільки добре прямокутник прогнозу збігається з еталонним прямокутником. Інша компонента відповідає за класифікацію, тобто за правильність визначення класу дефекту. Також

існує компонента, яка відповідає за оцінку впевненості або за відокремлення об'єктів від фону. У методичному плані важливо, що ці компоненти можуть по-різному впливати на навчання, особливо у випадку дисбалансу класів або у випадку дрібних об'єктів.

Для локалізації часто використовуються метрики перекриття, які враховують площу перетину і об'єднання прямокутників. Чим більше перекриття, тим краща локалізація. У випадку дрібних дефектів навіть невелике зміщення прямокутника може суттєво зменшити перекриття, тому навчання дрібних об'єктів є більш чутливим до якості розмітки і до масштабу зображення. Це пояснює, чому для дефектів типу "zits" якість може бути нижчою, якщо камера не дає достатньої деталізації або якщо розмітка має невизначеність.

Класифікаційна компонента визначає, чи модель правильно відрізняє типи дефектів. У задачах з кількома класами модель повинна навчитися розрізняти схожі візуальні ознаки. Наприклад, тонкі нитки можуть бути схожими на елементи моделі, а локальні дефекти поверхні можуть бути схожими на шум або текстуру. Тому методика підготовки даних повинна включати різноманітні приклади і забезпечувати узгоджені правила розмітки, щоб класи були відокремлені не лише словами, а й візуальним змістом.

Налаштування навчання детектора є окремим методичним завданням. На результат впливають: кількість епох, batch size, швидкість навчання, параметри регуляризації, а також стратегія донавчання від попередньо навчених ваг. Використання transfer learning є доцільним, оскільки базова модель вже містить універсальні ознаки для зображень, а у межах конкретної задачі виконується їх адаптація до проявів дефектів друку.

Під час навчання важливо контролювати не лише кінцеві метрики, а й поведінку кривих втрат: якщо втрати на тренуванні знижуються, а на валідації погіршуються, це може бути ознакою перенавчання або проблем у розподілі даних. Також суттєвим є контроль дисбалансу класів і якості розмітки: навіть правильні гіперпараметри не компенсують систематичних помилок у даних.

2.10. Метрики оцінки якості та принципи інтерпретації результатів

Оцінка якості детектора виконується на валідаційній або тестовій вибірці, яка не використовується для навчання. Це дозволяє оцінити, наскільки модель узагальнює і чи здатна вона працювати на нових даних. Для детекції застосовуються метрики, що враховують і локалізацію, і класифікацію.

Показник *precision* відображає частку правильних спрацювань серед усіх спрацювань моделі. Якщо модель часто робить хибні спрацювання, *precision* знижується. Показник *recall* відображає частку виявлених еталонних дефектів серед усіх дефектів, що присутні у розмітці. Якщо модель пропускає дефекти, *recall* знижується. У практичному моніторингу ці показники пов'язані з порогом впевненості: при зниженні порога модель видає більше спрацювань, і *recall* може зростати, але *precision* часто знижується. При підвищенні порога ситуація може бути протилежною.

Для детекції також застосовується показник *mAP*, який є узагальненням якості по різних порогах впевненості. Він розраховується на основі кривої, що відображає залежність *precision* від *recall*. У загальному сенсі *mAP* є інтегральною оцінкою здатності моделі одночасно знаходити дефекти і не давати багато хибних спрацювань. Оскільки якість локалізації також є важливою, *mAP* часто обчислюють з урахуванням порога перекриття прямокутників. Один варіант використовує поріг перекриття 0.5, що є відносно "м'яким" критерієм локалізації. Інший варіант використовує діапазон порогів, що робить критерій більш строгим і краще відображає точність локалізації.

Матриця помилок у задачі детекції використовується для аналізу того, які класи модель плутає між собою і як часто вона відносить дефекти до фону або навпаки. Для дефектів друку матриця помилок дозволяє побачити, чи модель схильна пропускати дрібні дефекти, чи вона плутає ниткоподібні утворення з іншими структурами. Це корисно не лише для оцінки, але й для подальшого покращення датасету, оскільки помилки часто вказують на недостатню представленість певних сценаріїв у даних.

Важливим практичним питанням є вибір робочого порога впевненості. Якщо система використовується як засіб підказки оператору, доцільно обирати поріг, який забезпечує прийнятний компроміс між кількістю хибних тривог і кількістю пропусків. Методично поріг можна вибирати на основі кривої F1, де F1 є гармонічним середнім precision і recall. Максимум F1 часто відповідає збалансованому режиму. Проте у реальних умовах може бути доцільним зміщення порога в той чи інший бік залежно від того, що є пріоритетним.

Метрики precision, recall та mAP є необхідними для кількісної оцінки, однак у задачах моніторингу друку їх слід трактувати через призму практичного використання. Високий precision означає, що більшість спрацювань є коректними, тобто система рідше «турбує» користувача хибними повідомленнями. Високий recall означає, що система рідше пропускає реальні дефекти, тобто краще виконує роль раннього попередження. У практичному застосунку часто потрібен баланс: надто суворий режим зменшує хибні спрацювання, але може пропускати початкові стадії дефекту; надто чутливий режим дає більше повідомлень.

Тому доцільно поєднувати метрики з якісним аналізом прикладів детекції на відео або тестових кадрах, а також із часовими правилами прийняття рішення. Якщо система підтверджує дефект серією кадрів, то одиничні хибні детекції не перетворюються на подію, а отже практична «якість системи» може бути вищою, ніж це здається з одного лише кадру. Саме тому методика оцінювання в роботі включає не тільки числові показники, а й аналіз прикладів детекції.

РОЗДІЛ 3

ПРАКТИЧНА ЧАСТИНА

3.1. Мета та постановка експериментальних досліджень

Метою практичної частини є експериментальна перевірка працездатності комп'ютерно-зорового підходу для виявлення відхилень у процесі FDM-друку на основі аналізу зображень з камери. На відміну від задачі загальної оцінки "добре/погано", у межах цієї роботи розглядається детекція, тобто пошук та локалізація проявів відхилення на кадрі з формуванням обмежувальних рамок і оцінкою впевненості. Такий підхід краще відповідає прикладній логіці моніторингу, оскільки дає оператору можливість бачити не тільки факт проблеми, а й місце її прояву в зоні друку.

Експериментальні дослідження організовано як два окремі цикли навчання моделі детекції на підготовленому наборі даних. Кожний цикл завершується валідацією та аналізом результатів за стандартними метриками детекції, а також за допомогою якісного аналізу прикладів детекції на валідаційній вибірці. Порівняння двох навчань у межах однієї тематики є необхідним, оскільки дозволяє не просто продемонструвати один результат, а показати вплив тривалості навчання та параметрів запуску на підсумкові метрики й характер помилок моделі.

Основним практичним критерієм корисності системи є здатність стабільно виявляти відхилення без надмірної кількості хибних спрацювань при зміні умов зйомки. Оскільки у реальному середовищі можливі відблиски, зміни освітлення і фонова динаміка, у розділі додатково розглядається принцип підвищення надійності через часову фільтрацію детекцій у відеопотоці, коли одиничні спрацювання не вважаються достатніми для прийняття рішення.

3.2. Сценарії використання системи та вимоги до практичного прототипу

Комп'ютерно-зоровий контроль процесу друку має сенс лише тоді, коли система відповідає практичним сценаріям. У межах роботи розглядаються два базові режими застосування: автономний режим перевірки папки з кадрами та режим аналізу послідовності кадрів з камери або відеопотоку. Перший режим є лабораторним і потрібен для перевірки моделі, порівняння різних навчань та швидкої діагностики помилок. Другий режим відповідає реальній експлуатації, де важлива стабільність і повторюваність спрацювань.

До прототипу сформульовано вимоги, що випливають з природи задачі. По-перше, система має бути стійкою до одиничних помилок сприйняття, тобто не реагувати на випадкові спотворення кадру. По-друге, система повинна дозволяти оператору отримати візуальне підтвердження, зокрема кадр із накладеними рамками та підписами класів. По-третє, система має працювати у відносно контрольованому середовищі з повторюваними умовами, тому в межах роботи приймається вимога до зйомки у добре освітленому приміщенні з по можливості стабільним розташуванням камери відносно принтера.

Окремо враховано, що метою цього проєкту є саме виявлення відхилень, а не автоматична зупинка друку чи інші керуючі впливи на обладнання. Тому всі практичні результати інтерпретуються як інформаційна підтримка оператора, а не як повністю автономна система керування.

3.3. Формування набору даних та структура проєкту для навчання

Для навчання детектора використано набір зображень, сформований з різних доступних джерел. Принцип формування полягав у тому, щоб забезпечити достатню різноманітність сцен, зберігаючи при цьому відносну однорідність умов спостереження. Це важливо для уникнення ситуації, коли модель "вчиться" не на ознаках дефекту, а на сторонніх ознаках, пов'язаних з освітленням, фоном або конкретним типом обладнання.

Структура набору даних відповідає типовому формату Ultralytics YOLO, де окремо зберігаються зображення та файли розмітки. Для кожного зображення формується файл розмітки у форматі YOLO з координатами обмежувальних рамок у нормалізованому вигляді відносно ширини та висоти кадру. Така нормалізація забезпечує коректність розмітки для зображень різних розмірів та спрощує перенесення датасету між середовищами.

Важливою складовою датасету є баланс класів. У задачах детекції дисбаланс може приводити до того, що модель добре виявляє домінуючий клас, але пропускає рідкісні або складні класи. Щоб продемонструвати структуру даних, у розділі використовуються зведені ілюстрації розподілу інстансів за класами для кожного навчання.

На рисунку 3.1 наведено розподіл інстансів за класами для навчання №1, що дозволяє оцінити дисбаланс і потенційні наслідки для узагальнення моделі.

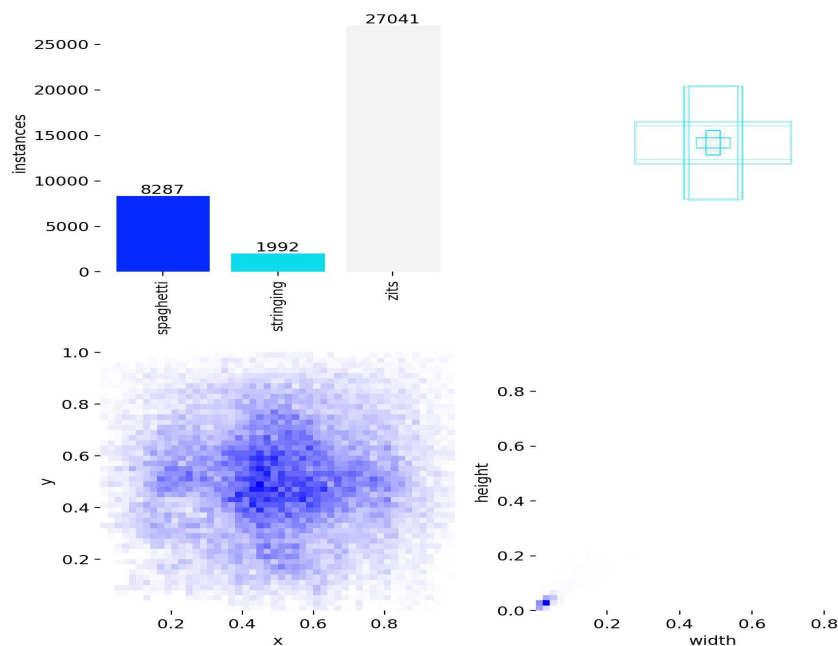


Рис. 3.1. Розподіл інстансів за класами у наборі даних для навчання №1

На рисунку 3.2 наведено аналогічний розподіл інстансів за класами для навчання №2. Порівняння цих ілюстрацій є важливим, оскільки різний склад даних або різний баланс може пояснювати відмінності метрик між навчаннями.

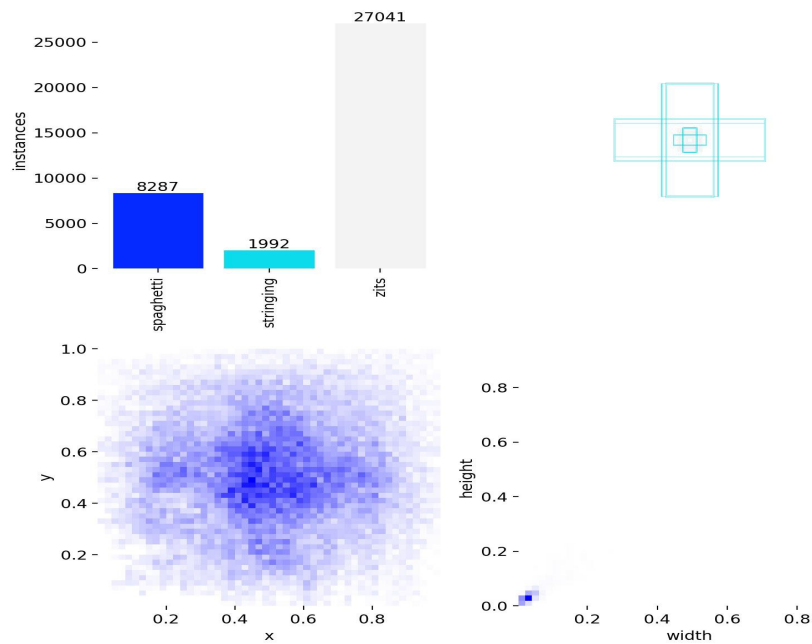


Рис. 3.2. Розподіл інстансів за класами у наборі даних для навчання №2

3.4. Попередня обробка кадрів та узгодження розміру входу детектора

Детектори класу YOLO працюють з фіксованим розміром входу, тому практично необхідно привести кадри до узгодженого формату. Для цієї задачі актуально, що початкові кадри можуть мати неквадратне співвідношення сторін, наприклад 384x640 або інші розміри залежно від камери. Пряме масштабування до квадрату без збереження пропорцій може спотворювати геометрію ознак дефектів, що знижує якість навчання та збільшує ризик хибних спрацювань.

З цієї причини застосовується приведення до квадратного розміру шляхом масштабування із збереженням пропорцій та заповнення вільних областей фоновим кольором. Такий підхід є стандартним для YOLO і відомий як letterbox. Він дозволяє зберегти пропорції сцени, а також забезпечити стабільність ознак при подачі на вхід мережі.

3.5. Налаштування навчання та параметри експериментів

Навчання виконано за допомогою інструментарію Ultralytics YOLOv8 у режимі detect. Як базу використано попередньо навчені ваги, що дозволяє

застосувати перенавчання під конкретну задачу і суттєво зменшити вимоги до обсягу даних у порівнянні з навчанням "з нуля". Під час запусків фіксувались параметри, такі як кількість епох, розмір batch, розмір входу `imgsz`, а також обраний пристрій виконання. Окремо важливо зазначити, що навчання виконувалось у режимі CPU, тому тривалість експериментів суттєво залежала від кількості епох.

3.6. Результати навчання №1

Перше навчання використовується як базова точка відліку. На цьому етапі зазвичай фіксуються типові проблеми: нестабільність метрик у перших епохах, відставання складних класів, вплив дисбалансу, а також характерні помилки локалізації. У рамках аналізу першого навчання головною метою є визначення того, чи здатна модель в принципі виділяти ознаки дефектів у кадрі та чи немає явних помилок у розмітці або структурі датасету.

Динаміка функцій втрат та метрик якості для навчання №1 наведена на рисунку 3.3. За цими кривими оцінюється, чи відбувається збіжність, а також чи є ознаки перенавчання. Якщо метрики на валідації зростають і стабілізуються, це свідчить, що модель навчається на релевантних ознаках і узагальнює на нові кадри.

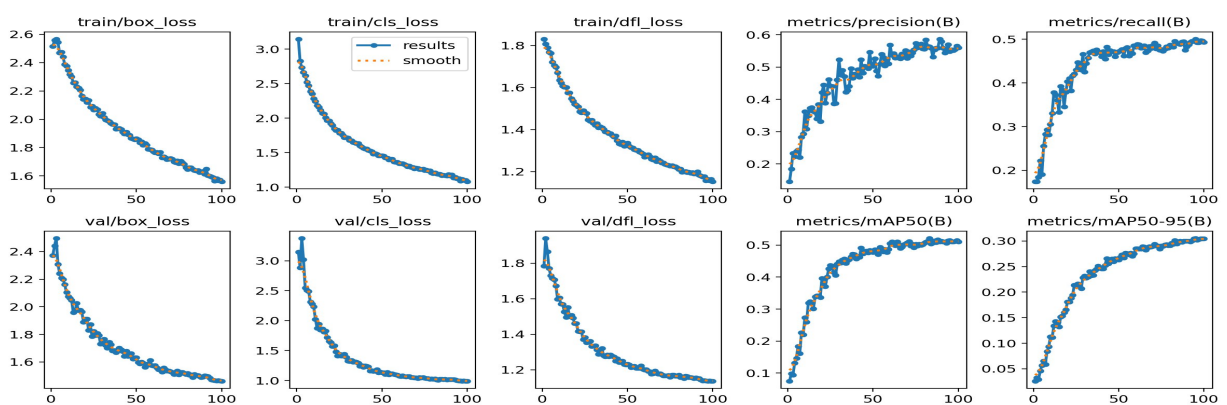


Рис. 3.3. Криві втрат і метрик якості під час навчання №1

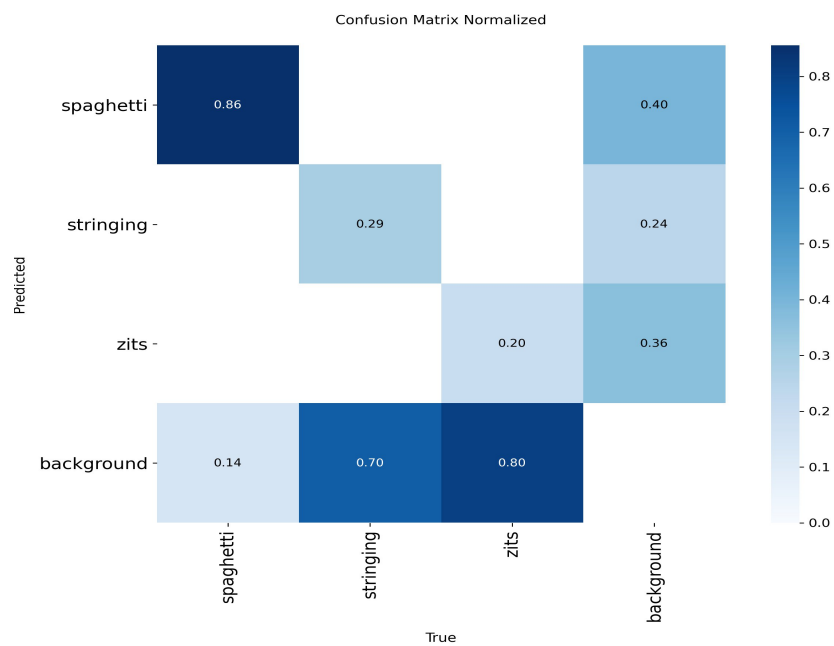


Рис. 3.4. Нормалізована матриця плутанини для навчання №1

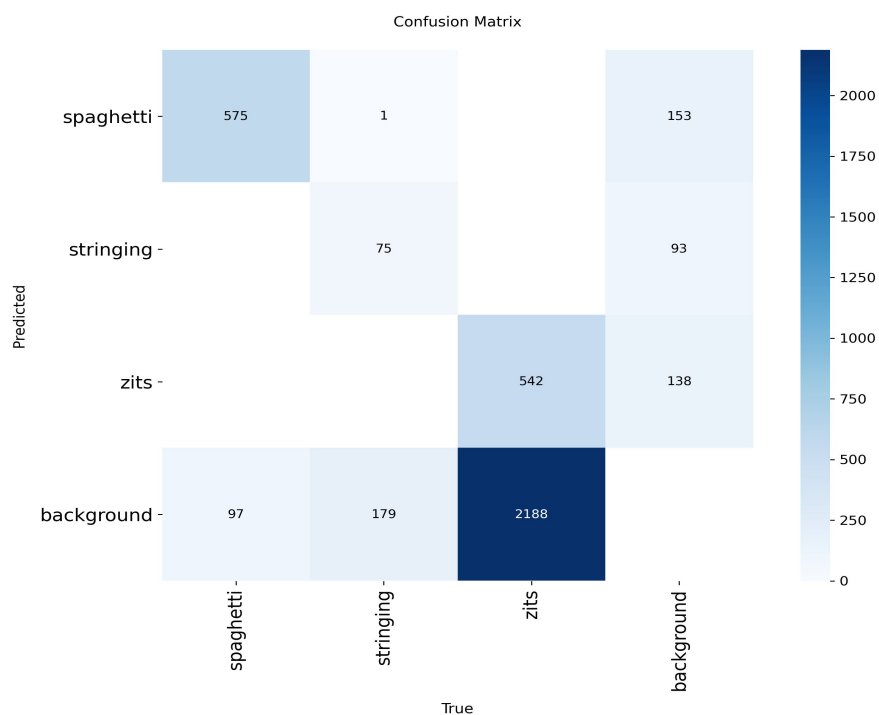


Рис. 3.5. Матриця плутанини, абсолютні значення для навчання №1

3.7. Якісний аналіз прикладів детекції для навчання №1

На рисунку 3.6 наведено приклад результатів детекції на валідаційній вибірці для навчання №1. У тексті під рисунком доцільно описати, чи правильно модель локалізує дефект, чи є пропуски, і чи спостерігаються хибні рамки на фоні.

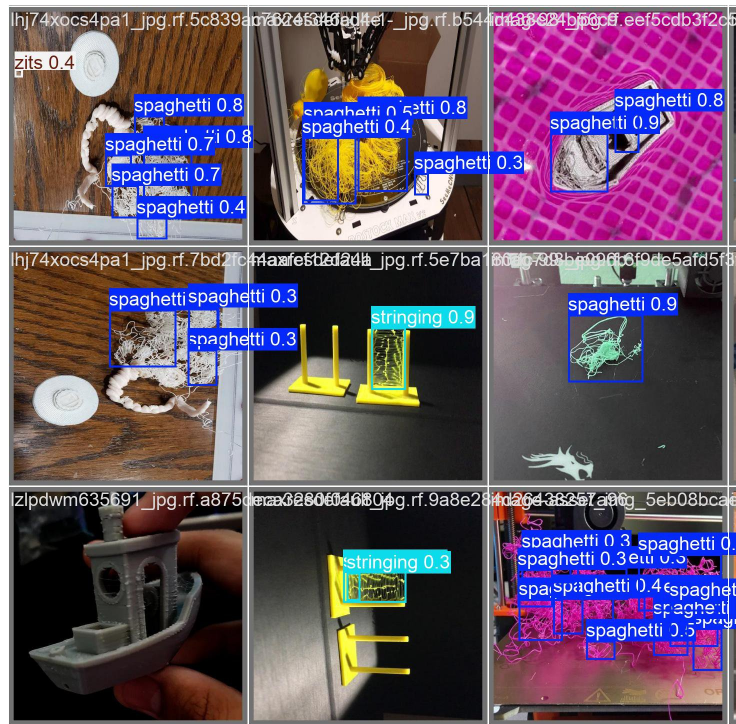


Рис. 3.6. Результати детекції на валідаційній вибірці

На рисунку 3.7 наведено додатковий приклад, який дозволяє порівняти поведінку моделі на іншому кадрі та зафіксувати типові помилки.

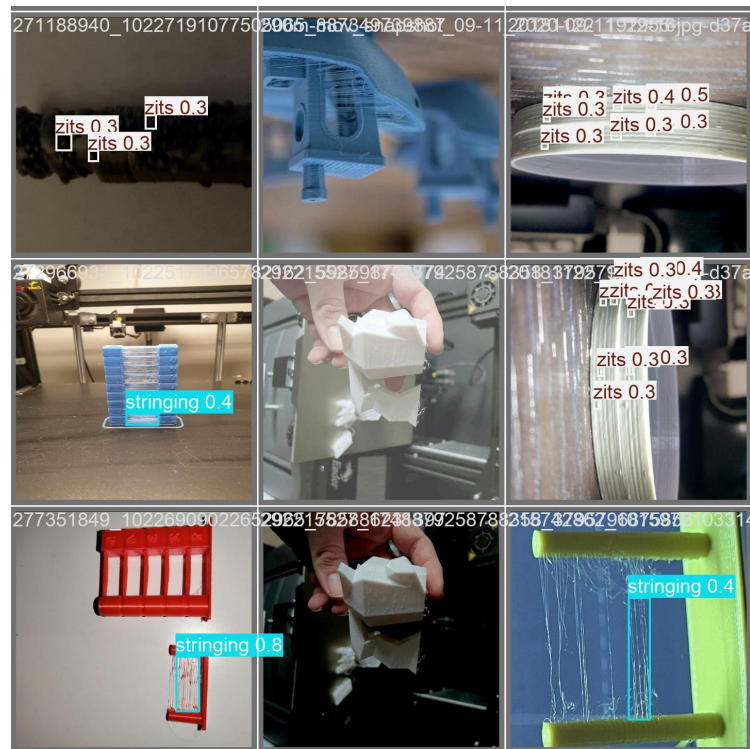


Рис. 3.7. Результати детекції на валідаційній вибірці приклад 2

3.8. Результати навчання №2 криві навчання, узагальнені метрики

Друге навчання є основним експериментом, спрямованим на покращення якості. Зазвичай при збільшенні кількості епох або при уточненні параметрів можна отримати більш стабільні метрики та кращу узагальнювальну здатність. У цьому контексті важливо аналізувати не тільки фінальні значення, але й форму кривих: чи зберігається тренд покращення, чи метрики виходять на плато, чи не з'являються ознаки деградації на валідації.

Криві втрат та метрик якості для навчання №2 наведені на рисунку 3.8. Саме цей рисунок найкраще ілюструє збіжність процесу навчання.

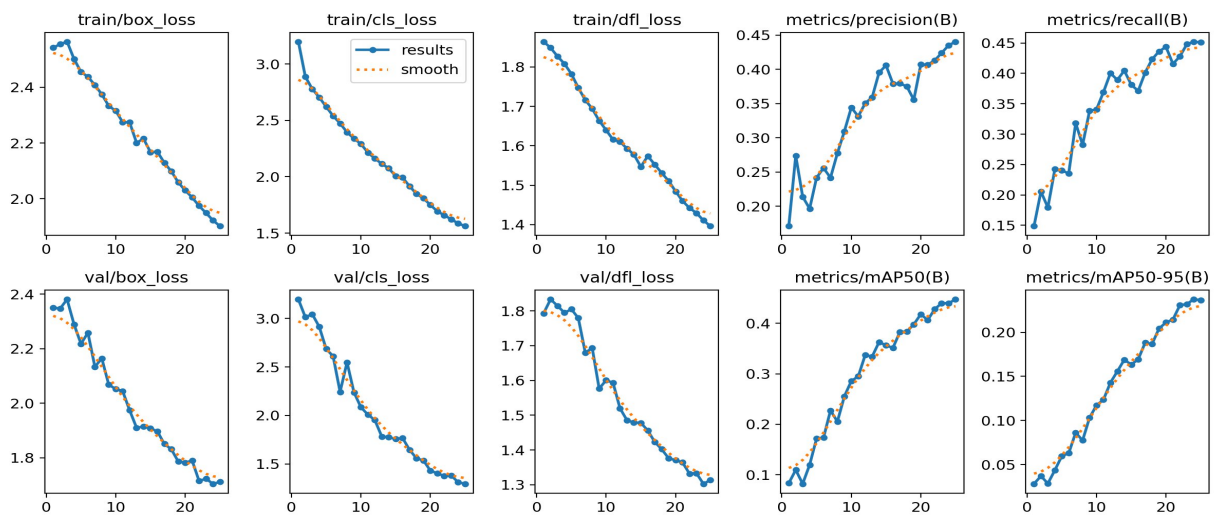


Рис. 3.8. Криві втрат і метрик якості під час навчання №2

За підсумками валідації другого навчання загальні метрики для всіх класів демонструють кращий рівень порівняно з першим навчанням. Окремо важливо, що якість суттєво відрізняється між класами: дефекти типу spaghetti, як правило, мають виразнішу структуру і краще детектуються, тоді як stringing та zits можуть бути низькоконтрастними і складними для локалізації. Саме тому розділ містить не лише узагальнені метрики, але й аналіз помилок за матрицею плутанини та за прикладами детекцій.

3.9. Матриця плутанини для навчання №2 та інтерпретація помилок

Нормалізована матриця плутанини для навчання №2 наведена на рисунку 3.9. Вона показує, у яких випадках модель схильна відносити об'єкти до фону, а також чи є змішування між конкретними класами. Такий аналіз є практично важливим, оскільки дозволяє визначити, які класи потребують додаткових даних або уточнення розмітки.

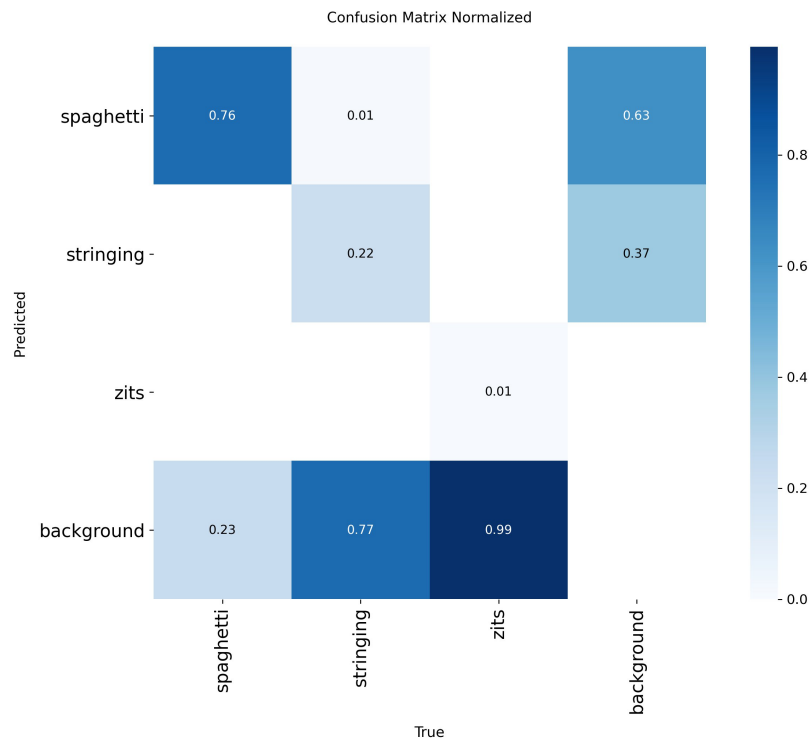


Рис. 3.9. Нормалізована матриця плутанини для навчання №2

Абсолютна матриця плутанини для навчання №2 наведена на рисунку 3.10. Вона допомагає побачити реальні кількості помилок і пов'язати їх із дисбалансом класів. Якщо певний клас має багато інстансів, то навіть невелика частка помилок може давати значні абсолютні значення, що важливо враховувати при практичних висновках.

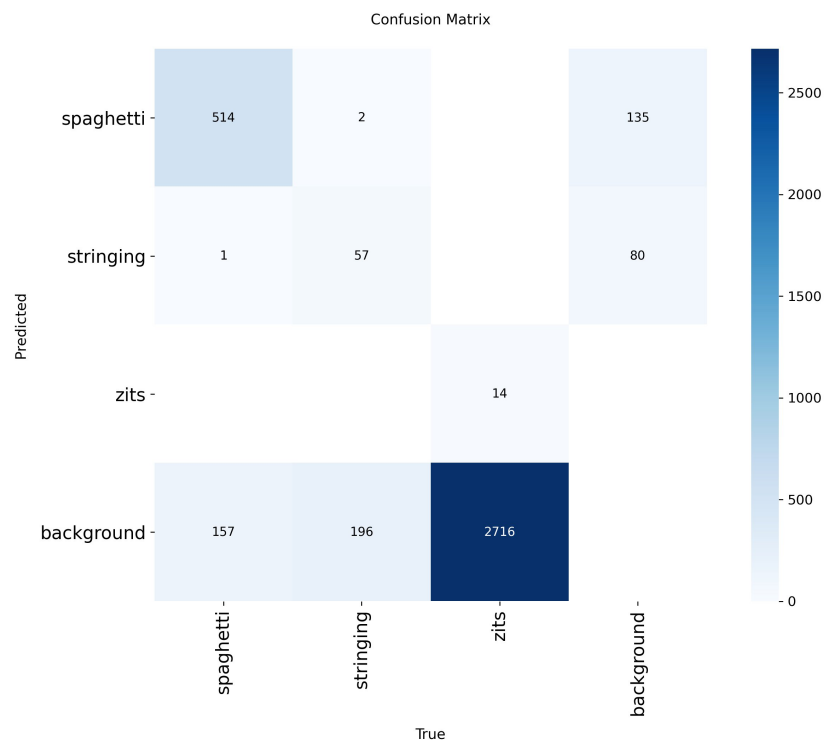


Рис. 3.10. Матриця плутанини для навчання №2

3.10. Аналіз залежностей precision, recall і F1 від порога confidence та вибір робочого порога

Перенесення моделі в прикладну систему вимагає вибору порога confidence. Навіть при однаковій моделі різний поріг може радикально змінити поведінку системи: або вона буде "мовчати" і пропускати дефекти, або буде часто спрацьовувати на шум. У практичному моніторингу FDM-друку важливо не лише мати високу точність, але й забезпечити передбачувану кількість сповіщень у часі.

Залежність precision від порога confidence наведено на рисунку 3.11, залежність recall від порога confidence наведено на рисунку 3.12, а залежність F1 від порога confidence наведено на рисунку 3.13. Ці графіки використовуються для обґрунтування порогів у двох контекстах: окремий поріг для критичного дефекту і більш обережна логіка для дефектів, які складно детектувати на одиничному кадрі.

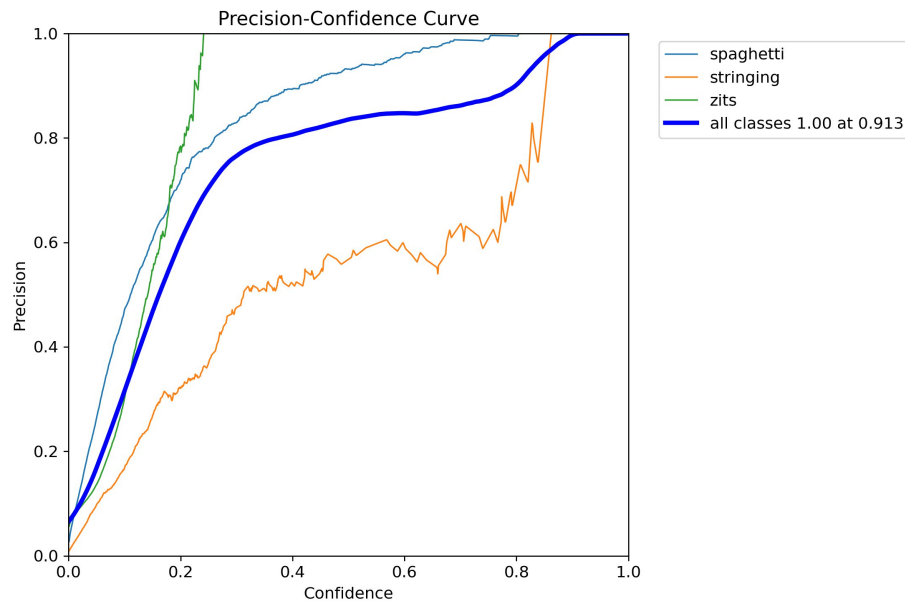


Рис. 3.11. Залежність precision від confidence для навчання №2

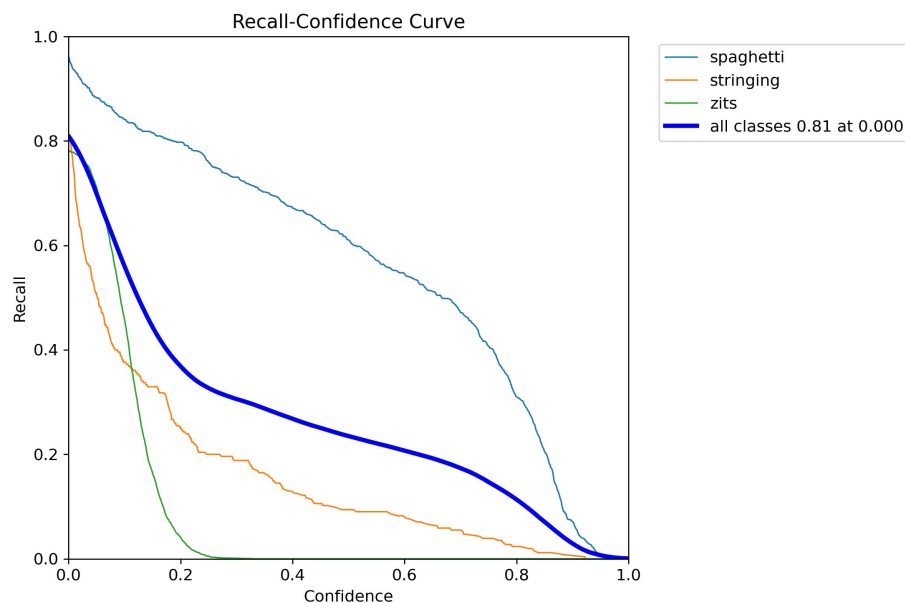


Рис. 3.12. Залежність recall від confidence для навчання №2

Графіки демонструють компроміс між precision і recall при зміні порога confidence: підвищення порога зменшує кількість хибних спрацювань, але збільшує ймовірність пропуску дефектів. Тому робочий поріг обирається з урахуванням балансу метрик та вимог до стабільності системи сповіщень.

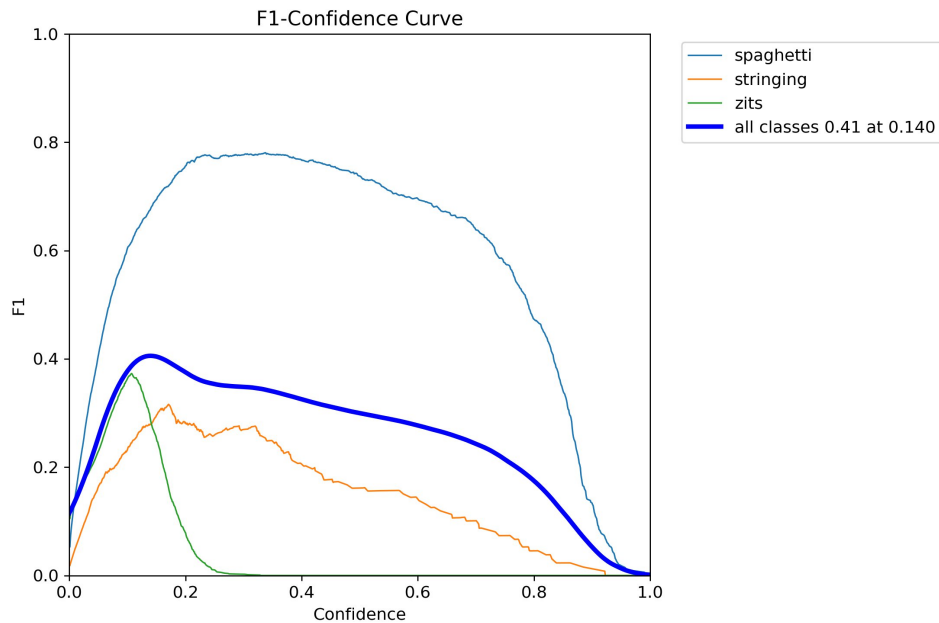


Рис. 3.13. Залежність F1 від confidence для навчання №2

3.11. Крива precision-recall та порівняння класів за складністю детекції

Крива precision-recall є важливим інструментом, оскільки дозволяє бачити компроміс між точністю та повнотою для різних порогів. У задачі моніторингу друку це особливо актуально, тому що оператор зазвичай готовий приймати певну кількість хибних спрацювань, якщо система стабільно "ловить" критичні збої, але не готовий до надмірної кількості сповіщень через дрібні або непевні прояви.

На рисунку 3.14 наведено криву precision-recall для навчання №2. Для виразного класу крива має більшу площу, що узгоджується з вищими значеннями mAP, тоді як для складних класів крива є більш "низькою", що відображає проблеми з узагальненням та залежність від умов зйомки.

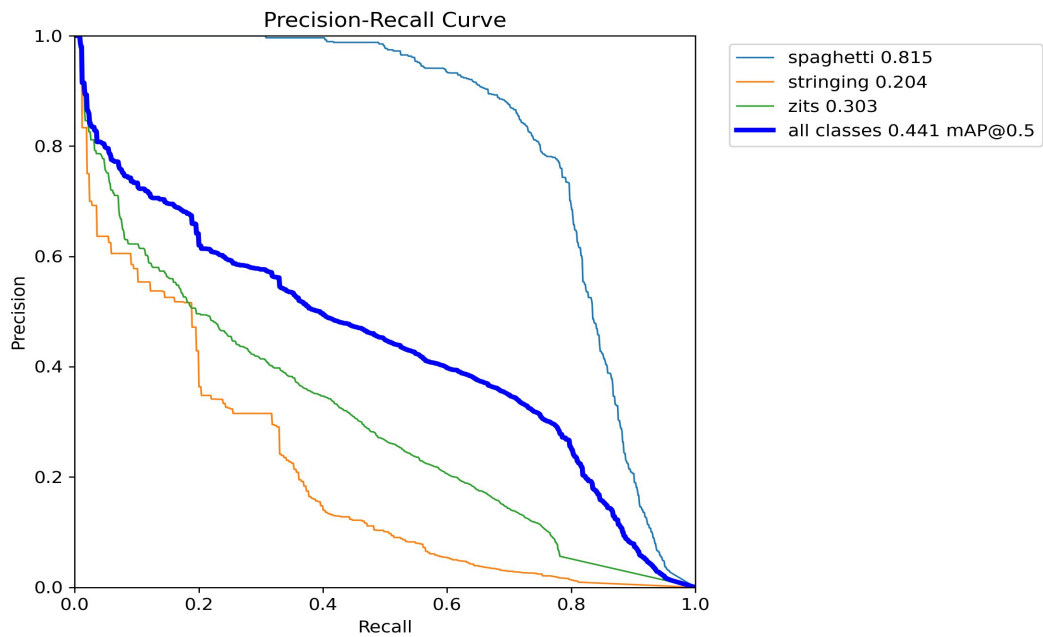


Рис. 3.14. Крива precision-recall для навчання №2

3.12. Якісний аналіз прикладів детекції для навчання №2 та порівняння з навчанням №1

На рисунку 3.15 наведено приклад детекції на валідаційній вибірці для навчання №2.

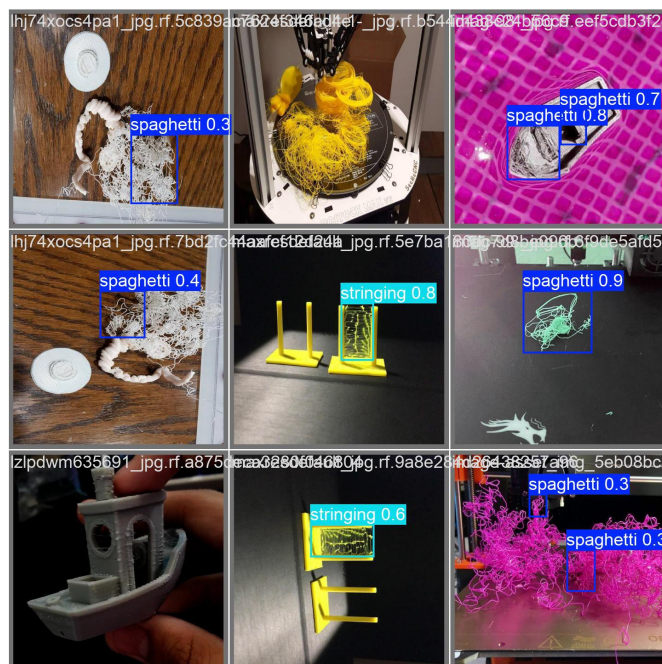


Рис. 3.15. Результати детекції на валідаційній вибірці приклад 1

На рисунку 3.16 наведено другий приклад, який дозволяє показати випадки, коли модель виявляє дефект з меншою впевненістю або локалізує його неідеально. Для задачі моніторингу це важливо, оскільки відхилення можуть формуватися поступово.

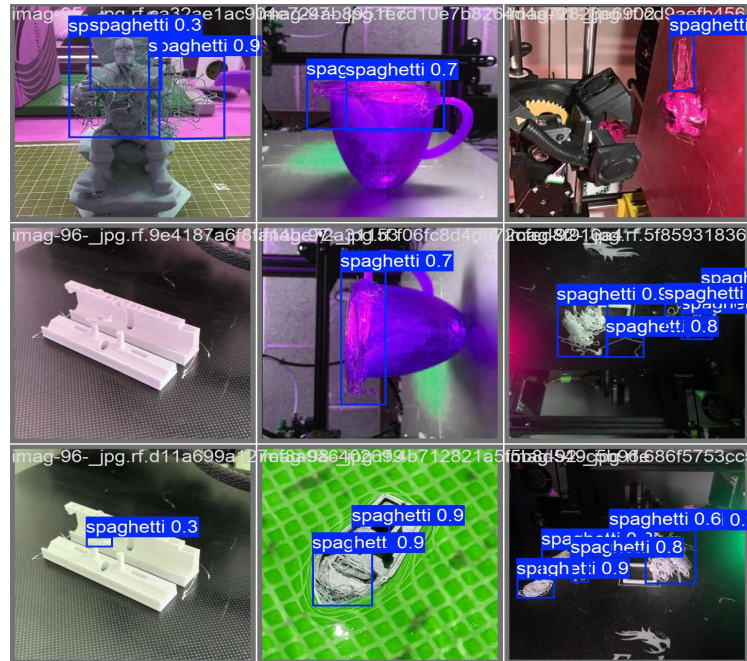


Рис. 3.16. Результати детекції на валідаційній вибірці приклад 2

На рисунку 3.17 наведено третій приклад, який використовується для демонстрації складних сцен, де є фрагменти механіки принтера, фонові текстури і можливі відблиски. Такі приклади важливі для практичної інтерпретації хибних спрацювань, оскільки вони відображають реальні умови експлуатації системи та дозволяють оцінити її стійкість до візуальних перешкод і шумів навколишнього середовища, а також допомагають виявити обмеження застосованих алгоритмів і визначити напрями їх подальшого вдосконалення.

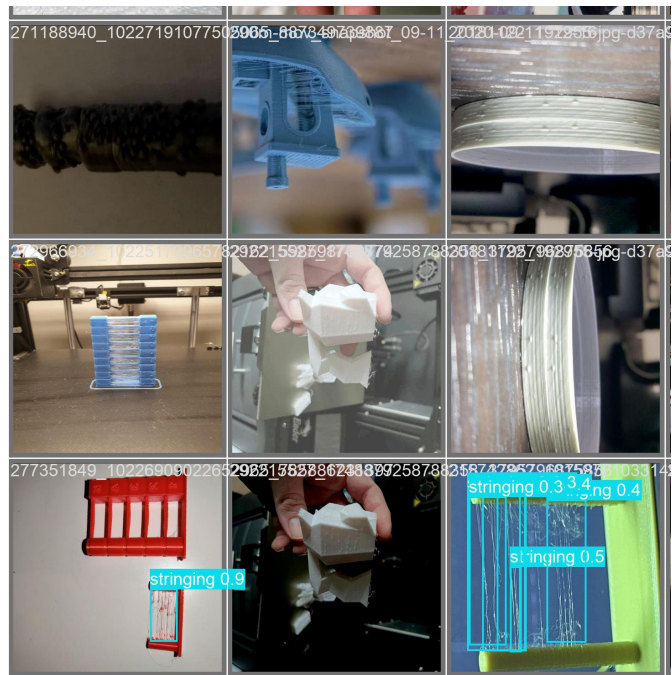


Рис. 3.17. Результати детекції на валідаційній вибірці приклад 3

3.13. Аналіз навчальних батчів та контроль якості даних

Навчальні батчі дозволяють переконатися, що дані не містять систематичних помилок розмітки та що перетворення, які застосовуються під час навчання, не руйнують ознаки дефекту. У випадку задачі FDM-друку це особливо важливо, оскільки частина дефектів може бути тонкою і мало контрастною. Якщо в процесі перетворень такі ознаки "згладжуються" або стають невидимими, модель втрачає здатність їх навчитися.

На рисунку 3.18 наведено приклад навчального батчу для навчання №2. У тексті доцільно зазначити, що візуальна перевірка батчів є елементом контролю якості датасету і дозволяє виявляти проблеми до отримання фінальних метрик.

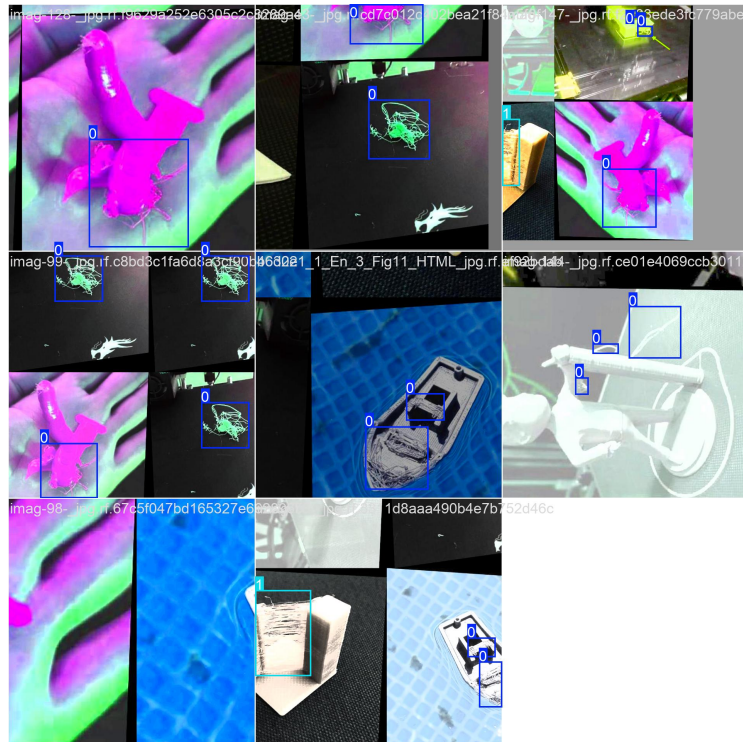


Рис. 3.18. Приклад навчального батчу

На рисунку 3.19 наведено додатковий приклад батчу для навчання №2. Це розширює практичну частину і дозволяє пояснити, що модель бачить різні сцени, різні масштаби та різні варіації дефектів. Наявність таких різномірних прикладів у батчі сприяє кращій узагальнювальній здатності моделі та зменшує ризик переобучення. Крім того, це демонструє, що навчання відбувається в умовах, наближених до реального процесу FDM-друку.

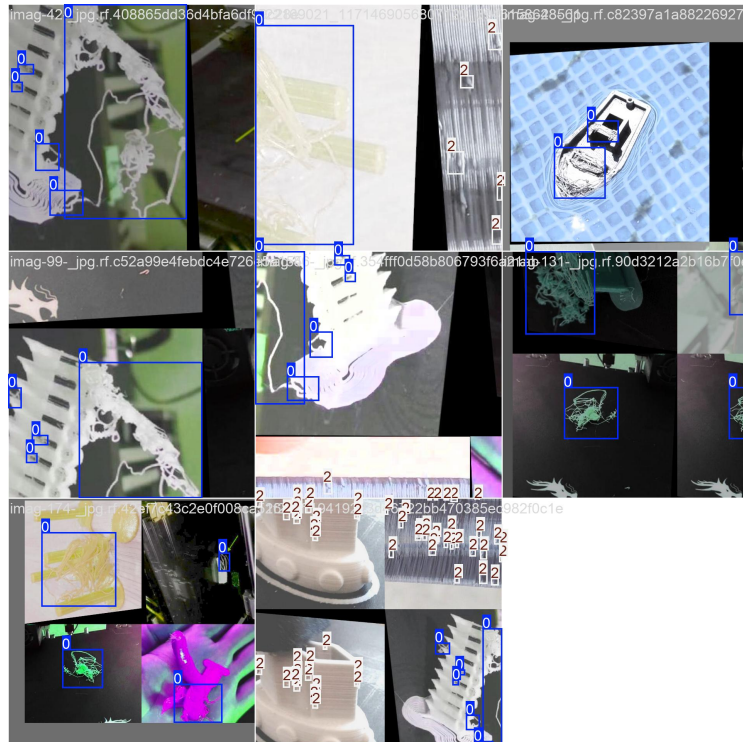


Рис. 3.19. Приклад навчального батчу навчання №2, додатковий приклад

3.14. Прототип прикладного використання а також перевірка папки кадрів і формування візуалізованих результатів

У практичній експлуатації зручно мати режим, який дозволяє проганяти модель по набору кадрів, збережених у папці. Такий режим використовується для тестування, для аналізу проблемних випадків, а також для порівняння різних версій моделі на одному і тому самому наборі зображень. Програмно це реалізується як послідовне читання файлів з папки, виконання *inference* для кожного зображення та збереження результату у вигляді копії зображення з накладеними рамками.

Практична цінність цього режиму полягає у можливості швидко зібрати "банк прикладів" правильних та неправильних спрацювань. Такі приклади згодом можна використати для до-розмітки та розширення датасету, що створює замкнений цикл покращення моделі. При цьому бажано зберігати не тільки зображення з рамками, але й текстовий журнал результатів, наприклад, кількість виявлень, класи, середні *confidence*, що спрощує статистичний аналіз.

3.15. Прототип для відеопотоку та часова фільтрація спрацювань

Перехід від статичних кадрів до відеопотоку змінює вимоги до системи. У відео можливі короткі спотворення, тремтіння камери, зміна фокуса або випадкове перекриття. Якщо реагувати на одиничний кадр, система буде давати нестабільний результат. Тому застосовується часова фільтрація: дефект вважається підтвердженням лише тоді, коли модель виявляє його протягом певного часу або протягом певної кількості послідовних кадрів [3, 4].

Часова фільтрація може бути реалізована як накопичення лічильника спрацювань для кожного класу, який збільшується при наявності детекції і зменшується або обнуляється при її відсутності. Така логіка дозволяє зробити систему більш стійкою без необхідності штучно підвищувати поріг confidence. Це особливо важливо для складних класів, де одиничні детекції можуть бути непевними, але стійка поява ознаки протягом кількох секунд є сильним сигналом.

У межах цієї роботи зазначений підхід розглядається як рекомендація для практичного впровадження, що дозволяє підвищити корисність прототипу, не змінюючи базову модель. Схему взаємодії програмних модулів наведено у додатку В.

3.16. Обмеження, фактори впливу та напрями удосконалення

Отримані результати показують типові обмеження комп'ютерно-зорового підходу в умовах FDM-друку. Найбільш значущими є дисбаланс класів, залежність від освітлення та ракурсу, а також різна складність дефектів. Класи, які мають виразні візуальні ознаки, детектуються стабільніше. Класи, які складаються з тонких і низькоконтрастних елементів, вимагають більшої кількості даних та більш ретельної розмітки.

Для підвищення якості системи доцільно розширювати датасет саме за рахунок складних випадків, а також додавати кадри з різними матеріалами, кольорами філаменту, різними типами моделей і різними фонами. Практично

корисним є підхід, коли система в експлуатації зберігає "спірні" кадри, а потім ці кадри до-розмічаються і додаються до тренувального набору. Така стратегія дозволяє поступово підвищувати стабільність моделі та зменшувати кількість помилок у реальних умовах.

3.17. Настільний застосунок PrintGuardian як приклад інтеграції детектора у систему моніторингу

Для демонстрації практичної інтеграції нейромережевого детектора у процес спостереження за FDM-друком було розроблено прототип настільного застосунку PrintGuardian. На відміну від тестових сценаріїв, де модель перевіряється на окремих зображеннях або на папці кадрів, у цьому випадку реалізовано повний цикл роботи з відеопотоком, відображенням результатів у реальному часі та формуванням реакції на стійке виявлення відхилення.

Прототип реалізовано на мові Python з використанням PyQt6 для побудови інтерфейсу, OpenCV для захоплення кадрів з USB-камери, RTSP-потоків або відеофайлу та бібліотеки Ultralytics для виконання інференсу моделі YOLOv8. Для взаємодії із зовнішніми сервісами застосовано HTTP-запити (requests): надсилання команд до OctoPrint API та формування push-сповіщень через ntfy.sh. Налаштування мережевих параметрів, ключів доступу, порогів та обраного джерела відео зберігаються у файлі settings.json, що забезпечує відтворюваність експериментів та зручність повторного запуску. В додатку Д наведений лістинг програми.

Оскільки обробка відео та інференс можуть бути ресурсоемними, у прототипі використано окремий потік виконання (QThread) для кадрів і детекції. Це дозволяє уникнути зависання інтерфейсу та забезпечує стабільну роботу при тривалому моніторингу. Рішення щодо спрацювання приймається не за одиничним кадром, а на основі часової фільтрації: тригер активується лише тоді, коли дефект фіксується у кількох кадрах поспіль. Такий підхід зменшує кількість хибних спрацювань і є критичним для практичного застосування, коли кожен "фальшивий" тригер може призвести до зайвої зупинки друку та втрати часу.

Після підтвердження стійкого виявлення відхилення прототип виконує набір дій: формує зображення-доказ (кадр зі візуалізацією детекції), надсилає сповіщення оператору та може відправити керувальну команду на принтер через OctoPrint API. Таким чином, застосунок використовується як демонстраційний модуль моніторингу, який поєднує нейромережеву детекцію, візуалізацію результатів та інтеграцію з інфраструктурою керування 3D-принтером. Структурну схему системи наведено у додатку Г.

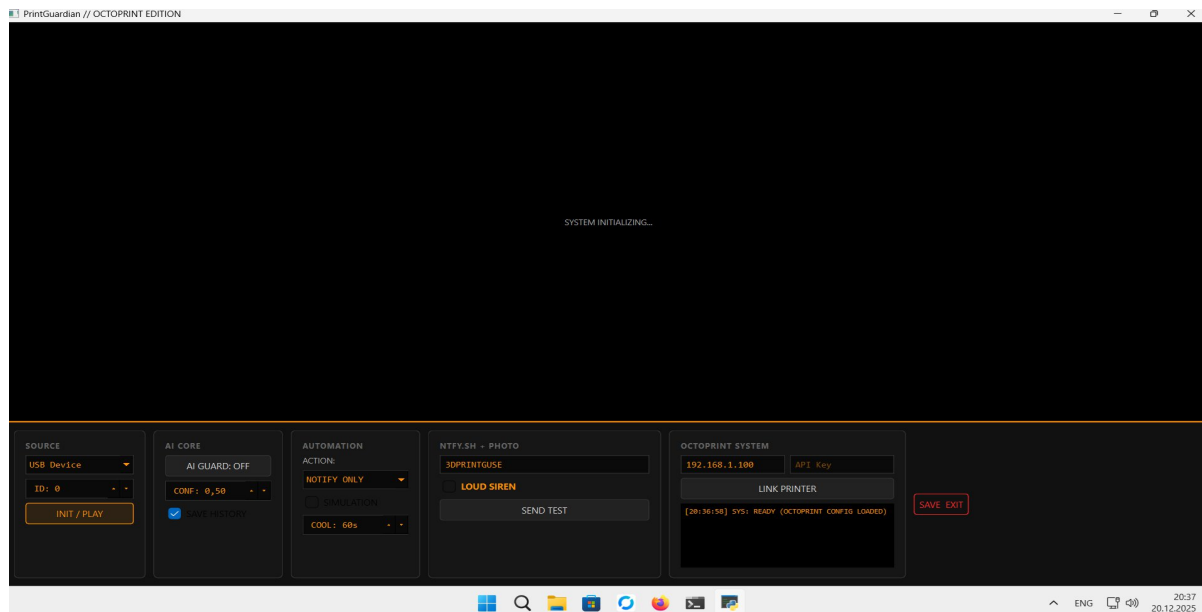


Рис. 3.20. Інтерфейс настільного застосунку PrintGuardian для моніторингу процесу друку

3.18. Візуалізація результатів детекції та інтерфейс користувача прототипу PrintGuardian

Інтерфейс користувача в PrintGuardian орієнтований на швидке відображення поточного стану друку та параметрів, які впливають на реакцію системи. У вікні застосунку передбачено область перегляду відео, журнал подій (логи), а також елементи керування підключенням до джерела відео та параметрами чутливості. Така організація дозволяє оператору отримувати пояснюваний результат: не лише

повідомлення "є/немає дефекту", а й наочне підтвердження у вигляді накладеної рамки на кадрі та числового значення confidence.

Візуалізація результатів детекції на кадрі виконується стандартним для задач детекції способом: на область, у якій модель виявила ознаки дефекту, накладається прямокутник (bounding box), а поряд виводиться назва класу та значення confidence. Для задачі моніторингу друку це важливо, оскільки оператор може одразу оцінити, чи відповідає спрацювання реальному дефекту, або це результат відблиску, тіні чи короткочасного артефакту. При використанні такого відображення в реальному часі знижується ризик неправильного трактування сигналу та спрощується аналіз причин збою.

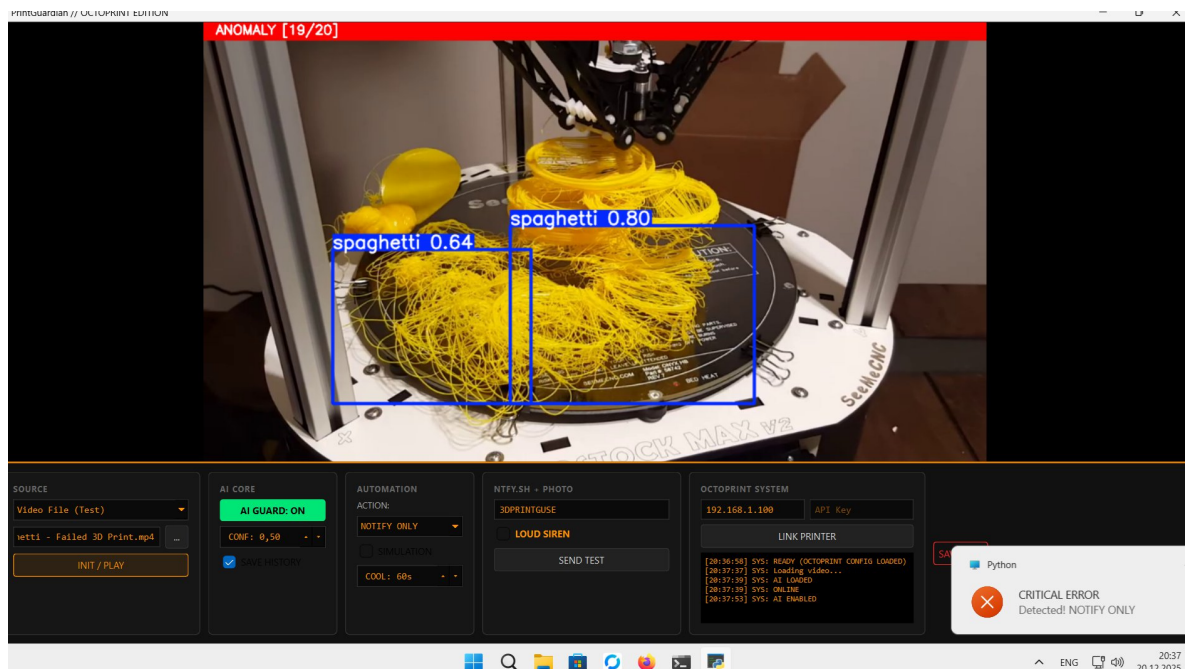


Рис. 3.21. Приклад виявлення відхилення на кадрі з накладеною рамкою та значенням confidence

3.19. Отримання повідомлення про відхилення та передача керувальної команди через OctoPrint

Для практичного застосування системи моніторингу важливо не лише виявити відхилення, але й оперативно донести інформацію до оператора та, за необхідності,

ініціювати дію, що мінімізує наслідки дефекту. У прототипі PrintGuardian це реалізовано через віддалене push-сповіщення на мобільний пристрій та можливість надсилання керувальної команди принтеру через OctoPrint API.

Після підтвердження стійкого спрацювання тригера програма формує повідомлення для сервісу ntfy.sh, додаючи текстовий опис події та зображення (скріншот кадру з накладеними рамками детекції і значенням confidence). З практичної точки зору такий формат є зручним, оскільки оператор отримує не тільки сигнал тривоги, а й візуальний контекст: що саме бачила модель, де розташована зона проблеми та з якою впевненістю визначено відхилення. Це дозволяє швидко оцінити ситуацію без доступу до робочого ПК та прийняти рішення щодо подальших дій.

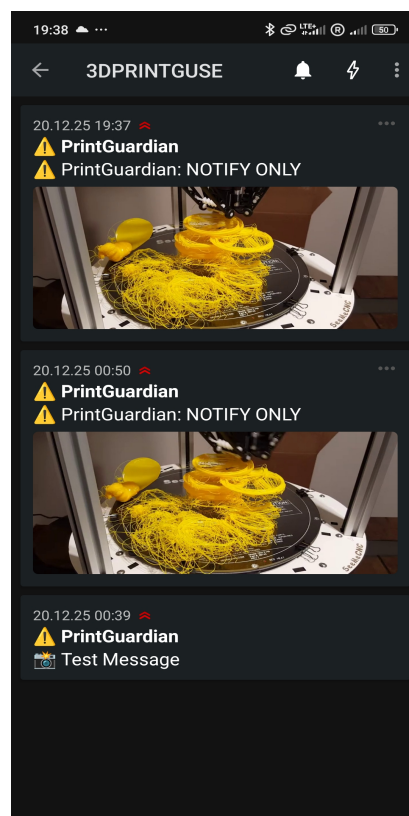


Рис. 3.22. Приклад push-повідомлення про відхилення у процесі друку

Паралельно з формуванням повідомлення може виконуватися керувальна дія через OctoPrint API. У базовому сценарії застосовується команда PAUSE, яка переводить принтер у безпечний режим очікування та дозволяє оператору

перевірити стан виробу і обладнання. Такий підхід є доцільним для настільного друку, оскільки зменшує ризик подальшого накопичення матеріалу в зоні друку та залишає можливість відновлення процесу після огляду. У разі критичної ситуації або за налаштуваннями користувача може застосовуватися команда STOP, яка завершує поточне завдання друку. Блок схема алгоритму роботи системи наведено у додатку Б.

РОЗДІЛ 4

ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ

4.1. Охорона праці

Кваліфікаційна робота присвячена розробленню та дослідженню методів і засобів адаптивного керування параметрами 3D-принтера на основі аналізу процесу друку. Практична реалізація отриманих результатів передбачає експлуатацію адитивного обладнання в лабораторних або виробничих умовах, використання витратних матеріалів та виконання операцій з післяобробки надрукованих деталей. Це зумовлює необхідність аналізу шкідливих і небезпечних виробничих факторів, які можуть виникати під час роботи з 3D-принтером, а також розроблення заходів щодо забезпечення безпечних умов праці оператора [6].

Робоче місце оператора 3D-принтера, як правило, розміщують у закритому приміщенні, де встановлено один або кілька принтерів, допоміжний стіл для підготовки й обробки моделей, а також шафи для зберігання витратних матеріалів та інструменту. На умови праці впливають параметри мікроклімату (температура, вологість, швидкість руху повітря), рівень шуму від роботи електромеханічних вузлів, якість повітря, наявність пилу й випарів, а також характер виконуваних операцій. До основних шкідливих та небезпечних факторів, що можуть мати місце при роботі з 3D-принтером, належать механічні, термічні та електричні небезпеки, а також фактори, пов'язані з мікрокліматом і організацією робочого простору.

Найбільш поширеними у повсякденній роботі оператора є механічні травмонебезпечні фактори, пов'язані з використанням ручних інструментів та обробкою надрукованих деталей. При експлуатації 3D-принтерів оператор регулярно використовує шпателі, ножі, кусачки, надфілі та інші ріжучі інструменти для відокремлення моделі від платформи й видалення підтримок чи зайвого матеріалу. Такі операції часто виконуються в обмеженому просторі робочої камери або на столі з великою кількістю дрібних деталей і відходів друку; для зняття

моделей, що міцно прилипили до платформи, доводиться прикладати значні зусилля, нерідко в незручних положеннях рук [7].

За таких умов зростає ризик зісковзування інструмента, його раптового зриву та отримання порізів або проколів м'яких тканин кисті й передпліччя. Додаткову небезпеку становлять гострі краї незачищених деталей, фрагменти підтримок і обрізки пластику. Зменшення ризику травмування досягається шляхом використання справних інструментів з ергономічними ручками, роботи в захисних рукавичках, застосування безпечних напрямків зусилля (від себе і вбік від тіла), а також фіксації деталі струбцинами чи в лещатах під час обробки. Інструменти слід зберігати у спеціальних підставках або ящиках, не залишаючи їх на краю стола чи в зоні проходу.

Окрім механічних травм, при експлуатації 3D-принтера суттєву небезпеку становлять фактори термічного впливу. Сопло екструдера, нагрівальний стіл, блок гарячого кінця (hotend) та окремі елементи системи підігріву в процесі друку мають температуру, яка суттєво перевищує безпечні для дотику значення (до 60–110 °C для платформи та понад 200 °C для сопла). Оператор виконує налаштування висоти стола, заміну філаменту, очищення сопла або зняття моделі безпосередньо в зоні розташування цих елементів, іноді при ще не повному їх охолодженні [8].

Унаслідок цього зростає ймовірність випадкового контакту з гарячими поверхнями, що створює ризик термічних опіків. Для зменшення дії цього фактора оператор повинен орієнтуватися на покази температури, що відображаються на дисплеї принтера або в програмному забезпеченні керування друком, суворо дотримуватися встановлених температурних режимів, не торкатися відкритих нагрітих поверхонь та використовувати спеціальні інструменти для маніпуляцій із деталями. Перед виконанням робіт з обслуговування, які не потребують нагріву, вузли необхідно охолоджувати до безпечної температури та відключати живлення обладнання.

Окрему групу небезпек становлять механічні ризики, пов'язані з рухомими вузлами та масою обладнання. Під час очищення та технічного обслуговування 3D-принтера оператор взаємодіє з каретками, ременями, напрямними та іншими

елементами механіки, а також із досить масивною конструкцією корпусу. Неправильне фіксування рухомих частин або виконання робіт при ввімкненому живленні створює ризик затягування одягу чи пальців у механізми, притискання деталей та інших механічних травм.

Крім того, при переміщенні або піднятті принтера існує небезпека перенавантаження опорно-рухового апарату, падіння обладнання та ушкодження спини й суглобів. Для мінімізації зазначених небезпек роботи з обслуговування, як правило, виконують при відключеному живленні та зафіксованих рухомих вузлах. Виняток становлять операції, які технологічно потребують нагріву, наприклад очищення сопла при заданій температурі плавлення філаменту, що вимагає дотримання підвищених запобіжних заходів.

Ще одним важливим аспектом безпеки є електробезпека. 3D-принтери та допоміжне електрообладнання працюють від мережі 220 В, тому необхідно забезпечити справність ізоляції проводів, наявність заземлення, використання сертифікованих подовжувачів і захисних автоматів. Забороняється торкатися внутрішніх електричних вузлів при підключеному живленні, перевантажувати розетки або використовувати пошкоджені кабелі та розгалужувачі.

Нагрівальні елементи принтерів також впливають на мікроклімат приміщення, зокрема сприяють підвищенню температури повітря та локальному тепловому навантаженню. Тому важливо підтримувати параметри мікроклімату в межах комфортних значень шляхом регулярного провітрювання та використання систем вентиляції. Реалізація зазначених заходів забезпечує безпечні умови праці оператора під час керування параметрами 3D-принтера на основі аналізу процесу друку.

4.2. Безпека в надзвичайних ситуаціях

Відповідно до методичних вказівок з дисциплін «Цивільна безпека» та «Безпека в надзвичайних ситуаціях», у підрозділі 4.2 необхідно проаналізувати можливі надзвичайні ситуації техногенного та природного характеру, оцінити

стійкість функціонування об'єкта й обґрунтувати комплекс інженерно-технічних, організаційних та режимних заходів захисту персоналу.

Під час роботи з 3D-принтером та іншим електронним обладнанням можливе виникнення надзвичайних ситуацій техногенного характеру локального масштабу: займання або задимлення обладнання, коротке замикання в електромережі, ураження електричним струмом, травмування працівників ріжучими або рухомими частинами. Для мінімізації наслідків таких подій необхідно завчасно передбачити можливі сценарії аварій, розробити алгоритми дій персоналу та забезпечити відповідні засоби захисту.

4.2.1. Пожежна безпека та запобігання аваріям

Найімовірнішою надзвичайною ситуацією при роботі з 3D-принтером є пожежа або задимлення в зоні друку внаслідок короткого замикання, перегріву елементів, використання неякісних клейових матеріалів чи неправильного розміщення легкозаймистих предметів поруч із нагрівальними елементами. Для запобігання таким випадкам необхідно регулярно оглядати стан електропроводки та клемних з'єднань, не залишати принтер без нагляду на початкових етапах друку, дотримуватися допустимих режимів навантаження на електромережу, не закривати вентиляційні отвори обладнання, не накривати принтер легкозаймистими матеріалами.

У приміщенні мають бути справні первинні засоби пожежогасіння (вогнегасники відповідного класу дії), а персонал – проінструктований щодо місця їх розташування та правил використання. Категорія приміщення за пожежною та вибухопожежною небезпекою визначається згідно з чинними нормами, що враховують кількість горючих матеріалів, особливості технологічного процесу та інженерних систем. У разі виявлення запаху горілого пластику, диму або ознак займання оператор повинен негайно припинити друк, вимкнути живлення обладнання (спочатку штатною кнопкою, потім – від мережі), повідомити відповідальну особу й за потреби викликати пожежно-рятувальну службу за номером 101. При незначному загорянні в електрообладнанні допускається

використання вогнегасника відповідного типу, не спрямовуючи струмінь безпосередньо на людину та не торкаючись металевих частин, що можуть бути під напругою. Евакуацію з приміщення здійснюють згідно з планом евакуації; при цьому заборонено користуватися ліфтами, а двері слід зачиняти за собою для зменшення поширення диму й вогню.

4.2.2. Ураження електричним струмом

Іншою можливою надзвичайною ситуацією є ураження оператора електричним струмом при дотику до пошкоджених проводів чи відкритих струмопровідних частин. У такому випадку першочерговим завданням є негайне знеструмлення ділянки (відключення автоматичного вимикача, вимикача живлення) й лише після цього – надання допомоги потерпілому. Якщо відключити живлення неможливо, слід відтягнути потерпілого від струмопровідних частин за допомогою сухих діелектричних предметів (дерев'яна палиця, сухий одяг), не торкаючись відкритих ділянок тіла.

Після звільнення від дії струму необхідно викликати бригаду екстреної медичної допомоги за номером 103 та, за потреби, розпочати серцево-легеневу реанімацію відповідно до затверджених алгоритмів домедичної допомоги.

4.2.3. Травмування персоналу та перша допомога

Можливі також травми, пов'язані з порізами, опіками чи ушкодженням опорно-рухового апарату. У випадку порізів слід зупинити кровотечу (накласти пов'язку, що давить), промити рану й накласти асептичну пов'язку; при опіках – охолодити уражену ділянку проточною водою, не здирати прилиплий одяг і звернутися по медичну допомогу. При підозрі на перелом або серйозне ушкодження хребта потерпілого необхідно зафіксувати в положенні, що не посилює біль, та викликати швидку допомогу.

Надання першої допомоги здійснюють відповідно до вимог Кодексу цивільного захисту України та інших нормативно-правових актів у сфері охорони

праці й цивільної безпеки, які регламентують алгоритми дій при травмах, опіках та гострих станах.

4.2.4. Організаційні заходи та планування дій у НС

Для організації безпеки в надзвичайних ситуаціях у приміщенні, де розміщені 3D-принтери, має бути розроблений і наочно розміщений план евакуації з позначенням основних і запасних виходів, місць розташування вогнегасників і засобів пожежної сигналізації. Персонал повинен бути ознайомлений з порядком дій у разі пожежі, аварії в електромережі, травмування працівника, а також проходити періодичні інструктажі з питань цивільного захисту та надання домедичної допомоги.

Загальний комплекс інженерно-технічних, технологічних та організаційних заходів, спрямованих на підвищення стійкості функціонування об'єкта та зниження ризику виникнення надзвичайних ситуацій, відповідає вимогам навчальних і методичних посібників з цивільної безпеки, що застосовуються в ТНТУ.

ВИСНОВКИ

У кваліфікаційній роботі магістра розглянуто задачу адаптивного керування параметрами 3D-принтера на основі аналізу процесу друку, в якій ключовим практичним інструментом є комп'ютерно-зорова система моніторингу стану FDM-друку. Обґрунтовано актуальність теми з огляду на високу вартість помилок друку, що призводять до втрати матеріалу, часу та ресурсу обладнання, а також показано доцільність автоматизованого виявлення відхилень за відеопотоком як для домашнього, так і для напівпрофесійного використання.

У першому розділі виконано аналітичний огляд процесу FDM-друку як керованого технологічного процесу та сформовано вимоги до системи візуального контролю, зокрема безперервність спостереження, працездатність у реальних умовах освітлення, можливість використання типової камери та інтеграції результатів аналізу з програмним середовищем користувача. На основі аналізу характерних дефектів друку визначено перелік цільових класів відхилень та обґрунтовано вибір підходу детекції об'єктів як такого, що забезпечує баланс між точністю локалізації та обчислювальною ефективністю.

У другому розділі узагальнено теоретичні засади побудови систем комп'ютерного зору для задач детектування об'єктів та обґрунтовано застосування архітектури YOLO для виявлення дефектів FDM-друку. Описано процес формування набору даних, підготовки розмітки, розбиття на підвибірки та нормалізації вхідних зображень. Показано, що якість даних, умови зйомки та баланс класів суттєво впливають на здатність моделі до узагальнення та стабільної роботи поза межами навчальної вибірки.

У третьому розділі реалізовано практичну частину роботи: проведено два цикли навчання моделі YOLOv8 та виконано експериментальну оцінку якості. Для моделі, навченої протягом 25 епох, досягнуто базового рівня працездатності з показниками precision 0.440, recall 0.451, mAP@0.5 0.448 та mAP@0.5:0.95 0.237. Навчання моделі протягом 100 епох дозволило покращити всі основні метрики, зокрема досягти значень precision 0.549, recall 0.500, mAP@0.5 0.514 та

mAP@0.5:0.95 0.305. Аналіз результатів за окремими класами показав найвищу ефективність для дефекту типу spaghetti, що є критичним з точки зору зриву друку, та нижчу якість для класів stringing і zits, що пояснюється їх меншою візуальною виразністю та дисбалансом у розмітці.

У результаті дослідження сформовано та підготовлено набір даних, реалізовано нейромережеву модель детекції дефектів FDM-друку, проведено її навчання та кількісну оцінку якості. Отримані результати підтвердили працездатність обраного підходу та його придатність для використання в системах раннього виявлення критичних збоїв процесу друку.

Разом з тим результати роботи створюють основу для подальшого вдосконалення системи. Перспективними напрямками розвитку є розширення та балансування набору даних, оптимізація параметрів навчання, використання GPU-прискорення, а також впровадження часової фільтрації рішень на основі аналізу послідовності кадрів, що дозволить зменшити кількість хибних спрацювань і підвищити надійність системи в реальних умовах експлуатації.

СПИСОК ВИКОРИСТАННИХ ДЖЕРЕЛ

1. Луцик Н.С., Луцків А.М., Осухівська Г.М., Тиш Є.В. Програма та методичні рекомендації з проходження практики за тематикою кваліфікаційної роботи для студентів спеціальності 123 "Комп'ютерна інженерія" другого (магістерського) рівня вищої освіти усіх форм навчання. Тернопіль: ТНТУ. 2024. 45 с.
2. Луцик Н.С., Луцків А.М., Осухівська Г.М., Тиш Є.В. Методичні рекомендації до виконання кваліфікаційної роботи магістра для студентів спеціальності 123 "Комп'ютерна інженерія" другого (магістерського) рівня вищої освіти усіх форм навчання. Тернопіль. 2024. 44 с.
3. D. Velychko, H. Osukhivska, Y. Palaniza, N. Lutsyk, Ł. Sobaszek. Artificial Intelligence Based Emergency Identification Computer System. Advances in Science and Technology Research Journal, 18(2), 2024. P.296-304.
4. Величко Д. Алгоритм роботи комп'ютеризованої системи відеонагляду з функцією ідентифікації екстрених ситуацій. Матеріали XI науково-технічної конференції „Інформаційні моделі, системи та технології“, 2023. С.138.
5. Варавін А.В., Лецишин Ю.З., Чайковський А.В. Методичні вказівки до виконання курсового проєкту з дисципліни "Дослідження і проєктування комп'ютерних систем та мереж" для здобувачів другого (магістерського) рівня вищої освіти спеціальності 123 "Комп'ютерна інженерія" усіх форм навчання. Тернопіль: ТНТУ, 2024. 32 с.
6. Стручок В.С. Безпека в надзвичайних ситуаціях: методичний посібник для здобувачів освітнього ступеня "магістр" всіх спеціальностей денної та заочної (дистанційної) форм навчання. Тернопіль: ФОП Паляниця В.А., 156 с.
7. Стручок В.С. Техноекологія та цивільна безпека. Частина "Цивільна безпека": навчальний посібник. Тернопіль: ФОП Паляниця В.А., 156 с.
8. Методичний посібник для здобувачів освітнього ступеня "магістр" всіх спеціальностей денної та заочної (дистанційної) форм навчання "Безпека в надзвичайних ситуаціях". Тернопіль: ФОП Паляниця В.А. 156 с

9. Стручок В.С. Навчальний посібник "Техноекологія та цивільна безпека. Частина "Цивільна безпека". Тернопіль: ФОП Паляниця В.А. 156 с.

10. Кобрин О.В. "Комп'ютерно-зорова система контролю стабільності процесу FDM-друку" Матеріали XIII науково-технічної конференції Інформаційні моделі, системи та технології Тернопільського національного технічного університету імені Івана Пулюя (Тернопіль, 17-18 грудня 2025 року), Тернопіль: ТНТУ, 2025. С. 222 .

11. Кобрин О.В. "Експериментальна оцінка комп'ютерно-зорової системи виявлення відхилень у процесі FDM-друку": Матеріали XIII науково-технічної конференції Інформаційні моделі, системи та технології Тернопільського національного технічного університету імені Івана Пулюя (Тернопіль, 17-18 грудня 2025 року), Тернопіль: ТНТУ, 2025. С. 115.

12. M. Chebiyyam, R. D. Reddy, D. P. Dogra, H. Bhaskar, L. Mihaylova. Motion anomaly detection and trajectory analysis in visual surveillance. Multimedia Tools and Applications, vol. 77, P. 16223-16248, 2018.

Додаток А
Тези конференції

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ТЕРНОПІЛЬСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
ІМЕНІ ІВАНА ПУЛЮЯ

МАТЕРІАЛИ

ХІІІ НАУКОВО-ТЕХНІЧНОЇ КОНФЕРЕНЦІЇ
«ІНФОРМАЦІЙНІ МОДЕЛІ,
СИСТЕМИ ТА ТЕХНОЛОГІЇ»



17-18 грудня 2025 року

ТЕРНОПІЛЬ
2025

Y. Holotenko, R. Koroliuk
 MICROCONTROLLERS OF INDUSTRIAL INTERNET OF THINGS:
 ARCHITECTURE OVERVIEW

О. Кобрын, Н. Стадник ЕКСПЕРИМЕНТАЛЬНА ОЦІНКА КОМП'ЮТЕРНО-ЗОРОВОЇ СИСТЕМИ ВИЯВЛЕННЯ ВІДХИЛЕНЬ У ПРОЦЕСІ FDM-ДРУКУ O. Kobryn, N. Stadnyk EXPERIMENTAL EVALUATION OF A COMPUTER VISION SYSTEM FOR DETECTING DEVIATIONS IN THE FDM PRINTING PROCESS	115
В. Ковальський, Є. Тиш ПРОГНОЗУВАННЯ МЕРЕЖЕВИХ АНОМАЛІЙ У ХМАРНОМУ СЕРЕДОВИЩІ З ВИКОРИСТАННЯМ ГЕНЕРАТИВНИХ АЛГОРИТМІВ V. Kovalskyi, Ie. Tysh PREDICTION OF NETWORK ANOMALIES IN A CLOUD ENVIRONMENT USING GENERATIVE ALGORITHMS	116
В. Ковальський, Є. Тиш ІНТЕЛЕКТУАЛЬНА СИСТЕМА ПРОГНОЗУВАННЯ АНОМАЛІЙ МЕРЕЖЕВОГО ТРАФІКУ НА ОСНОВІ TIMEGAN У ХМАРНОМУ СЕРЕДОВИЩІ V. Kovalskyi, Ie. Tysh INTELLIGENT SYSTEM FOR PREDICTING NETWORK TRAFFIC ANOMALIES BASED ON TIMEGAN IN A CLOUD ENVIRONMENT	117
В. Козачок, І. Баран ОГЛЯД МАТЕМАТИЧНИХ МОДЕЛЕЙ РЕОКАРДІОСИГНАЛІВ V. Kozachok, I. Baran A REVIEW OF MATHEMATICAL MODELS OF REHECARDIO SIGNALS	118
Р. Кондратюк, Є. Тиш OPENCV ЯК ОСНОВА СУЧАСНИХ СИСТЕМ КОМП'ЮТЕРНОГО ЗОРУ R. Kondratiuk, Ie. Tysh OPENCV AS THE BASIS OF MODERN COMPUTER VISION SYSTEMS	119
Р. Кондратюк, Є. Тиш МАШИННИЙ ЗІР: СУТНІСТЬ ТЕХНОЛОГІЇ, ПРИНЦИПИ РОБОТИ ТА СФЕРИ ЗАСТОСУВАННЯ R. Kondratiuk, Ie. Tysh MACHINE VISION: THE ESSENCE OF THE TECHNOLOGY, PRINCIPLES OF OPERATION AND AREAS OF APPLICATION	120
М. Конотопський, Ю. Лещиншин ОЦІНКА ЕФЕКТИВНОСТІ РОБОТИ ПРОГРАМНО-АПАРАТНОГО КОМПЛЕКСУ КОНТРОЛЮ ЗА ТЕПЛОПОСТАЧАННЯМ З УРАХУВАННЯМ ПОГОДНИХ УМОВ M. Konotopskyi, Y. Leshchyshyn ASSESSMENT OF THE EFFICIENCY OF THE SOFTWARE AND HARDWARE COMPLEX FOR HEAT SUPPLY CONTROL CONSIDERING WEATHER CONDITIONS	121
Б. Котельніков, І. Козбур, Г. Козбур ДОСЛІДЖЕННЯ СУЧАСНИХ ЗАСОБІВ АВТОМАТИЗАЦІЇ ДЛЯ ПРОВЕДЕННЯ ВІДЕОКОНФЕРЕНЦІЙ	

УДК 004.932

О. Кобрин; Н. Стадник, к. т. н.

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

ЕКСПЕРИМЕНТАЛЬНА ОЦІНКА КОМП'ЮТЕРНО-ЗОРОВОЇ СИСТЕМИ ВИЯВЛЕННЯ ВІДХИЛЕНЬ У ПРОЦЕСІ FDM-ДРУКУ

UDC 004.932

O. Kobryn; N. Stadnyk, PhD.

EXPERIMENTAL EVALUATION OF A COMPUTER VISION SYSTEM FOR DETECTING DEVIATIONS IN THE FDM PRINTING PROCESS

Робота присвячена експериментальній перевірці комп'ютерно-зорової системи контролю процесу FDM-друку, яка аналізує відеопотік з камери в зоні робочого поля 3D-принтера та класифікує поточний стан друку на кілька типових класів: «нормальний друк», «спагеті», «відлипання моделі», «дефект екструзії» тощо. Основна увага зосереджена на оцінюванні точності виявлення відхилень та впливу структури навчальних даних на результати роботи моделі.

Навчальні вибірки сформовано на основі зображень і кадрів відео FDM-друку, отриманих з різних відкритих ресурсів та власних записів. До датасету увійшли приклади стабільного друку, виражених дефектів та кадри з ранніми проявами відхилень. Проведено дві серії навчання: у першій використано базовий набір із фіксованим ракурсом камери та стандартним масштабуванням зображення до 640×640 пікселів; у другій – розширений датасет з додатковими прикладами початкових стадій дефектів, варіаціями ракурсу й освітлення та уточненим кадруванням робочої зони.

Якість роботи системи оцінювали за показниками mAP@0.5, precision, recall і матрицями неточностей. У першій серії навчання модель продемонструвала інтегральний показник mAP@0.5 на рівні близько 0,8; найвища точність спостерігалася для класів «нормальний друк» та виражених дефектів типу «спагеті», тоді як ранні стадії відлипання моделі від платформи іноді помилково відносилися до нормального стану, що відображалася у характерній плутанині між «пограничними» класами.

У другій серії, після розширення вибірки та корекції схеми попередньої обробки (масштабування й вибір області інтересу), mAP@0.5 зросла до значення порядку 0,85–0,9. Одночасно зменшилася кількість хибних спрацювань і пропусків дефектів: зросла частка правильних класифікацій для класів, пов'язаних із відлипанням моделі й дефектами екструзії, а діагональні елементи матриці неточностей стали більш «щільними» для основних класів, що свідчить про покращення чутливості й селективності моделі щодо відхилень у процесі друку.

Отримані результати підтверджують, що ключовим чинником підвищення ефективності комп'ютерно-зорової системи є різноманітність і репрезентативність навчальних даних, зокрема наявність кадрів із проміжними станами між нормальним друком і сформованим дефектом. Розширення датасету та правильний вибір параметрів обробки зображення дозволяють суттєво покращити показники mAP, precision і recall без помітної втрати швидкодії та демонструють перспективність інтеграції такої системи в практичні засоби віддаленого моніторингу FDM-друку.

Література

1. Cao, M., Fu, L., Ai, F., Zhou, K. An Improved 3D Printing Extrusion Defect Detection Method Based On YOLO-v8. Research Square, 2024. 5-10 ст. Електронний ресурс: <https://www.researchgate.net/publication/378727971>

А. Станько, С. Козак, С. Кульчицький ЗАХИЩЕНА ТЕЛЕМЕТРІЯ В ІОТ: ПОРІВНЯЛЬНИЙ АНАЛІЗ MQTT OVER TLS, HTTPS ТА COAP/DTLS ДЛЯ ДОМАШНІХ І ПРОМИСЛОВИХ ЗАСТОСУВАНЬ A. Stanko, S. Kozak, S. Kulchytskyi SECURE TELEMETRY IN THE IOT: A COMPARATIVE ANALYSIS OF MQTT OVER TLS, HTTPS AND COAP/DTLS FOR HOME AND INDUSTRIAL APPLICATIONS	214
М. Трембач, Р. Павелко, Д. Биць ІОТ-СИСТЕМА БЕЗПЕЧНОГО МОНІТОРИНГУ ТА КЕРУВАННЯ ДОМАШНІМИ ПРИСТРОЯМИ З ПІДТРИМКОЮ OPENHAB M. Trembach, R. Pavelko, D. Byts IOT SYSTEM FOR SECURE MONITORING AND CONTROL OF HOME DEVICES WITH OPENHAB SUPPORT	215
В. Яцишин, М. Кіт СИСТЕМНИЙ ПІДХІД У ФОРМУВАННІ КОМАНДИ ДЛЯ ВИКОНАННЯ ІТ-ПРОЄКТІВ V. Yatsyshyn, M. Kit SYSTEM-BASED APPROACH TO IT PROJECT TEAM FORMATION	216
В. Яцишин, М. Кіт ФАКТОРИ ВПЛИВУ НА ЕФЕКТИВНІСТЬ ВИКОНАННЯ ІТ-ПРОЄКТІВ V. Yatsyshyn, M. Kit KEY FACTORS INFLUENCING IT PROJECT PERFORMANCE	217
Я. Гриневич, Я. Шкіра АНАЛІЗ АВТОМАТИЗОВАНИХ МЕТОДІВ УПРАВЛІННЯ МЕТАЛУРГІЙНИМ ВИРОБНИЦТВОМ НА ОСНОВІ ВИКОРИСТАННЯ MES-СИСТЕМ Ya. Hrynevych, Ya. Shkira ANALYSIS OF AUTOMATED METHODS OF METALLURGICAL PRODUCTION MANAGEMENT BASED ON THE USE OF MES SYSTEMS	218
Н. Вітвіцький Н. МОДЕЛЮВАННЯ ТА РЕАЛІЗАЦІЯ БАГАТОРІВНЕВОЇ СИСТЕМИ БЕЗПЕКИ ВЕБ-РЕСУРСУ N. Vitvitskyi MODELING AND IMPLEMENTATION OF A MULTI-LAYER WEB RESOURCE SECURITY SYSTEM	220
Н. Гашин, К. Макаричева ПРОГНОЗУВАННЯ ЕКОНОМІЧНИХ ПОКАЗНИКІВ ТРАНСПОРТНИХ ПЕРЕВЕЗЕНЬ З ВИКОРИСТАННЯМ ЕЛЕКТРОННИХ ТАБЛИЦЬ MS EXCEL N. Gashchyn, K. Makarycheva FORECASTING ECONOMIC INDICATORS OF TRANSPORTATION USING MS EXCEL SPREADSHEETS	221
О. Кобрин, Н. Стадник КОМП'ЮТЕРНО-ЗОРОВА СИСТЕМА КОНТРОЛЮ СТАБІЛЬНОСТІ ПРОЦЕСУ FDM-ДРУКУ O. Kobryn, N. Stadnyk COMPUTER VISION SYSTEM FOR STABILITY MONITORING OF THE FDM PRINTING PROCESS	222
В. Гайдук, О. Печіль, В. Ясний ОЧИЩА ВЕЖА В СКЛАДІ ЕЛЕВАТОРНОГО КОМПЛЕКСУ V. Haiduk, O. Pechiel, V. Yasniy PROJECT OF A PURIFICATION TOWER AS PART OF AN ELEVATOR COMPLEX	223
ЗМІСТ	225

УДК 004.932

О.В. Кобрин, Н.Б. Стадник, к.т.н.

Тернопільський національний технічний університет імені Івана Пулюя, Україна

КОМП'ЮТЕРНО-ЗОРОВА СИСТЕМА КОНТРОЛЮ СТАБІЛЬНОСТІ ПРОЦЕСУ FDM-ДРУКУ

O.V. Kobryn, N.B. Stadnyk, PhD.

COMPUTER VISION SYSTEM FOR STABILITY MONITORING OF THE FDM PRINTING PROCESS

Технології адитивного виробництва активно застосовуються у прототипуванні, технічній освіті та малосерійному виготовленні виробів. Ефективність FDM-3D-друку значною мірою залежить від стабільності процесу формування шарів, точності переміщення та сталості параметрів екструзії. У разі виникнення відхилень під час друку якість моделі суттєво погіршується, а сам процес може продовжуватися без візуального контролю з боку оператора, що призводить до значних втрат часу та матеріалу.

Одним із перспективних напрямів удосконалення адитивних систем є застосування методів комп'ютерного зору для виявлення помилок та нестабільностей у процесі друку. Використання алгоритмів аналізу зображень та глибокого навчання дає змогу визначати аномальні ситуації на основі аналізу відеопотоку, фіксуючи зміни у поведінці інструмента, структурі шару чи загальній динаміці процесу. Такий підхід не потребує додаткових апаратних сенсорів і може бути реалізований з використанням доступних камер.

Для забезпечення надійності аналізу використовувалися зображення та відео з різних відкритих джерел, а також матеріали, отримані у різних умовах зйомки, включаючи добре освітлені приміщення. Це дозволило охопити широкий спектр ситуацій, що підвищує здатність алгоритмів працювати в реальних умовах експлуатації різних моделей принтерів і матеріалів.

Методи обробки відеопотоку включають оцінювання структури сформованих шарів, аналіз послідовності кадрів, відстеження стабільності руху екструдера та визначення ознак відхилення від нормального режиму роботи. Комбінування таких методів дає змогу виявляти появу некоректних ситуацій значно раніше, ніж це може зробити оператор під час візуального контролю.

Розроблення систем виявлення помилок на основі аналізу процесу друку створює передумови для підвищення надійності адитивного виробництва та зменшення кількості бракованих виробів. Подальші дослідження можуть бути спрямовані на покращення точності аналізу, зменшення залежності від зовнішніх умов освітлення та оптимізацію алгоритмів для роботи в реальному часі на локальних обчислювальних пристроях.

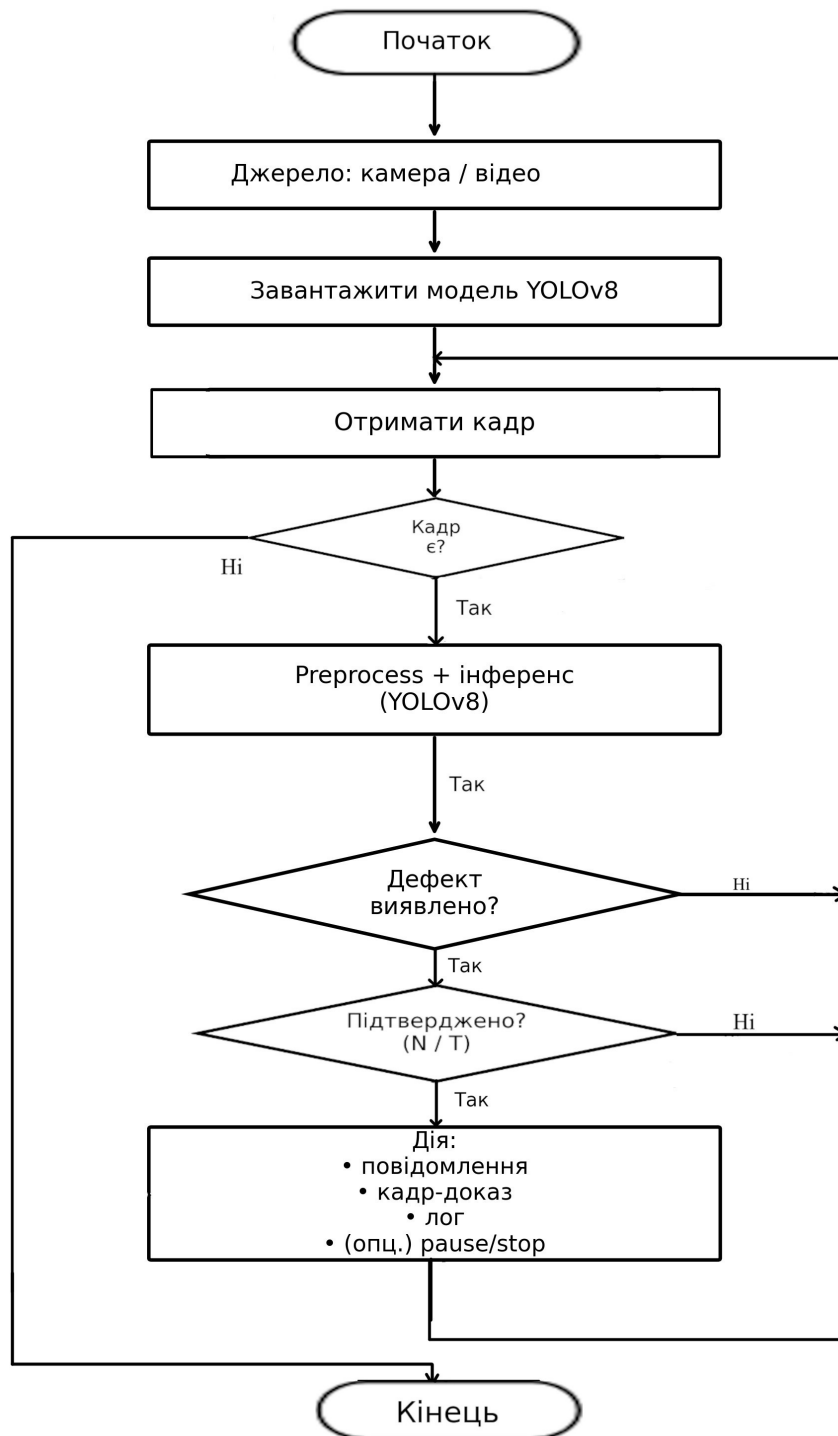
Література

1. Zhang, Y., Bernard, A., Harik, R. A framework for real-time monitoring of FDM additive manufacturing processes using computer vision. The International Journal of Advanced Manufacturing Technology, 2019. 10-11 ст.

Додаток Б

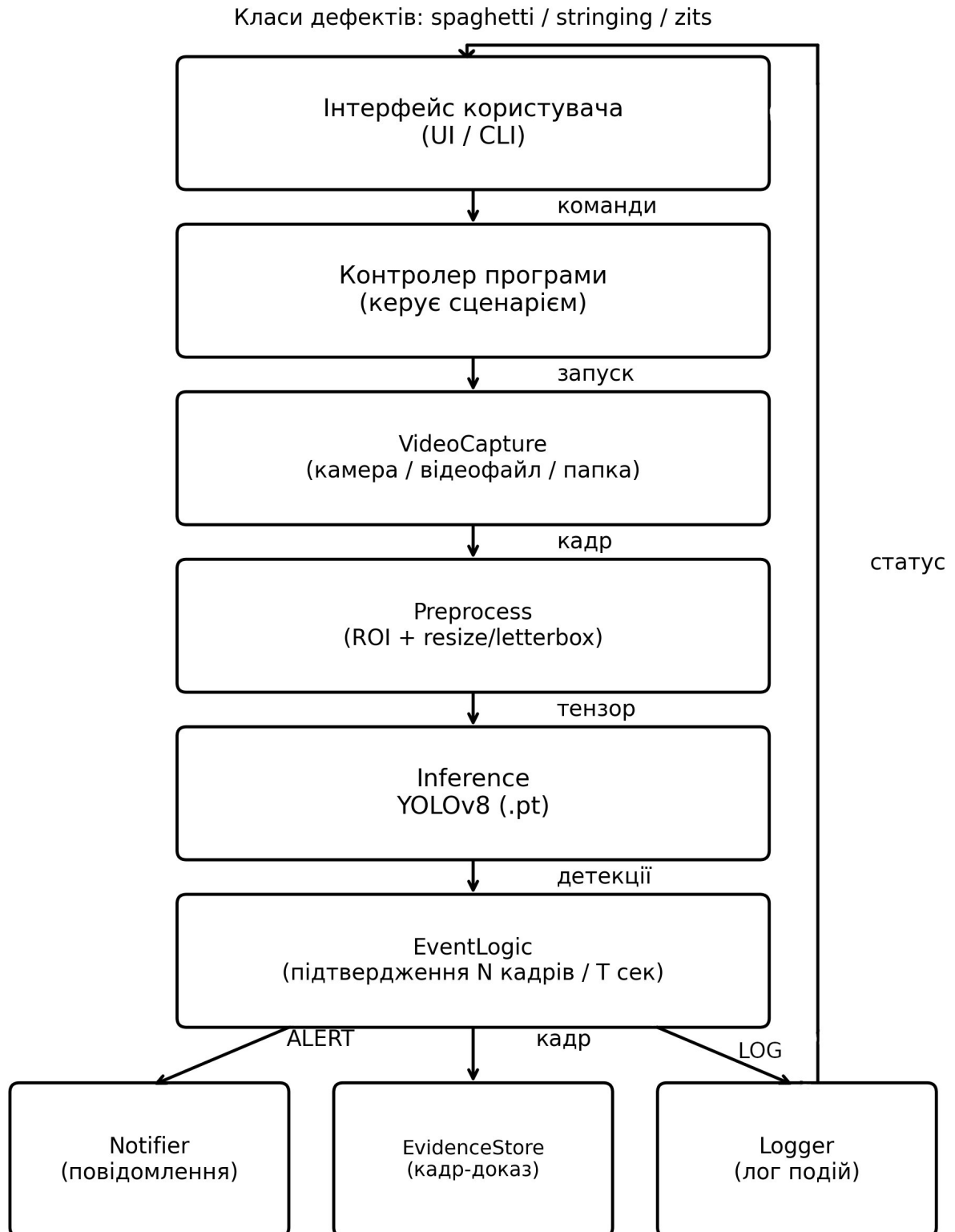
Блок схема алгоритму роботи системи

Класи дефектів: spaghetti / stringing / zits



Додаток В

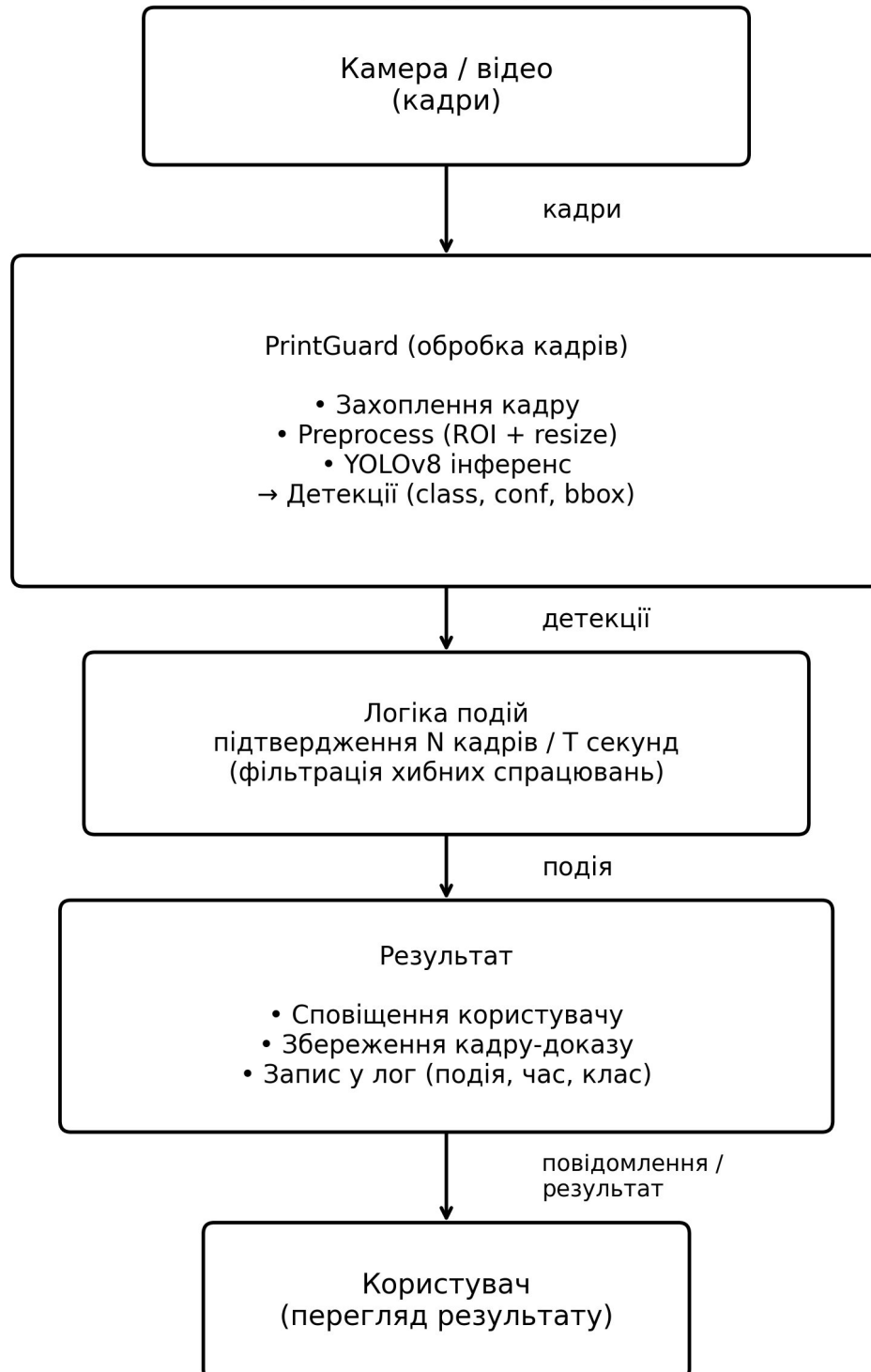
Схема взаємодії модулів програмного забезпечення



Додаток Г

Структурна схема системи

Класи дефектів: spaghetti / stringing / zits



Додаток Д

Лістинг програми

Main.py

```
import sys
import os
import cv2
import numpy as np
import time
import json
import requests
import winsound
from datetime import datetime
from PyQt6.QtWidgets import (QApplication, QMainWindow, QWidget, QVBoxLayout,
                             QHBoxLayout, QLabel, QPushButton, QLineEdit,
                             QTextEdit, QFrame, QComboBox, QSpinBox,
                             QScrollArea, QCheckBox, QDoubleSpinBox,
                             QSystemTrayIcon, QStyle, QSizePolicy, QFileDialog)
from PyQt6.QtCore import pyqtSignal, pyqtSlot, Qt, QThread
from PyQt6.QtGui import QPixmap, QImage
from printer_api import OctoPrintClient

STYLESHEET = """
QMainWindow { background-color: #050505; }
QLabel#VideoScreen { background-color: #000; border: 1px solid #333; border-
bottom: 2px solid #ff9100; }
QScrollArea#DeckScroll { background-color: #111; border: none; border-top:
1px solid #222; }
QWidget#DeckContainer { background-color: #111; }
QFrame.ControlModule { background-color: #161616; border: 1px solid #333;
border-radius: 6px; padding: 5px; }
QLabel.ModuleTitle { color: #666; font-size: 10px; font-weight: bold; text-
transform: uppercase; letter-spacing: 1px; }
QLabel { color: #aaa; font-family: 'Segoe UI'; font-size: 11px; }
QPushButton { background-color: #222; color: #ddd; border: 1px solid #333;
border-radius: 4px; padding: 6px; }
QPushButton:hover { background-color: #333; border-color: #555; }
```

```

QPushButton#PrimaryBtn { background-color: rgba(255, 145, 0, 0.1); color:
#ff9100; border: 1px solid #ff9100; }
QPushButton#PrimaryBtn:hover { background-color: #ff9100; color: #000; font-
weight: bold; }
QPushButton#DangerBtn { color: #ff2a2a; border: 1px solid #ff2a2a; background:
transparent; }
QPushButton#DangerBtn:hover { background-color: #ff2a2a; color: #000; }
QPushButton#AiToggle:checked { background-color: #00e676; color: #000;
border: none; font-weight: bold; }
QLineEdit, QSpinBox, QComboBox, QDoubleSpinBox {
    background-color: #0a0a0a; border: 1px solid #333; color: #ff9100; padding:
4px; font-family: 'Consolas';
}
QTextEdit { background-color: #000; border: 1px solid #222; color: #ff9100;
font-family: 'Consolas'; font-size: 10px; }
"""

# --- SETTINGS MANAGER ---
class SettingsManager:
    FILE_NAME = "settings.json"
    @staticmethod
    def save(data):
        try:
            with open(SettingsManager.FILE_NAME, 'w') as f:
                json.dump(data, f, indent=4)
        except Exception as e:
            print(f"Error saving settings: {e}")
    @staticmethod
    def load():
        if os.path.exists(SettingsManager.FILE_NAME):
            try:
                with open(SettingsManager.FILE_NAME, 'r') as f:
                    return json.load(f)
            except:
                return {}
        return {}

# --- NOTIFICATION MANAGER ---
class NotificationManager:
    @staticmethod
    def play_siren():

```

```

        try:
            winsound.Beep(1000, 300)
            winsound.Beep(1500, 300)
        except:
            pass

    @staticmethod
    def send_ntfy(topic_name, message, image_path=None):
        if not topic_name:
            return False, "No Topic"
        url = f"https://ntfy.sh/{topic_name}"
        try:
            headers = {"Title": "PrintGuardian", "Priority": "high", "Tags":
"warning"}
            if image_path and os.path.exists(image_path):
                headers["Filename"] = "evidence.jpg"
                headers["Message"] = message.encode('utf-8')
                with open(image_path, "rb") as f:
                    resp = requests.put(url, data=f, headers=headers)
            else:
                resp = requests.post(url, data=message.encode('utf-8'),
headers=headers)
            if resp.status_code == 200:
                return True, "OK"
            else:
                return False, f"Status Code: {resp.status_code}, {resp.text}"
        except Exception as e:
            return False, str(e)

    @staticmethod
    def log_to_file(message):
        t = datetime.now().strftime("%Y-%m-%d %H:%M:%S")
        if not os.path.exists("logs"): os.makedirs("logs")
        try:
            with open(f"logs/log_{datetime.now().strftime('%Y-%m-%d')}.txt",
"a", encoding='utf-8') as f:
                f.write(f"[{t}] {message}\n")
        except:
            pass

class VideoThread(QThread):
    change_pixmap_signal = pyqtSignal(np.ndarray)

```

```

alert_signal = pyqtSignal(str, str)
status_signal = pyqtSignal(str)
def __init__(self):
    super().__init__()
    self._run_flag = False
    self.enable_ai = False
    self.video_source = 0
    self.model = None
    self.conf_threshold = 0.50
    self.trigger_limit = 20
    self.defect_counter = 0
    self.action_mode = "PAUSE"
    self.is_simulation = False
    self.cooldown_duration = 60
    self.save_evidence = True
def load_model(self):
    if self.model is None:
        from ultralytics import YOLO
        if os.path.exists("best.pt"):
            try:
                self.model = YOLO("best.pt")
                self.status_signal.emit("SYS: AI LOADED")
            except:
                self.status_signal.emit("ERR: AI LOAD FAIL")
        else:
            self.status_signal.emit("WRN: NO 'best.pt'")
def update_params(self, conf, action, sim, cooldown, save):
    self.conf_threshold = conf
    self.action_mode = action
    self.is_simulation = sim
    self.cooldown_duration = cooldown
    self.save_evidence = save
def set_source(self, source):
    self.video_source = source
def run(self):
    self._run_flag = True
    self.load_model()

    if not os.path.exists("defects"): os.makedirs("defects")

```

```

cap = cv2.VideoCapture(self.video_source)
is_live_cam = isinstance(self.video_source, int)
if is_live_cam:
    cap.set(3, 1280)
    cap.set(4, 720)
if not cap.isOpened():
    self.status_signal.emit("ERR: SOURCE FAIL")
    self._run_flag = False
    return
self.status_signal.emit("SYS: ONLINE")
last_alert_time = 0
while self._run_flag:
    ret, frame = cap.read()
    if not ret:
        if not is_live_cam: self.status_signal.emit("SYS: VIDEO END")
        else: self.status_signal.emit("ERR: STREAM LOST")
        break
    if ret:
        final = frame
        detected = False

        if self.enable_ai and self.model:
            results = self.model.predict(frame, verbose=False,
            conf=self.conf_threshold, imsz=640)
            detected = len(results[0].boxes) > 0
            final = results[0].plot()
            if detected:
                self.defect_counter += 1
                cv2.rectangle(final, (0,0), (final.shape[1], 30),
(0,0,255), -1)
                cv2.putText(final, f"ANOMALY
[{self.defect_counter}/{self.trigger_limit}]",
(20, 20), cv2.FONT_HERSHEY_SIMPLEX, 0.7,
(255, 255, 255), 2)
                current_time = time.time()
                if self.defect_counter >= self.trigger_limit:
                    if current_time - last_alert_time >
self.cooldown_duration:
                        img_path = None

```

```

        t_stamp = datetime.now().strftime("%Y%m%d_%H%M%S")
        img_path = f"defects/alert_{t_stamp}.jpg"
        cv2.imwrite(img_path, frame)
        self.alert_signal.emit(self.action_mode, img_path)
        last_alert_time = current_time
        self.defect_counter = 0
    else:
        if self.defect_counter > 0: self.defect_counter -= 1
        self.change_pixmap_signal.emit(final)
        time.sleep(0.03)
    cap.release()
    self.status_signal.emit("SYS: STOPPED")
def stop(self):
    self._run_flag = False
    self.wait()
# --- MAIN WINDOW ---
class MainWindow(QMainWindow):
    def __init__(self):
        super().__init__()
        self.setWindowTitle("PrintGuardian // OCTOPRINT EDITION")
        self.setMinimumSize(1024, 750)
        self.setStyleSheet(STYLESHEET)
        self.tray = QSystemTrayIcon(self)
self.tray.setIcon(self.style().standardIcon(QStyle.StandardPixmap.SP_ComputerIcon))

        self.tray.show()
        self.thread = VideoThread()
        self.thread.change_pixmap_signal.connect(self.update_image)
        self.thread.status_signal.connect(self.log)
        self.thread.alert_signal.connect(self.handle_automation)
        self.client = None
        central = QWidget()
        self.setCentralWidget(central)
        layout = QVBoxLayout(central)
        layout.setContentsMargins(0, 0, 0, 0)
        layout.setSpacing(0)
        # Video
        self.video_label = QLabel("SYSTEM INITIALIZING...")
        self.video_label.setObjectName("VideoScreen")

```

```

        self.video_label.setAlignment(Qt.AlignmentFlag.AlignCenter)
        self.video_label.setSizePolicy(QSizePolicy.Policy.Expanding,
QSizePolicy.Policy.Expanding)
        layout.addWidget(self.video_label, stretch=1)
        # Deck
        scroll = QScrollArea()
        scroll.setObjectName("DeckScroll")
        scroll.setFixedHeight(230)
        scroll.setWidgetResizable(True)

scroll.setVerticalScrollBarPolicy(Qt.ScrollBarPolicy.ScrollBarAlwaysOff)
        deck = QWidget()
        deck.setObjectName("DeckContainer")
        deck_layout = QHBoxLayout(deck)
        deck_layout.setContentsMargins(10, 10, 10, 10)
        deck_layout.setSpacing(10)
        self.create_input_module(deck_layout)
        self.create_ai_module(deck_layout)
        self.create_automation_module(deck_layout)
        self.create_ntfy_module(deck_layout)
        self.create_sys_module(deck_layout)
        btn_exit = QPushButton("SAVE & EXIT")
        btn_exit.setObjectName("DangerBtn")
        btn_exit.setFixedWidth(70)
        btn_exit.clicked.connect(self.close)
        deck_layout.addWidget(btn_exit)
        deck_layout.addStretch()
        scroll.setWidget(deck)
        layout.addWidget(scroll)
        self.showMaximized()
        self.load_config()
        self.log("SYS: READY (OCTOPRINT CONFIG LOADED)")
def create_header(self, text):
    lbl = QLabel(text)
    lbl.setProperty("class", "ModuleTitle")
    lbl.setStyleSheet("font-weight:bold; color:#666;")
    return lbl
def create_input_module(self, layout):
    frame = QFrame()

```

```

frame.setProperty("class", "ControlModule")
vbox = QVBoxLayout(frame)
vbox.addWidget(self.create_header("SOURCE"))
self.combo_src = QComboBox()
self.combo_src.addItem(["USB Device", "RTSP Stream", "Video File
(Test)"])
self.combo_src.currentIndexChanged.connect(self.toggle_inputs)
self.spin_usb = QSpinBox()
self.spin_usb.setPrefix("ID: ")
self.input_rtsp = QLineEdit()
self.input_rtsp.setPlaceholderText("rtsp://...")
self.input_rtsp.setVisible(False)
self.file_layout = QHBoxLayout()
self.input_file = QLineEdit()
self.input_file.setPlaceholderText("path/to/video.mp4")
self.input_file.setReadOnly(True)
btn_browse = QPushButton("...")
btn_browse.setFixedWidth(30)
btn_browse.clicked.connect(self.browse_file)
self.file_layout.addWidget(self.input_file)
self.file_layout.addWidget(btn_browse)
self.input_file.setVisible(False)
btn_browse.setVisible(False)
btn = QPushButton("INIT / PLAY")
btn.setObjectName("PrimaryBtn")
btn.clicked.connect(self.start_camera)
vbox.addWidget(self.combo_src)
vbox.addWidget(self.spin_usb)
vbox.addWidget(self.input_rtsp)
vbox.addLayout(self.file_layout)
vbox.addWidget(btn)
vbox.addStretch()
layout.addWidget(frame)

def create_ai_module(self, layout):
    frame = QFrame()
    frame.setProperty("class", "ControlModule")
    vbox = QVBoxLayout(frame)
    vbox.addWidget(self.create_header("AI CORE"))

```

```

self.btn_ai = QPushButton("AI GUARD: OFF")
self.btn_ai.setObjectName("AiToggle")
self.btn_ai.setCheckable(True)
self.btn_ai.clicked.connect(self.toggle_ai)
self.spin_conf = QDoubleSpinBox()
self.spin_conf.setPrefix("CONF: ")
self.spin_conf.setRange(0.1, 1.0)
self.spin_conf.setValue(0.5)
self.spin_conf.valueChanged.connect(self.update_live_config)
self.chk_save = QCheckBox("SAVE HISTORY")
self.chk_save.setChecked(True)
self.chk_save.stateChanged.connect(self.update_live_config)
vbox.addWidget(self.btn_ai)
vbox.addWidget(self.spin_conf)
vbox.addWidget(self.chk_save)
vbox.addStretch()
layout.addWidget(frame)
def create_automation_module(self, layout):
    frame = QFrame()
    frame.setProperty("class", "ControlModule")
    frame.setMinimumWidth(160)
    vbox = QVBoxLayout(frame)
    vbox.addWidget(self.create_header("AUTOMATION"))
    vbox.addWidget(QLabel("ACTION:"))
    self.combo_action = QComboBox()
    self.combo_action.addItem(["PAUSE PRINT", "STOP PRINT", "NOTIFY
ONLY"])
    self.chk_sim = QCheckBox("SIMULATION")
    self.chk_sim.stateChanged.connect(self.update_live_config)
    self.spin_cool = QSpinBox()
    self.spin_cool.setPrefix("COOL: ")
    self.spin_cool.setSuffix("s")
    self.spin_cool.setRange(10, 600)
    self.spin_cool.setValue(60)
    self.spin_cool.valueChanged.connect(self.update_live_config)
    vbox.addWidget(self.combo_action)
    vbox.addWidget(self.chk_sim)
    vbox.addWidget(self.spin_cool)
    vbox.addStretch()

```

```

        layout.addWidget(frame)
def create_ntfy_module(self, layout):
    frame = QFrame()
    frame.setProperty("class", "ControlModule")
    frame.setMinimumWidth(200)
    vbox = QVBoxLayout(frame)
    vbox.addWidget(self.create_header("NTFY.SH + PHOTO"))
    self.input_topic = QLineEdit()
    self.input_topic.setPlaceholderText("Topic Name")
    self.chk_sound = QCheckBox("LOUD SIREN")
    self.chk_sound.setChecked(False)
    self.chk_sound.setStyleSheet("color: #ff9100; font-weight: bold;")
    btn_test = QPushButton("SEND TEST")
    btn_test.clicked.connect(self.test_notification)
    vbox.addWidget(self.input_topic)
    vbox.addWidget(self.chk_sound)
    vbox.addWidget(btn_test)
    vbox.addStretch()
    layout.addWidget(frame)
def create_sys_module(self, layout):
    frame = QFrame()
    frame.setProperty("class", "ControlModule")
    frame.setMinimumWidth(250)
    vbox = QVBoxLayout(frame)
    vbox.addWidget(self.create_header("OCTOPRINT SYSTEM"))
    h = QHBoxLayout()
    self.in_ip = QLineEdit("192.168.1.100")
    self.in_key = QLineEdit()
    self.in_key.setPlaceholderText("API Key")
    self.in_key.setEchoMode(QLineEdit.EchoMode.Password)
    h.addWidget(self.in_ip)
    h.addWidget(self.in_key)
    btn_conn = QPushButton("LINK PRINTER")
    btn_conn.clicked.connect(self.connect_printer)
    self.log_area = QTextEdit()
    self.log_area.setReadOnly(True)
    vbox.addLayout(h)
    vbox.addWidget(btn_conn)
    vbox.addWidget(self.log_area)

```

```

        layout.addWidget(frame)
def log(self, text):
    t = datetime.now().strftime("%H:%M:%S")
    self.log_area.append(f"[{t}] {text}")
    NotificationManager.log_to_file(text)
def browse_file(self):
    fname, _ = QFileDialog.getOpenFileName(self, "Open Video", "", "Video
Files (*.mp4 *.avi *.mkv)")
    if fname:
        self.input_file.setText(fname)
def toggle_inputs(self):
    idx = self.combo_src.currentIndex()
    self.spin_usb.setVisible(idx == 0)
    self.input_rtsp.setVisible(idx == 1)
    self.input_file.setVisible(idx == 2)
    self.file_layout.itemAt(1).widget().setVisible(idx == 2)
def start_camera(self):
    if self.thread.isRunning(): self.thread.stop()
    idx = self.combo_src.currentIndex()
    src = 0
    if idx == 0: src = self.spin_usb.value()
    elif idx == 1: src = self.input_rtsp.text()
    elif idx == 2:
        src = self.input_file.text()
        if not os.path.exists(src):
            self.log("ERR: File not found")
            return
        self.log(f"SYS: Loading video...")
    self.thread.set_source(src)
    self.thread.start()
    self.update_live_config()
def toggle_ai(self):
    state = self.btn_ai.isChecked()
    self.thread.enable_ai = state
    self.btn_ai.setText("AI GUARD: ON" if state else "AI GUARD: OFF")
    self.log(f"SYS: AI {'ENABLED' if state else 'DISABLED'}")
def update_live_config(self):
    conf = self.spin_conf.value()
    action = self.combo_action.currentText()

```

```

sim = self.chk_sim.isChecked()
cool = self.spin_cool.value()
save = self.chk_save.isChecked()
self.thread.update_params(conf, action, sim, cool, save)
def connect_printer(self):
    # ВИПРАВЛЕНО: Тільки 2 аргументи для OctoPrint
    self.client = OctoPrintClient(self.in_ip.text(), self.in_key.text())
    ok, msg = self.client.get_connection_status()
    if ok: self.log(f"NET: {msg}")
    else: self.log(f"ERR: {msg}")
def test_notification(self):
    topic = self.input_topic.text()
    ok, msg = NotificationManager.send_ntfy(topic, " Test Message")
    if ok: self.log("NET: Ntfy Sent")
    else: self.log(f"ERR: {msg}")
    if self.chk_sound.isChecked(): NotificationManager.play_siren()
def handle_automation(self, action_mode, img_path):
    msg = f"!!! CRITICAL DEFECT !!! Mode: {action_mode}"
    self.log(msg)
    self.tray.showMessage("CRITICAL ERROR", f"Detected! {action_mode}",
QSystemTrayIcon.MessageIcon.Critical, 5000)
    if self.chk_sound.isChecked():
        NotificationManager.play_siren()
    topic = self.input_topic.text()
    if topic:
        self.log("NET: Uploading evidence...")
        ok, err_msg = NotificationManager.send_ntfy(topic, f"△
PrintGuardian: {action_mode}", img_path)
        if ok:
            self.log("NET: Upload SUCCESS!")
        else:
            self.log(f"ERR: Upload FAILED ({err_msg})")
    if self.client and not self.chk_sim.isChecked():
        if "PAUSE" in action_mode:
            self.client.pause_print()
            self.log("CMD: OCTOPRINT PAUSED")
        elif "STOP" in action_mode:
            self.client.stop_print()
            self.log("CMD: OCTOPRINT STOPPED")

```

```

        elif self.chk_sim.isChecked():
            self.log("SIM: No printer action")
    @pyqtSlot(np.ndarray)
    def update_image(self, cv_img):
        rgb = cv2.cvtColor(cv_img, cv2.COLOR_BGR2RGB)
        h, w, ch = rgb.shape
        qt_img = QImage(rgb.data, w, h, ch*w, QImage.Format.Format_RGB888)
        scaled = qt_img.scaled(self.video_label.size(),
Qt.AspectRatioMode.KeepAspectRatio,
Qt.TransformationMode.SmoothTransformation)
        self.video_label.setPixmap(QPixmap.fromImage(scaled))
    def save_config(self):
        data = {
            "source_idx": self.combo_src.currentIndex(),
            "usb_id": self.spin_usb.value(),
            "rtsp_url": self.input_rtsp.text(),
            "file_path": self.input_file.text(),
            "conf": self.spin_conf.value(),
            "action": self.combo_action.currentIndex(),
            "sim_mode": self.chk_sim.isChecked(),
            "cooldown": self.spin_cool.value(),
            "topic": self.input_topic.text(),
            "siren": self.chk_sound.isChecked(),
            "printer_ip": self.in_ip.text(),
            "printer_key": self.in_key.text()
        }
        SettingsManager.save(data)
        self.log("SYS: Settings Saved")
    def load_config(self):
        data = SettingsManager.load()
        if not data: return
        try:
            self.combo_src.setCurrentIndex(data.get("source_idx", 0))
            self.spin_usb.setValue(data.get("usb_id", 0))
            self.input_rtsp.setText(data.get("rtsp_url", ""))
            self.input_file.setText(data.get("file_path", ""))
            self.spin_conf.setValue(data.get("conf", 0.5))
            self.combo_action.setCurrentIndex(data.get("action", 0))
            self.chk_sim.setChecked(data.get("sim_mode", False))

```

```

        self.spin_cool.setValue(data.get("cooldown", 60))
        self.input_topic.setText(data.get("topic", ""))
        self.chk_sound.setChecked(data.get("siren", False))
        self.in_ip.setText(data.get("printer_ip", "192.168.1.100"))
        self.in_key.setText(data.get("printer_key", ""))
        self.toggle_inputs()
    except Exception as e:
        print(f"Error loading settings: {e}")
    def closeEvent(self, event):
        self.save_config()
        self.thread.stop()
        event.accept()
if __name__ == "__main__":
    app = QApplication(sys.argv)
    window = MainWindow()
    window.show()
    sys.exit(app.exec())

```

Api.py

```
import requests
```

```
class OctoPrintClient:
```

```

    def __init__(self, ip_address, api_key):
        # Автоматично додаємо http://, якщо користувач забув
        clean_ip = ip_address.strip()
        if not clean_ip.startswith("http"):
            self.base_url = f"http://{clean_ip}/api"
        else:
            self.base_url = f"{clean_ip}/api"
        self.headers = {"X-API-Key": api_key.strip()}
    def get_connection_status(self):
        """Перевіряє зв'язок з OctoPrint"""
        try:
            # Запит на перевірку версії/статусу
            response = requests.get(f"{self.base_url}/connection",
headers=self.headers, timeout=3)
            if response.status_code == 200:

```

```

        data = response.json()
        state = data.get("current", {}).get("state", "Unknown")
        return True, f"OctoPrint Online: {state}"
    elif response.status_code == 403:
        return False, "Invalid API Key (Доступ заборонено)"
    else:
        return False, f"HTTP Error: {response.status_code}"
except Exception as e:
    return False, f"Connection Failed: {e}"
def pause_print(self):
    """Ставить друк на паузу"""
    print("OctoPrint: Sending PAUSE...")
    return self._post_job_command({"command": "pause", "action": "pause"})
def stop_print(self):
    """Скасовує друк"""
    print("OctoPrint: Sending CANCEL...")
    return self._post_job_command({"command": "cancel"})
def _post_job_command(self, payload):
    """Внутрішня функція відправки команд"""
    try:
        response = requests.post(f"{self.base_url}/job",
headers=self.headers, json=payload, timeout=5)
        if response.status_code == 204:
            return True
        else:
            print(f"OctoPrint Command Error: {response.text}")
            return False
    except Exception as e:
        print(f"Network Error: {e}")
        return False

```