

Міністерство освіти і науки України  
Харківський національний університет радіоелектроніки

Факультет Автоматики і комп'ютеризованих технологій

(повна назва)

Кафедра Комп'ютерно-інтегрованих технологій, автоматизації  
та мехатроніки

(повна назва)

## АТЕСТАЦІЙНА РОБОТА

### Пояснювальна записка

другий (магістерський)

(рівень вищої освіти)

Розробка методу ідентифікації деталей для процесу сортування на базі  
комп'ютерного зору

(тема)

Виконав:

студент 6 курсу, групи АУТПм-19-1

Калашников Михайло Юрійович

Спеціальності 151 Автоматизація та комп'ютерно-  
інтегровані технології

Освітня програма: Автоматизоване управління  
технологічними процесами

Тип програми : Освітньо професійна

Керівник проф. Євсєєв В.В.

Допускається до захисту  
Зав. кафедри КІТАМ

(підпис)

Невлюдов І.Ш.

(прізвище, ініціали)

2020 р.

# ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

Факультет Автоматики і комп'ютеризованих технологій  
Кафедра Комп'ютерно-інтегрованих технологій, автоматизації та мехатроніки  
Рівень вищої освіти другий (магістерський)  
Спеціальність 151 Автоматизація та комп'ютерно-інтегровані технології  
Освітня програма Автоматизоване управління технологічними процесами  
(шифр і назва)

ЗАТВЕРДЖУЮ:

Зав.кафедри КІТАМ\_  
Невлюдов І.Ш  
(підпис)  
«» 2020 р.

## ЗАВДАННЯ НА АТЕСТАЦІЙНУ РОБОТУ

студентові Калашикову Михайлу Юрійовичу  
(прізвище, ім'я, по батькові)

1. Тема роботи Розробка методу ідентифікації деталей для процесу сортування на базі комп'ютерного зору

Затверджена наказом по університету від 2 листопада 2020 р. №1510 Ст

2. Термін подання студентом роботи до екзаменаційної комісії 20.06.2020 р.

3. Вихідні дані до роботи Деталь типу корпус, середовище розробки PyCharm, максимальні розміри деталі 300 мм × 300 мм × 300 мм

4. Перелік питань, що потрібно опрацювати в роботі  
Вступ

1. Аналіз технологій, які використовуються на сучасному приладобудівному виробництві

2. Розробка методу ідентифікації деталей для процесу сортування на базі комп'ютерного зору

3. Створення програмного засобу для ідентифікації деталей для процесу сортування на базі комп'ютерного зору

4. Проведення експериментальних досліджень з розробленим програмним забезпеченням

5. Охорона праці

6. Висновки

Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій \_\_\_\_\_

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів) Графічний матеріал у вигляді презентації – 12 с. ф. А4

#### 6. Консультанти розділів роботи

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

### КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Аналіз технічного завдання	2.11.2020	виконано
2	Опрацювання літератури за темою роботи.	9.11.2020	виконано
3	Вибір мови програмування та засобів	10.11.2020	виконано
4	Робота над практичною частиною	18.11.2020	виконано
5	Проведення експериментів	22.11.2020	виконано
6	Оформлення пояснювальної записки	25.11.2020	виконано
7	Оформлення презентації	27.11.2020	виконано
8	Подання роботи на рецензію		
9	Подання роботи на підпис зав. кафедри		
10	Подання атестаційної роботи в ЕК		

Дата видачі завдання 11.05.2020 р.

Студент \_\_\_\_\_ Калашников М.Ю.  
(підпис)

Керівник роботи \_\_\_\_\_ проф. Євсєєв В.В.  
(підпис) (посада, прізвище, ініціали)

## РЕФЕРАТ

Пояснювальна записка: 80 с., 11 рис., 4 табл., 26 джер., 1 додат.

### ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ, ІДЕНТИФІКАЦІЯ, РОЗПІЗНАВАННЯ, КОМП'ЮТЕРНИЙ ЗІР, КЕРУВАННЯ, НАВЧАННЯ

З запуском Industry 4.0 компаніям важко залишатися на висоті без впровадження будь-якої інтелектуальної системи. Великі дані, що генеруються безліччю датчиків, вимагають складних систем, здатних отримувати корисну інформацію і приймати розумні рішення.

Предмет дослідження – методи розпізнання та ідентифікації деталей на виробництві.

Об'єкт дослідження – процес сортування деталей за допомогою системи комп'ютерного зору на приладобудівному виробництві.

Метою дослідження є розробка моделі та методу ідентифікації деталей на базі комп'ютерного зору для впровадження на конвеєрно виробничу лінію

Методи дослідження – математичні моделі та методи розроблені на базі моделі, параметрична модель.

Розробка здійснювалась із використанням середовища на IBM-сумісному персональному комп'ютері (тактова частота процесора 4x2,8 ГГц, обсяг оперативної пам'яті – 8 Гб, жорсткого диска – 1 ТБ).

## **ABSTRACT**

Master's certification work contains 62 pages, 11 figures, 4 tables, 26 sources, 1 appendix.

**SOFTWARE, IDENTIFICATION, RECOGNITION, COMPUTER VISION,  
KERUVANNA, TRAINING**

With the launch of Industry 4.0, it is important for companies to be able to work without the need for an intelligent system. Great talent, which generates without the need for sensors, can be used for folding systems, built-in information and intelligent solutions.

The object of research is intelligent instrument making.

The subject of further inspection is the process of sorting parts using a computer vision system in an instrument-making industry.

The aim of the study is to develop a method for identifying parts for the sorting process based on computer vision using neural network technologies.

Design Methods - Software Development Tools, Tensorflow Machine Learning Libraries, Keras, cvlib.

The development was carried out using the environment on an IBM-compatible personal computer (processor clock speed 4x2.8 GHz, RAM – 8 GB, hard disk – 1 TB).

## ЗМІСТ

Перелік скорочень і термінів .....	7
Вступ.....	8
1 Аналіз технологій, які використовуються на сучасному приладобудівному виробництві .....	10
1.1 Сучасне виробництво та технологія Industry 4.0 .....	11
1.2 Сучасні конвеєри та їх датчики .....	15
1.3 Процес сортування на конвеєрному виробництві .....	18
1.4 Використання технологій комп'ютерного та машинного зору для сортування деталей .....	19
1.5 Нейронні мережі та глибоке навчання .....	21
1.6 Інтелектуальне виробництво та глибоке навчання у концепції Industry 4.0.....	25
1.7 Постановка задач дослідження.....	26
1.8 Висновки до першого розділу.....	26
2 Розробка методу ідентифікації деталей для процесу сортування на базі комп'ютерного зору .....	27
2.1 Основні технології ідентифікації .....	27
2.2 Розробка моделі ідентифікації деталей для процесу сортування на базі комп'ютерного зору .....	35
2.3 Розробка методу ідентифікації деталей для процесу сортування на базі комп'ютерного зору .....	40
2.4 Висновки до другого розділу.....	45
3 Розробка програмного засобу для ідентифікації деталей для процесу сортування на базі комп'ютерного зору .....	46
3.1 Вибір і обґрунтування технологій для створення програмного забезпечення .....	46
3.2 Алгоритм роботи програми .....	47

3.3 Розробка програмного забезпечення .....	48
3.4 Висновки до третього розділу.....	52
4 Проведення експериментальних досліджень з розробленим програмним забезпеченням .....	53
4.1 Експеримент зі швидкості навчання класифікатору .....	53
4.2 Експеримент з ідентифікації деталей типу корпус для різного рівня освітлення.....	54
4.3 Експеримент з ідентифікації деталей типу корпус для різного куту повороту.....	56
4.4 Висновки до четвертого розділу.....	57
5 Охорона праці.....	58
5.1 Аналіз умов праці на робочому місці.....	58
5.2 Промислова безпека .....	61
Висновки.....	64
Перелік джерел посилання .....	65
Додаток А Демонстраційний графічний матеріал .....	69

## ПЕРЕЛІК СКОРОЧЕНЬ І ТЕРМІНІВ

БГП – блоки графічних процесорів  
ГВС – гнучкі виробничі системи  
ГН – глибоке навчання  
ДР – доповнена реальність  
ЗЕО – загальні показники ефективності  
ІоТ – інтернет речей  
ІЗ – інтелектуальний завод  
КІВ – комп'ютерно-інтегроване виробництво  
КПЕ – ключові показники ефективності  
КФС – кібер-фізична система  
НМ – нейронна мережа  
ЧПУ – числове програмне управління  
ІІ – штучний інтелект  
BigData – великі дані  
Cloud computing – хмарні розрахунки  
CUDA – Compute Unified Device Architecture  
GPU – графічні процесори  
IPP – Integrated Performance Primitives  
OpenCL – Open Computing Language

## ВСТУП

На сучасному автоматизованому виробництві, що вимагає промислова революція Industry 4.0, дуже актуальним питанням в наш час стає саме питання двох основних складових – грошей та часу. Для того, щоб прискорювати роботу великих виробництв, їм потрібно надавати такий параметр як гнучкість, тобто можливість швидкої переналадки в будь-який час під певний виріб. Операція сортування є основою будь-якого конвеєрного виробництва. Використання цієї операції вимагає великої точності та швидкості, щоб відповідати трендам сучасного виробництва. Одним з можливих варіантів є використання нейронних мереж для ідентифікації деталей завдяки їх швидкості і якості розпізнавання. Використання такого роду систем дасть можливість прискорити роботу будь-якої автоматизованої системи, що й свідчить про актуальність даної роботи.

Одним з напрямлень, що потрібно розвивати на сучасному виробництві в епоху Industry 4.0 є конвеєрні лінії, які служать для багатьох завдань, зокрема для переміщення та сортування деталей, що свідчить про актуальність теми.

Предмет дослідження – методи розпізнавання та ідентифікації деталей на виробництві.

Об'єкт дослідження – процес сортування деталей за допомогою системи комп'ютерного зору на приладобудівному виробництві.

Метою дослідження є розробка моделі та методу ідентифікації деталей на базі комп'ютерного зору для впровадження на конвеєрно виробничу лінію

Для досягнення мети атестаційної роботи магістра потрібно вирішити наступні питання:

- провести аналіз технологій, які використовуються на сучасному приладобудівному виробництві;
- розробити метод ідентифікації деталей для процесу сортування на базі комп'ютерного зору;

- на основі методу розробити модель ідентифікації деталей для процесу сортування на базі комп'ютерного зору;
- на основі методу та моделі розробити програмне забезпечення ідентифікації деталей для процесу сортування на базі комп'ютерного зору;
- провести експерименти з розпізнавання деталей типу корпус при різних початкових умовах;
- оформити пояснювальну записку згідно [1] та [2].

Дана атестаційна робота пройшла апробацію у збірнику ADED-2020(2).

# 1 АНАЛІЗ ТЕХНОЛОГІЙ, ЯКІ ВИКОРИСТОВУЮТЬСЯ НА СУЧАСНОМУ ПРИЛАДОБУДІВНОМУ ВИРОБНИЦТВІ

## 1.1 Сучасне виробництво та технологія Industry 4.0

Використовуючи автоматизацію, будь-яка організація може виконувати процеси практично без втручання людини, особливо це стосується сучасного приладобудівного виробництва. Завдяки автоматизації є можливість приводити в дію цілий ряд обладнання, яке потім може виконувати безліч завдань в широкому спектрі виробничих середовищ.

Автоматизація настільки ефективна, завдяки тому що вона збільшує якість, продуктивність і ефективність за рахунок зменшення участі людини, тим самим різко знижуючи ризик помилки та збільшуючи швидкість виконання поставлених завдань.

У своїй основній формі автоматизація використовує контролер, який оцінює вимірний стан за заздалегідь визначеним списком значень для підтримки бажаних умов і умов, необхідних для ефективності [3].

Автоматизація в промислових умовах використовує систему управління, таку як комп'ютер, наприклад, і величезні обсяги даних (Big Data) для управління обладнанням і процесами у виробничому середовищі. Компанії, що працюють в цій галузі, завжди шукають способи збільшення випуску, продуктивності і ефективності; автоматизація підтримує обладнання в певному стані, що може ідентифікуватися та проводити налаштування.

Більшість автоматизованих виробничих ліній складаються з робочих станцій і транспортувальних систем, які переміщує елемент виробництва (корпуси, деталі, тощо) через численні етапи виробництва, використовуючи безліч різних інструментів для зміни призначення або зовнішнього вигляду продукту.

Логічний контролер виконує цей процес, керуючи послідовністю, в якій використовується обладнання, і тим, як довго кожна машина повинна працювати з продуктом.

Підприємства можуть використовувати інфраструктури автоматизації для виробництва, очищення та виробництва окремих деталей, а також, при необхідності, для складання кінцевого продукту.

У наш час існує чотири основних типів автоматизації, кожен з яких служить визначеної мети:

- гнучкі виробничі системи (ГВС) – їх використання дозволяє виробничим лініям розширювати можливості програмованих систем, щоб забезпечити переналаштування цих систем без або з мінімальною затримкою у виробництві;

- програмована автоматизація дозволяє адаптувати і змінювати порядок виконання всього виробничого процесу з урахуванням відхилень в готовому продукті, таких як, наприклад, зміна кольору корпусу пристрою, тощо – для цього типу зазвичай використовуються верстати з числовим програмним управлінням (ЧПУ), які керовані комп'ютерними програмами для виробництва партій з будь-якими змінами;

- жорсткі, фіксовані або жорсткі типи – ці системи фіксуються протягом всього виробничого процесу і не можуть бути змінені без значних змін (використання такого роду систем зазвичай обмежується виробничими лініями, що виробляють окремі вироби у великих обсягах, наприклад, виробництво автомобілів);

- комп'ютерно-інтегроване виробництво (КІВ) або комп'ютерні інтегровані виробничі системи охоплюють всі виробничі процеси, пов'язані з автоматизацією і виробництвом, які покладаються на комп'ютерну систему [4].

Системи КІВ найчастіше включають такі елементи:

- автоматичні крани і транспортні системи;
- верстати з ЧПУ;

- інтеграція CAD і CAM;
- комп'ютерне планування;
- автоматизоване планування і виробництво;
- машинні системи;
- робототехніка.

Industry 4.0, також відома як четверта промислова революція або просто інтелектуальне виробництво являє собою процес використання автоматизації і наборів даних у всіх сферах виробництва. Цей процес є інтуїтивно зрозумілим і взаємопов'язаним і дозволяє виробничим лініям відповідати вимогам постійним змінам в тій галузі, в якій використовується.

В рамках використання Industry 4.0, будь-які активності тепер можуть використовуватись для зв'язку з будь-яким іншим процесом, підвищуючи швидкість і точність, тим самим підвищуючи цінність всієї операції.

Industry 4.0 складається з 9 основних напрямків (рис. 1.1)

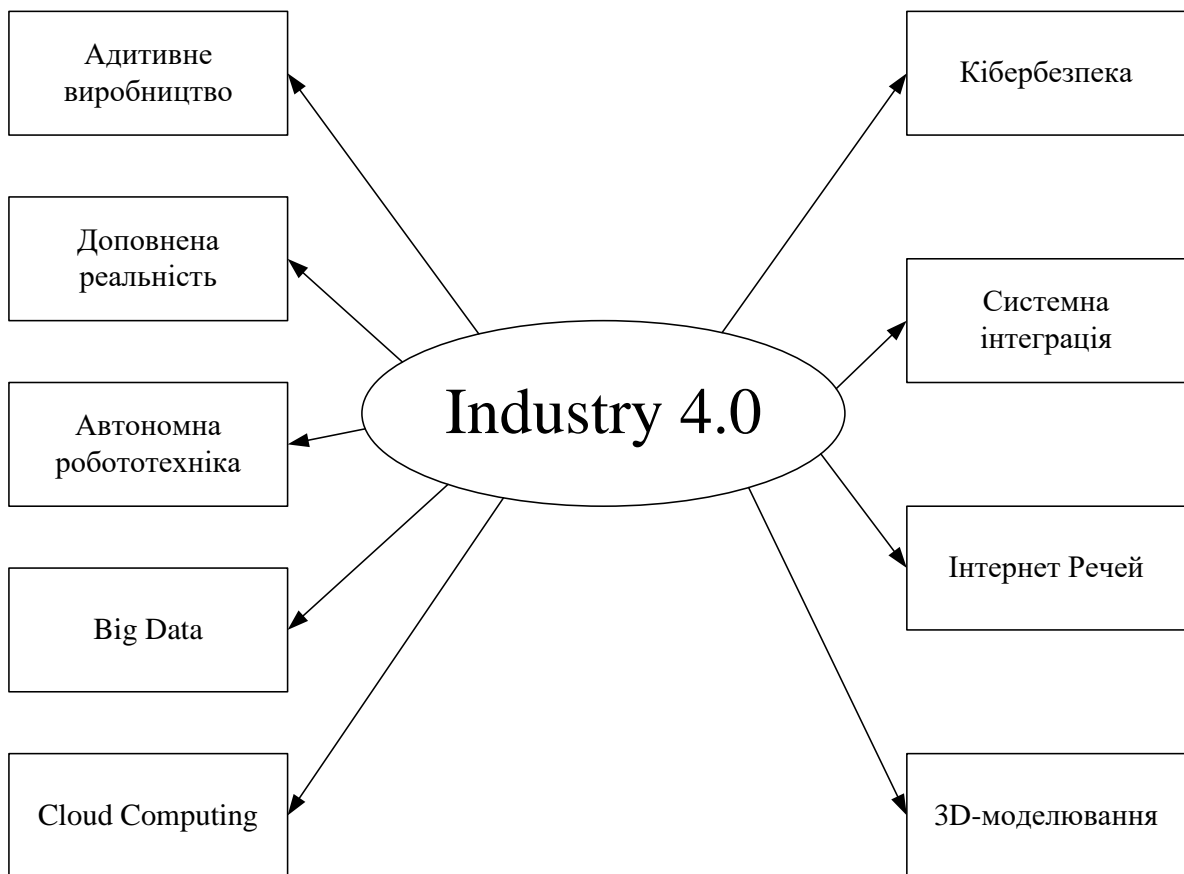


Рисунок 1.1 – Основні напрямки Industry 4.0

– адитивне виробництво – це один термінів, який є основним напрямом для 3D-друку, створення продуктів за допомогою якого, використовує 3D-технологій шляхом створення шарів, що утворюють весь об'єкт; також ця технологія зазвичай використовується для невеликих партій нестандартної продукції, легких деталей машин і прототипів;

– доповнена реальність (ДР, AR) – ідеально підходить для виявлення проблем в системі, а також для цілей навчання і діагностики;

– автономна робототехніка може змінювати свої основні цілі в залежності від того, на якому етапі перебуває виробництво – її використовують для безпечної роботи в середовищі, де може бути присутня людина, також роботи можуть спілкуватися один з одним;

– великі дані (Big Data) і аналітика – використовують для підвищення ефективності великого виробництва, де потрібні великі обсяги даних і аналітичних ресурсів, тобто навіть виробничі лінії можуть збирати дані на кожному етапі, щоб поліпшити свої процеси і перерозподілити ресурси там, де вони вважають за потрібне;

– хмарні розрахунки (Cloud computing) – виробництво вимагає співпраці, що виходить за межі приміщень компанії, а це означає, що швидкі хмарні обчислення необхідні для збору, аналізу, зберігання і моніторингу даних;

– кібербезпека – Industry 4.0 спрямована на відхід від традиційно закритих процесів в більш сучасну і взаємопов'язану середу, однак для цього потрібні більш складні рівні доступу користувачів і велика залежність від безпеки в мережах;

– горизонтальна і вертикальна системна інтеграція – можливість слідкувати за кожним активом процесу виробництва, тобто працівник виробничої лінії може стежити за іншими процесами виробництва;

– інтернет речей (IoT) – завдяки використанню різного роду датчиків є можливість обмінюватися даними між будь-якими зонами виробництва, щоб

дати виробничому персоналу можливість глибше зрозуміти роботу виробничої лінії;

– 3D-моделювання продуктів, матеріалів і процедур може використовувати дані в реальному часі, які потім використовуються для перетворення їх у віртуальну модель всього виробничого процесу.

Інтелектуальний завод (ІЗ) поєднує інтелектуальні фізичні пристрої за допомогою фізичних пристроїв промислового Інтернету. Інтелектуальні пристрої можуть виконувати обчислення, зв'язок, точне та віддалене управління, а також працювати автономно. В майбутньому нам буде потрібно отримати індивідуальну інформацію про клієнта, потім віртуальна фабрика, заснована на хмарному проектуванні і виробництві, буде виконувати проектування і моделювання [5].

Інтелектуальні пристрої можуть аналізувати інформацію після її передачі. На основі неї, інтелектуальні пристрої можуть виконувати виробничі, випробувальні та інші дії.

Інтелектуальна фабрика взагалі – це підприємство, засноване на даних. Щоденна робота інтелектуальної фабрики в основному залежить від наступного програмного і апаратного забезпечення: галузеві мережі, галузева мережева безпека, галузеві великі дані, платформа хмарних обчислень, система MES, віртуальна реальність, RFID, 3D-друк, машина. бачення, інтелектуальна логістика і так далі. Обладнання включає в себе промислових роботів, збір даних, промислові комутатори [6].

Оскільки будь-яка сфера промисловості стикається з великою кількістю проблем, таких як низька ефективність використання ресурсів, сильний тиск навколишнього середовища, висока вартість робочої сили, інтелектуальне виробництво є неминучою тенденцією в промисловості майбутнього. Industry 4.0 має такі переваги, як висока ефективність розподілу ресурсів, швидке реагування на ринковий попит, низькі витрати на робочу силу і логістику. З цього сказаного раніше можна зробити висновок, що використання Industry 4.0 у сфері виробництва в Україні є актуальною темою.

## 1.2 Сучасні конвеєри та їх датчики

Конвеєрні системи лежать в основі Industry 4.0 у виробничому середовищі майже для всіх сфер – всі форми автоматизації, ефективності оптимізації та продуктивності є основою цієї нової парадигми виробництва. Щоб конкурувати в світі, що розвивається в епоху Industry 4.0, конвеєрна компанія повинна створювати рішення, які можна інтегрувати в ці складні середовища [7].

Як ключовий елемент підйомно-транспортного обладнання, конвеєри повинні працювати безвідмовно та швидко і взаємодіяти з великою кількістю інших рішень для вантажно-розвантажувальних робіт, з яких складається сучасне виробництво. Використання таких рішень для Industry 4.0 можна побачити в їх прогнозованому зростанні на світовому ринку майже на 200 млрд доларів до 2025 року.

Статистика цього прогнозу тільки натякає на важливість конвеєра в сучасному виробництві, де провідні конвеєрні компанії створюють індивідуальні рішення для цих середовищ.

Конвеєр являє собою елемент з'єднання між всіма елементами виробництва, а тому він повинен працювати безперебійно і безперервно. У той час як технології автоматизації забезпечують швидкі, безпечні та ефективні виробничі лінії, конвеєри надають рух автоматизації.

Сучасні технології автоматизації, такі як датчики Інтернету речей (IoT), RFID і робототехніка, а також інші технології, покращують продуктивність, відстеження та ефективність виробництва в рамках Industry 4.0. Всі ці технології покладаються на передачу і обмін даними між конвеєрами на виробництва та для упаковки, щоб забезпечити безперервний рух у виробничому середовищі.

Більшість виробничих процесів покладаються на конвеєри для завершення процесу передачі даних, щоб виробництво тривало.

Розумні підприємства повинні знайти правильну конвеєрну систему, яка інтегрується в складні інтелектуальні системи автоматизації. Виробники можуть побачити це через призму датчиків, елементів управління, RFID та інших пристроїв зв'язку і передачі даних, де вони є основою Industry 4.0.

Наприклад, конвеєри повинні працювати бездоганно, щоб відповідати вимогам виробничої лінії, синхронізувати з роботизованими системами у складі ГВС і системами управління рухом для максимального збільшення продуктивності. В таких умовах жодна конвеєрна компанія не може задовольнити ці потреби за допомогою застарілих механічних і пневматичних конвеєрів.

Сьогоднішні провідні конвеєрні компанії, що використовують інтелектуальне виробництво, перейшли на повністю електричні приводи, які більш надійні, енергоефективні та безшумні. Ці конвеєрні системи розроблені для легкої інтеграції з крупносерійним автоматизованим виробництвом.

Зараз потрібні конвеєри, на яких відбувається збір даних для виробництва і відстеження – все це можливо завдяки тому, що конвеєри сортують, об'єднують і накопичують продукти (деталі, товари, елементи виробництва, тощо), щоб він потрапляв на автоматизовані пристрої в потрібний час і правильним способом. Вони також відіграють важливу роль у зборі інформації в рамках відстеження загальної ефективності обладнання (ЗЕО) і ключових показників ефективності (КПЕ).

Розумний конвеєр – це кібер-фізична система (КФС) [8], що складається з конвеєра будь-якого типу, який виконує двосторонню передачу даних з використанням протоколу зв'язку. В рамках цього обміну дані передаються в сумісній формі і в залежності від роботи інтелектуального конвеєра. Потім ці дані обробляються і оцінюються в системі з використанням хмарних технологій. Цей процес спрямований на отримання широкого спектру різних показників, що дають інформацію про процес роботи конвеєра. На основі отриманих і оцінених даних можна згенерувати керуючі послідовності за

допомогою експертної системи управління. Потім вони відправляються назад через відповідний інтерфейс зв'язку в блок управління конвеєром (рис. 1.2).

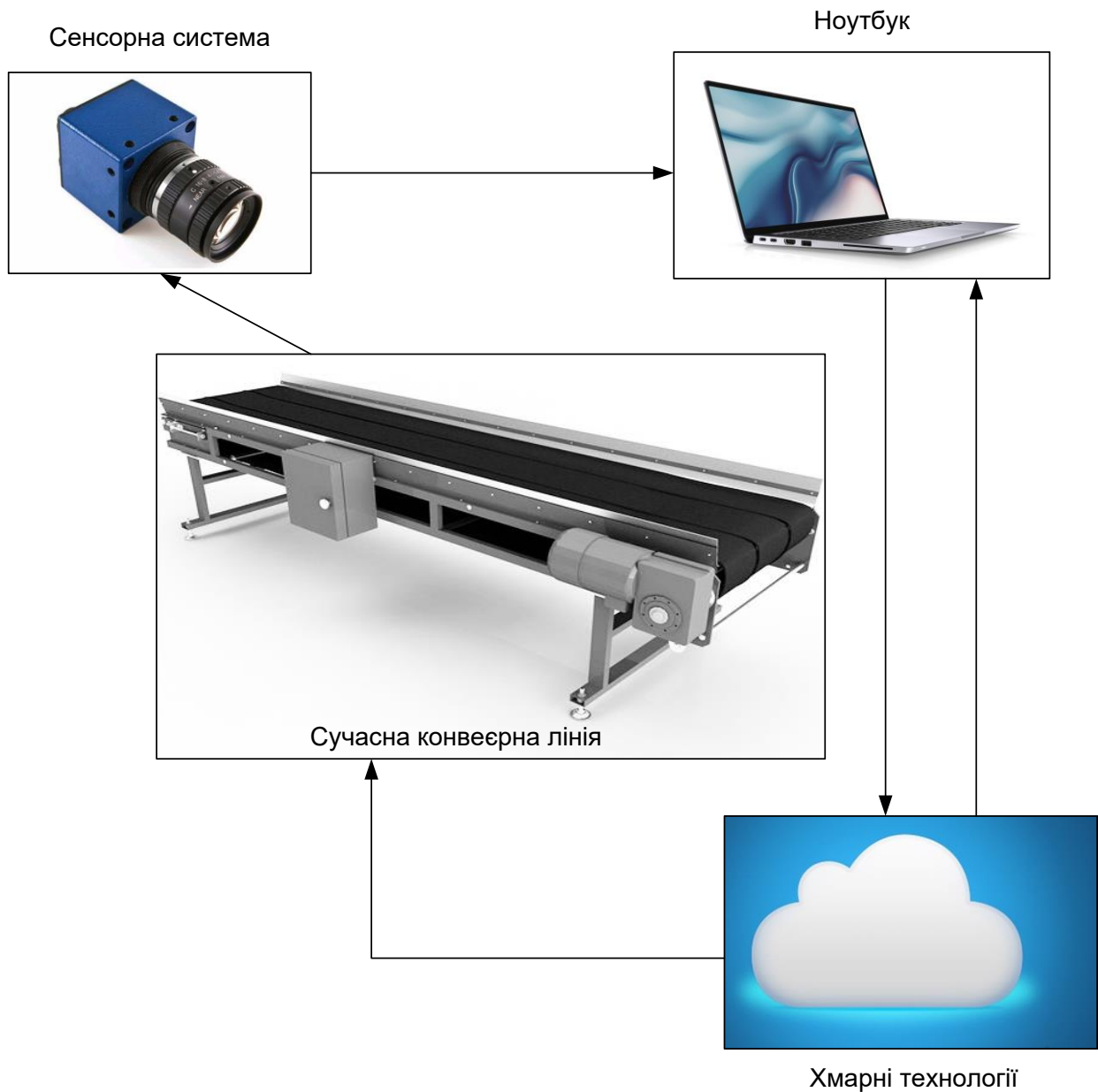


Рисунок 1.2 – Сучасна конвеєрна лінія в Industry 4.0

Розумний конвеєр є багатоагентною системою, оскільки вона складається з окремих структурних елементів, які оснащені сенсорними і керуючими елементами. Такий промисловий агент здатний планувати, координувати або навіть активно виконувати дії в рамках інтелектуального конвеєра.

### 1.3 Процес сортування на конвеєрному виробництві

Сортування є процесом ідентифікації предметів на конвеєрній системі і їх перенаправлення в певне місце призначення в рамках виконання операції, наприклад, на різні типи станцій упаковки або відбракування [9].

У той час як ручне сортування покладається на людей, які ідентифікують і відхиляють елементи в міру необхідності, автоматичне сортування використовує різні технології, такі як сканери штрих-коду та інші датчики, щоб виконати це без автоматичного втручання людини.

Сучасні автоматизовані сортувальні системи істотно різняться залежно від базової технології, на якій вони побудовані. Датчики, засоби транспортування і метод відводу можуть бути адаптовані до конкретних операцій з обробки продукту і потреб в продуктивності.

Оптимізувавши конвеєрні лінії за допомогою операції сортування можна отримати такі переваги:

- зменшення подвійної обробки замовлень;
- збільшення кількості продукції, що випускається;
- зниження витрат на робочу силу;
- комплектації готової продукції за менший час;
- пакувальні операції продукції за менший час;
- підвищена точність замовлення;
- більш швидка їх обробка;
- менше аварійних ситуацій;
- менше зламаних або пошкоджених продуктів виробництва;
- більш ефективне використання площі.

Коли справа доходить до процесу сортування та зборки, то це все привертає до себе всю увагу, і не без підстав – автоматичне сортування може мати велике значення для оптимізації системи на виконання замовлень з урахуванням сучасних технологій Industry 4.0 [10-11].

## **1.4 Використання технологій комп'ютерного та машинного зору для сортування деталей**

У системах машинного зору використовуються існуючі технології для механічного «перегляду» етапів виробничої лінії. Це допомагає виробникам виявляти недоліки в своїх продуктах до того, як вони будуть упаковані, або, наприклад, компаніям з розповсюдження харчових продуктів забезпечити правильну маркування своїх продуктів.

З появою комп'ютерного зору погляд на машинний зір змінився. Система машинного зору використовує камеру для перегляду зображення, потім алгоритми комп'ютерного зору обробляють і інтерпретують зображення, перш ніж дати іншим компонентам системи команду діяти на основі цих даних [12].

Комп'ютерний зір можна використовувати окремо, без необхідності бути частиною більш великої машинної системи. Але система машинного зору не працює без комп'ютера і спеціального програмного забезпечення. Це виходить далеко за рамки обробки зображень. З точки зору комп'ютерного зору (КЗ) зображення навіть не обов'язково має бути фотографією або відео; це може бути так зване зображення від теплового або інфрачервоного датчика, детекторів руху або інших джерел.

Методи комп'ютерного зору все частіше можуть обробляти тривимірні і рухомі зображення, в тому числі непередбачувані спостереження, з якими не могли впоратися більш ранні версії такої технології. Складні операції виявляють всілякі функції в зображенні, аналізують їх і надають велику інформацію про ці зображення.

У процесі розвитку комп'ютерного зору потенційні додатки машинного зору збільшуються в геометричній прогресії. Те, що колись було прерогативою важкої промисловості для визначення простих бінарних дій, тепер з'являється в гальмівних системах автономних транспортних засобів,

порівнює наші особи з нашими фотографіями в паспорті біля воріт безпеки аеропорту і допомагає роботам виконувати операції.

Стандартними компонентами системи машинного зору є:

- камери або фотоапарати;
- освітлення для чіткого зображення;
- лінза;
- фрейм-граббер;
- комп'ютер і програмне забезпечення для аналізу і обробки зображень.

Зростаюче впровадження рішень для комп'ютерного зору на базі штучного інтелекту, споживчих дронів; і зростаюче впровадження Industry 4.0 призведе до феноменальних змін. Ось основні тенденції комп'ютерного зору, які стоять за зростанням комп'ютерного зору для сучасних підприємств [13].

Можна використовувати зіставлення зі зразком і інші алгоритми в залежності від характеру аналізованих зображень.

Компоненти виведення: може бути екран для представлення даних плюс механічні компоненти, такі як роботизована рука, наприклад.

Процес сортування на основі камер і машинного зору зустрічаються в основному в сортувальниках різних об'єктів.

На основі цього були розроблені сортувальні машини, що розділяють різні предмети на конвеєрних лініях.

При сортуванні сміття з'являються такі методи машинного зору, як глибоке навчання. За допомогою таких методів можна класифікувати і виявляти різні об'єкти, такі як скляні пляшки, паперові коробки, паперові стаканчики, керамічні тарілки і т. д.

Більшість методів машинного зору при сортуванні застосовуються у виробництві елементів радіоелектронної апаратури та інших технічних об'єктів.

Вивчаючи сортувальні машини, було виявлено, що існують різні маніпулятори для фізичного поділу частинок. Це різні пневматичні, електричні і гідравлічні маніпулятори для сортування більших об'єктів. Також

широко використовується викид стисненим повітрям і пневматичними соплами для більш дрібних об'єктів. Також існують просунуті алгоритми для відстеження та моделювання виділення одиночних частинок. Ці алгоритми можуть відстежувати об'єкти на конвеєрній стрічці і відокремлювати їх від інших об'єктів з великою точністю.

Сортувальна машина проводить оптичний контроль елементів виробництва з допомогою камери, яка працює у видимому діапазоні світла. Частини зображень потрібно досліджувати, тому використовується елементи машинного зору.

### **1.5 Нейронні мережі та глибоке навчання**

Застосування технологій нейронних мереж (НМ) та Deep Learning (Глибокого навчання, ГН) не можуть бути запрограмовані за допомогою точно визначених алгоритмів для вирішення певних проблем – вони потребують навчання на реальній великій кількості даних, щоб потім була можливість використання у різних ситуаціях для вирішення поставлених перед ними завдань [14].

ГН частіш за все використовується для таких завдань як:

- аналіз текстової інформації;
- аналіз вимовленого слова;
- розпізнавання зображень;
- розумне моделювання поведінки.

З вищенаведеного списку очевидно, що застосування систем ГН в основному використовується для вирішення вузьких цілеспрямованих проблем.

Основним завданням, яке повинно вирішуватись у сучасному світі є підвищення ступеню автоматизації – виробниче підприємство повинні розуміти як складне, згідно поняттю Industry 4.0, а тому успішне

впровадження систем глибокого навчання, в тому числі нейронних мереж, повинно складатися з інтерпретації даних у складному вигляді [15].

Широкий спектр успішних застосувань систем глибокого навчання – це їх застосування у фінансовому секторі, а останнім часом і в орієнтованих на людини системах (поведінковий аналіз, маркетинговий аналіз, прогнози). Їх застосування в технічному секторі або в машинобудуванні ще не широко поширене, тому це є актуальним завданням.

Багато розроблених методологій та інструментів для систем глибокого навчання потребують адаптації для успішного використання їх у технічному секторі. Внутрішні принципи роботи систем залишатимуться подібними до існуючих додатків.

Також при використанні технологій НМ та ГН особливу увагу заслуговує підготовка даних на основі вхідних з подальшою інтерпретацією на вихідній стороні систем глибокого навчання.

Щоб провести модернізацію програм систем ГН в галузі машинобудування, зокрема виробничих системах, необхідно знайти аналогії між даними, з яких і в які дані є можливість перетворити за допомогою систем глибокого навчання (рис. 1.3):

- Розпізнавання мовлення для керування виробництвом – подібна система з використанням глибокого навчання може бути розгорнута в системі моніторингу стану та діагностики виробничих ліній. Після отримання достатньої кількості вихідних зразків (база даних слів), ми можемо створити досить повну базу даних для створення системи, яка зможе оцінити поточний стан виробничої лінії і, можливо, передбачити потенційну несправність. Потім цю базу даних можна використовувати в сценаріях контрольованого навчання, після чого слід подальша реалізація навчання з підкріпленням і змагальних алгоритмів, є можливість довести систему до процесу самонавчання.

- Розпізнавання зображень – система глибокого навчання буде використовуватись для оцінки зображень, отриманих за допомогою машинного/комп'ютерного зору (камери). Початковий набір даних буде

підданий аналізу експертів в цій галузі, щоб ми могли використовувати контрольоване навчання для автономного розпізнавання відомих аномалій на основі бази даних оцінених зразків. Цю систему можливо застосувати в процесі самонастроювання роботизованого пристрою при роботі з продуктами. На основі порівняння інформації про зображення в базі знань система зможе розпізнати поточний стан і внести виправлення в реальному часі. У порівнянні з іншими традиційними підходами до обробки зображень, система ГН має значну перевагу в своїй гнучкості і точності в процесі оцінки і значно підвищує гнучкість процесів виробництва і автоматизації.



Рисунок 1.3 – Використання методів ГН на виробництві

- Аналіз текстової інформації – в цьому випадку необхідно тільки змінити метрику і кодування даних відомих алгоритмів машинного навчання. Застосування систем ГН дозволяє підвищити адаптивність системи

управління до використання ітераційних змін параметрів при підвищенні стійкості системи управління до виникнення фатальних відмов.

- Розумне моделювання поведінки є найскладнішим з існуючих методів. Система управління з використанням ГН отримає нові можливості адаптації для взаємодії з навколишнім середовищем за допомогою підсистеми, здатної до самонавчання і зможе автономно реагувати на нові вимоги. Для того. Щоб ця система працювала, необхідна дуже велика база знань, а також дуже точний і надійний аналіз даних, опис ітерацій між усіма можливими потенційними учасниками, причинний аналіз і, звичайно ж, дуже докладна і точна інтерпретація всіх ітераційних елементів і потрібна алгоритмізація всіх процесів і явищ. Отримані вхідні дані дозволять отримати точну модель всього процесу управління, яка сформує базовий об'єкт для моделювання інтелектуальної поведінки.

## **1.6 Інтелектуальне виробництво та глибоке навчання у концепції Industry4.0**

Під інтелектуальним виробництвом розуміють системи, які надають нам повну інформацію про кожен з етапів життєвого циклу продукту. Глибоке навчання використовується для аналітики в таких напрямках, як перевірка якості продукції, діагностика та оцінка несправностей і прогнозування дефектів [15].

Першим аспектом впровадження глибокого навчання в виробництво Industry 4.0 є контроль якості продукції. Було показано, що машинний зір на основі глибокого навчання є ефективним для контролю якості поверхні. Він використовується для виявлення відхилень – подряпин, потертостей, забруднень, різних змін кольору та глибоких нерівностей на текстурованих або нетекстурованих поверхнях. Більш того, глибокі згорткові мережі (ГЗМ) дозволяють нам виконувати виявлення об'єктів, яке може бути застосоване як

додаткова система контролю роботів (наприклад, рука робота, що встановлює деталь на верстаті з ЧПУ, маніпулювання об'єктами робота або сортування).

Завдання діагностики і оцінки несправностей представляють собою другу категорію, в якій методи глибокого навчання виявляються необхідними. У машинному обладнанні ГЗМ використовувалися для оцінки дефектів підшипників, редукторів, роторів і вітряних генераторів. Діагностика планетарного редуктора проводилася з використанням штучного інтелекту (ШІ) і глибокої нейронної мережі (ГНМ) в різних умовах експлуатації.

Остання область використання глибокого навчання – прогнозування дефектів, тобто прогнозування технічного обслуговування. Наприклад, за допомогою технологій глибокого навчання було проведено довгостроковий прогноз стану підшипників кочення, зроблено прогноз щодо зносу обробного інструменту, забезпечено високу точність прогнозу строку служби двигунів.

Технології глибинного навчання, як і нейронних мереж, мають вирішальне значення для зростання продуктивності виробництва в технології Industry 4.0 і зниження витрат на обслуговування.

## **1.7 Постановка задач дослідження**

Основним завданням дослідження є:

- аналіз існуючих методів автоматизації;
- створення моделі ідентифікації;
- створення методу ідентифікації на основі моделі.

Для підтвердження роботи в цілому буде розроблено програмне забезпечення та проведено експерименти.

## **1.8 Висновки до першого розділу**

В результаті аналізу матеріалу для виконання атестаційної роботи магістра були розглянуті питання:

- аналіз технологій сортування;
- огляд сучасних систем прилодобудування, які використовуються;
- огляд актуальності даної теми у концепції Industry 4.0.

## 2 РОЗРОБКА МЕТОДУ ІДЕНТИФІКАЦІЇ ДЕТАЛЕЙ ДЛЯ ПРОЦЕСУ СОРТУВАННЯ НА БАЗІ КОМП'ЮТЕРНОГО ЗОРУ

### 2.1 Основні технології ідентифікації

#### 2.1.1 Бібліотека OpenCV

Сучасне програмне забезпечення для комп'ютерного зору включає бібліотеки для програмування, інструменти для роботи з нейронними мережами та хмарні рішення для виконання завдань із зображеннями.

Розглянемо найпопулярніші інструменти, які зараз широко використовуються на практиці.

Бібліотека OpenCV є безкоштовною бібліотекою, яка служить для обробки зображень у реальному часі та є стандартною для всіх програм, що пов'язані з комп'ютерним зором.

OpenCV реалізована для більшості популярних операційних систем (Linux, OS X, Windows, Android, iOS та інші). Перша її реалізація була мовою C, але популярність бібліотек значимо повернулась до виходу версії 2.0, переписаної на C++. Нові функції пишуться лише на C++. Завдяки цьому, в даний час з'являються повнофункціональні інтерфейси між OpenCV та іншими мовами програмування, такими як Java, Python та MATLAB/Octave, а також для таких мов, як C#, Ruby та Perl, щоб популяризувати бібліотеку серед програмістів [16].

Для отримання максимальної продуктивності при вирішенні великих завдань, які потребують багато процесорного часу, тобто машинного зору, OpenCV включає підтримку наступних можливостей:

- багатопоточність – при роботі на багатоядерних комп'ютерах за рахунок використання шаблонної бібліотеки Threading Building Blocks (ТВВ), розробленої Intel;

- підмножини бібліотеки Integrated Performance Primitives (IPP) на процесорах Intel для різкого підвищення продуктивності., які включені в бета-версію 3.0;

- інтерфейси для роботи з графічними процесорами (GPU) з прикладом архітектури Compute Unified Device Architecture (CUDA) і Open Computing Language (OpenCL).

OpenCV застосовується в таких областях, як сегментація, виділення примітивів у двомірних та тримірних зображеннях, ідентифікація об'єктів, розпізнавання осіб, трасування руху, розпізнавання жестів, зшивка зображень, обробка зображень з широким динамічним діапазоном (високий динамічний діапазон, HDR), доповнена реальність і т . д. Крім того, для підтримки деяких з завдань включений модуль, що містить функції статистичного машинного навчання.

Основними модулями є:

- `sxcore` – ядро, яке містить основні структури даних і алгоритми (операції з масивами, матричну алгебру, математичні функції, генератори випадкових чисел, запис та відновлення структур даних XML, базові функції 2D графіки)

- `CV` – модуль, який служить для обробки зображень і комп'ютерного зору (фільтрація, геометричні перетворення, перетворення колірного простору, виділення характерних ознак, морфології, пошук контурів, гістограм, аналіз руху, відстеження об'єктів, розпізнавання об'єктів, зокрема осіб, зміна параметрів калібрування камери, відновлення елементів просторової структури)

- `Highgui` – модуль для введення-виведення зображень і відео, створення призначеного для користувача інтерфейсу (робота з відеоданими, елементами інтерфейсу користувача, робота з стереозображеннями) [17].

## 2.1.2 Бібліотека Tensorflow

На сьогоднішній день це найбільш популярна бібліотека машинного навчання і глибокого навчання. Популярність Tensorflow швидко росла і перевершувала існуючі бібліотеки завдяки простоті API. Google випустив його в листопаді 2015 року.

TensorFlow є безкоштовна бібліотека з відкритим вихідним кодом для потоків даних і диференціального програмування. Це символічна математична бібліотека, яка також використовується для додатків машинного навчання, таких як нейронні мережі [18].

У TensorFlow 2.0 обчислення конкретних значень відбувається по шляху, щоб побудувати повний обчислювальний граф. Це спрощує налагодження моделі і усуває необхідність в стандартному коді.

Також є можливість використовувати стандартні структури Python в якості структур даних, швидко перевіряти гіпотези і легко налагоджувати код на невеликих моделях і невеликих обсягах даних. Активне виконання також підтримує прискорення графічного процесора і розподілені обчислення на багатьох машинах.

TensorFlow 2.0 полегшує реалізацію попередньо навчених моделей, налаштованих для розпізнавання зображень і мови, виявлення об'єктів, рекомендацій, посиленого навчання і т. д. Такі еталонні моделі служать відправною точкою для розробки нових додатків з навчанням.

Що найбільш важливо, можна розбити графік на кілька частин і запустити їх паралельно на декількох процесорах або графічних процесорах. TensorFlow також підтримує розподілені обчислення, тому ви можете навчати колосальні нейронні мережі на величезних навчальних наборах за розумний час, розділяючи обчислення між сотнями серверів. TensorFlow може навчати мережу з мільйонами параметрів на навчальному наборі, що складається з мільярдів примірників з мільйонами функцій кожен.

### 2.1.3 Мова програмування CUDA

Блоки графічних процесорів (БГП) еволюціонували для обробки робочих навантажень, орієнтованих на пропускну здатність, в яких повинна виконуватися велика кількість паралельних потоків. Такі потоки організовані навколо спільної пам'яті, що дозволяє синхронізувати загальні дані і взаємодіяти з ними. Сучасні графічні процесори можуть виконувати десятки тисяч апаратних потоків і оптимізовані для графічних робочих навантажень. Було розроблено кілька мов високого рівня, що дозволяють легко програмувати графічні процесори для вирішення спільних обчислювальних задач. Використання мов високого рівня вимагає високо оптимізуються компіляторів, націлених на пристрій з паралельним графічним процесором.

CUDA (Compute Unified Device Architecture) є апаратною і програмною архітектурою для паралельних обчислень, яка служить для підвищення продуктивності обчислень за рахунок використання графічних процесорів Nvidia [19].

CUDA SDK надає можливість включати процедури виклику мовою C, що працюють на графічних процесорах Nvidia. Це робиться за допомогою команд, написаних на мові C. Архітектура CUDA дає розробнику можливість на свій розсуд отримувати доступ до набору інструкцій графічного прискорювача і керувати ним.

З використанням CUDA, додатки можуть досягати більшої продуктивності при роботі з потоковим відео з великою частотою кадрів [20].

За допомогою CUDA можна робити наступні функції:

- сегментація налаштування рівня освітлення;
- сегментація відео;
- реалізація мультикласової SVM в CUDA;
- виявлення пішоходів;
- виявлення щільного оптичного потоку;
- виявлення байєсівського оптичного потоку;
- машинне навчання та обробка даних;

- апаратне прискорення команд;
- швидкий пошук найближчого сусіда з використанням графічного процесора.

За допомогою CUDA завдання комп'ютерного зору можуть вирішуватися швидше, а технології виходять на новий рівень; наприклад, можливість автоматичного керування транспортними засобами, обробки великих обсягів фото- і відеоінформації, обробки медичних відеоданих і т. д.

#### **2.1.4 Бібліотека Theano**

Theano – це бібліотека Python, яка дозволяє вам визначати, оптимізувати і оцінювати математичні вирази, особливо з багатовимірними масивами (numpy.ndarray). Використовуючи Theano, можна досягти швидкості, порівнянної з створеними вручну реалізаціями мовою C для задач, пов'язаних з великими обсягами даних. Ця бібліотека також може на багато порядків перевершити C на процесорі, використовуючи переваги останніх графічних процесорів [21-23].

Theano поєднує аспекти системи комп'ютерної алгебри (СКА) з аспектами оптимізуючого компілятора. Також є можливість генерувати індивідуальний код мови C для багатьох математичних операцій. Ця комбінація СКА з оптимізуючою компіляцією особливо корисна для завдань, в яких складні математичні вирази обчислюються багаторазово, а швидкість оцінки має вирішальне значення. Для ситуацій, коли кожне з безлічі різних виразів обчислюється один раз, Theano може мінімізувати обсяг витрат часу на компіляцію та аналіз, але все ж надати символічні функції, такі як автоматичне розрізнення.

Компілятор Theano застосовує до цих символічних виразів безліч оптимізацій різної складності. Ці оптимізації включають, але не обмежуються:

- використання GPU для обчислень;
- постійне складання;
- об'єднання схожих підграфів, щоб уникнути зайвих обчислень;

- арифметичне спрощення;
- вставка ефективних операцій BLAS (наприклад, GEMM) в різні контексти;
- використання псевдонімів пам'яті, щоб уникнути обчислень;
- використання внутрішніх операцій всюди, де це не заважає згладжування;
- злиття циклів для поелементних підвиразів;
- поліпшення чисельної стабільності.

### **2.1.5 Бібліотека Keras**

Keras є бібліотекою для мови Python, яка використовується для глибокого навчання і поєднує в собі функції Tensorflow, Theano і CNTK. У Keras є перевага перед конкурентами, такими як Scikit-learn і PyTorch, оскільки вона працює поверх Tensorflow [24].

Keras може працювати поверх TensorFlow, Microsoft Cognitive Toolkit, Theano або PlaidML. Вона розроблена для швидкого проведення експериментів над глибокими нейронними мережами (ГНМ) та орієнтована на зручність використання, модульність і розширюваність.

Keras є API, яким дуже легко користуватись – вона пропонує послідовні і прості API-інтерфейси і зводить до мінімуму кількість дій користувача, необхідних для типових випадків використання.

Зараз Keras є частиною TensorFlow для кращої інтеграції з іншими API TensorFlow, такими як нетерпляче виконання, tf.data та багато інших переваг.

### **2.1.6 Бібліотека Scikit-learn**

Ця популярна бібліотека машинного навчання побудована на NumPy, SciPy та matplotlib. Та основу увагу приділяє алгоритмам машинного навчання [25]:

- навчання під контролем;
- навчання без нагляду;

- лінійна регресія;
- логістична регресія;
- підтримує векторний апарат (SVM);
- наївний класифікатор Байєса;
- посилення градієнта;
- скупчення;
- метод К-засобів.

Ця бібліотека менш розвинена, ніж Tensorflow, однак вона пропонує прості та ефективні інструменти для виявлення та аналізу даних

### **2.1.7 Алгоритми YOLO**

YOLO є система виявлення об'єктів, розроблена спеціально для обробки в реальному часі. Реалізаціями цієї системи є: SSD300, SSD500, YOLOv2 і Tiny YOLO. Їх алгоритм застосовує нейронну мережу до всього зображення, а нейронна мережа ділить зображення на сітку і зазначає області з виявленими об'єктами. Перевагами цієї системи є те, що вона дуже швидка і має гарну продуктивність для обробки в реальному часі, а також прогнози робляться з однією єдиною мережі [26].

YOLO є узагальненим. Він перевершує інші методи при узагальненні на основі природних зображень.

Методи виявлення частини (регіону) на зображенні обмежують класифікатор конкретним регіоном. YOLO звертається до всього зображення при прогнозуванні кордонів, а також виявляє один об'єкт на кожну клітинку сітки.

### **2.1.8 Хмарні інструменти комп'ютерного зору API Google Cloud і Mobile Vision**

Google надає AutoML Vision у вигляді хмарної служби. Це дозволяє автоматизувати навчання моделей машинного навчання. Необхідно завантажувати зображення і навчати призначені для користувача моделі

зображень за допомогою простого у використанні графічного інтерфейсу AutoML, що дозволяє оптимізувати моделі по точності, затримки і розміром; і експортувати їх в ваш додаток в хмарі або на безліч пристроїв.

Google Cloud Vision API пропонує потужні попередньо навчені моделі машинного навчання за допомогою REST і RPC API. Він привласнює зображень ярлики і швидко класифікує їх за мільйонами зумовлених категорій. Google Vision API дозволяє нам виявляти об'єкти і особи, читати друкований і рукописний текст і створювати цінні метадані в каталозі зображень [25].

API Vision Google Cloud дозволяє розробникам аналізувати зображення і контекстні дані, використовуючи принципи самонавчання,

Ще однією з таких систем є Amazon Rekognition, яка дозволяє легко вбудовувати аналітику зображень і відео в програму за допомогою глибокого навчання. Служба може розпізнавати об'єкти, людей, текст, сцени і дії, а також виявляти неприйнятний контент. Крім того, Amazon Rekognition точно аналізує і розпізнає обличчя на зображеннях і відео клієнтів. Amazon Rekognition є простим API, який швидко аналізує будь-які зображення і відео, що зберігаються в Amazon S3.

Mobile Vision – це API, який допомагає нам знаходити об'єкти на фотографіях і відео за допомогою технології Google Vision на пристрої в реальному часі. Ця структура може виявляти об'єкти на фотографіях і відео.

В даний час Android API може виявляти<sup>^</sup>

- обличчя;
- штрих код;
- текст.

Доступні три типи детекторів

- детектор осіб;
- детектор штрих-коду і QR-коду;
- детектор тексту.

## 2.2 Розробка моделі ідентифікації деталей для процесу сортування на базі комп'ютерного зору

Модель може допомогти пояснити систему і вивчити ефекти різних компонентів системи, щоб передбачити її поведінку.

Математичне моделювання системи (параметрична модель) розпізнавання виглядає наступним чином.

Нехай нам потрібно провести розпізнавання та ідентифікацію об'єктів  $Rcg_{Obj}$ . За своєю суттю вона являє собою наступний процес:

$$Rcg_{Obj} = \{Set_{in}, Set_{out}, \delta, Param\}$$

Цей процес вимагає в своєму складі багато різних параметрів.

Так як використано нейронні мережі, то будуть присутні як вхідні, так і вихідні дані.

Першим з них є  $Set_{in}$  – набір вхідних даних для розпізнавання, який складається з таких параметрів як:

- інформація про відеофайл, з якого будуть братися об'єкти для розпізнавання та ідентифікації;
- інформація про зображення з яких будуть братися об'єкти для розпізнавання та ідентифікації;
- інформація про сам об'єкт і помилки розпізнавання та ідентифікації.

Математичним способом описання це буде виглядати так:

$$Set_{in} = \{vid_f, img_f, ch_f\}$$

де  $vid_f$  – інформація про відеофайл,

$img_f$  – інформація про зображення,

$ch_f$  – інформація про об'єкт розпізнавання.

Другим параметром, що потрібен для цього процесу є набір вихідних даних розпізнавання  $Set_{out}$ . Для цього набору також є математичний опис, який буде складатися з параметрів:

- інформація, про те, що знайшлися співпадиння;
- сповіщення про співпадиння;
- інформація про те, що співпадиння не знайшлися.

Параметрично це можна записати в наступному вигляді:

$$Set_{out} = \{ pos_{inf}, ntf, neg_{inf} \}$$

де  $pos_{inf}$  – інформація, що знайшлися співпадиння,

$ntf$  – сповіщення про співпадиння;

$neg_{inf}$  – інформація, що співпадиння не знайшлися.

Наступним параметром є  $\delta$ . Цей параметр відповідає за розпізнавання та ідентифікацію об'єкта, а саме виділення його на зображенні та мінімальної Евклідової відстані. Цей параметр у своєму складі буде мати декілька шагів.

Перший шаг є початковим. Тут відбувається ініціалізація процесу розпізнавання та ідентифікації.

Другим шагом є читання набору для тренування  $NN_{set}$ , що буде складатися з  $N \times N$  зображень.

$$NN_{set} = N \times N$$

Наступним шагом є приведення цього набору до набору потрібного розміру  $NN_{norm}$ . Математично це можна записати у вигляді:

$$NN_{norm} = NN_{set} \times Mat_{norm}$$

де  $Mat_{norm}$  – матриця нормалізації, яка автоматично обраховується при отриманні набору зображень.

Обирається набір даних для тренування  $Tr_{data}$ , що матиме певний розмір  $i$  і позначає кількість вимірювань, матиме наступний математичний вираз:

$$Tr_{data} = NN_{norm} \times M$$

де  $M$  – кількість зразків зображень.

Знаходимо потрібний об'єкт на зображенні з усіх об'єктів у наборі даних, формуємо матрицю  $Mat_{obj}$  з основними параметрами об'єкту.

Для цього спочатку знаходимо усереднене зображення  $Av_{img}$  – усереднений набір параметрів для розпізнавання і ідентифікації, що буде представлений у вигляді матриці або вектору.

Математично цей вираз матиме наступний вид:

$$Av_{img} = \frac{1}{M} \sum_{i=1}^M Vec_i$$

Де  $M$  – кількість зразків зображень,

$Vec_i$  – вектор зображення.

Виділяємо набір параметрів  $Dif_i$ , що відрізняє цей об'єкт розпізнавання.

Математично цей вираз можна записати у вигляді:

$$Dif_i = Vec_i - Av_{img}$$

де  $i = 1, 2, 3, \dots, M$

Тоді матриця  $Mat_{obj}$  з основними параметрами об'єкту матиме наступний вигляд:

$$Mat_{obj} = [Dif_1, Dif_2, \dots, Dif_M]$$

Обчислюється коваріаційна матриця  $Cov$ , що математично може бути записана як:

$$Cov = Mat_{obj}^T \times Mat_{obj}$$

Обчислюємо власні вектори  $Vec_{Cov}$  коваріаційної матриці  $Cov$ .

$$Cov \rightarrow Vec_{Cov}$$

Обчислити власні об'єкти Eigenfaces. Ця технологія використовує двовимірні зображення в градаціях сірого, які представляють відмінні характеристики зображення обличчя (в нашому випадку – об'єкту розпізнавання та ідентифікації).

$$Eigenfaces = Tr_{data} - Vec_{Cov}$$

Створюємо зменшений простір  $Eigenfaces_s$ , який виділяє тільки основні риси об'єкту розпізнавання та ідентифікації, що математично має наступний вид:

$$Eigenfaces_s = Eigenfaces \times Mat_{obj}$$

Далі обчислюється  $Eigenfaces_{obj}$  – об'єкту, який повинен бути розпізнаний та проідентифікований.

$$Eigenfaces_{obj} = Eigenfaces \times Mat_{obj1}$$

Обчислюється евклідова відстань між  $Eigenfaces_{obj}$  та  $Eigenfaces_s$ :

$$Ecd_{Dist} = Eigenfaces_{obj} - Eigenfaces_s$$

Якщо ця відстань мінімальна, то об'єкт розпізнано та проідентифіковано:

$$\min |Ecd_{Dist}| \rightarrow 0$$

І останнім параметром  $Param$  є параметр, що зберігає наступну інформацію:

- система не буде обробляти дані;
- $Eigenfaces$  буде генерувати зображення в градаціях сірого;
- алгоритм буде працювати тільки з ключовими кадрами.

Математично це може бути записано:

$$Param = \{ Data_{mis}, Eigenfaces_{gray}, frame_{key} \}$$

де  $Data_{mis}$  – необроблені дані;

$Eigenfaces_{gray}$  – згенеровані сірі зображення ;

$frame_{key}$  – лише ключові кадри.

### **2.3 Розробка методу ідентифікації деталей для процесу сортування на базі комп'ютерного зору**

Розробимо метод ідентифікації деталей  $Rcg_{Obj}$  для процесу сортування.

В загальному вигляді основними кроками цього методу будуть наступні.

Етап 1. Збір і підготовка даних. На цьому етапі потрібно отримати зображення  $vid_f$ , відеодані  $vid_f$  та інформацію про об'єкт розпізнавання  $ch_f$

які будуть використовуватись для навчання нейронної мережі –  $Set_{in}$ . Також слід брати до уваги те, що зображення можуть мати погану якість, тому, можливо прийдеться проводити фільтрацію цих зображень. Також важливо, щоб розміри цих зображень були однакові. Після того, як зображення будуть приведені до нормального виду, потрібно буде створити файл з назвами класів, до яких будуть відноситись об'єкти на зображеннях. Далі потрібно створити файл з їх характеристиками – це можуть координати (координати верхнього лівого та нижнього правого кутів рамки, яка буде обводити об'єкт) рамки, що позначає границі об'єкту, який буде ідентифікуватись, також назва або ідентифікатор класу. Всі ці дії потрібно проводити над кожним з зображень. Також всі дані потрібно розбити на дві множини: множину для навчання (тестову) та еталонні, до яких буде йти навчання.

Етап 2. Підготовка вихідних даних. Існує два підходи до навчання – з учителем та без. Навчання з учителем полягає в тому, що для розпізнавання образів з учителем є вибірка з еталонними відповідями на питання (еталонні зображення), що подаються на вхід. На основі цієї інформації обчислюється помилка, що порівнює вихідні значення розпізнаних зображень з еталонними. Потім деякі налаштування змінюються, поки помилка не стане мінімальною.

У випадку навчання без учителя у навчальній вибірці немає еталонних зображень та класів і основним завданням стає самостійне знаходження закономірностей в даних на основі аналізу і отримання основних ознак про об'єкти методом, наприклад, кластеризації – підбір схожих даних та знаходження спільних ознак з подальшим групуванням їх разом. Точність такого алгоритму визначити дуже складно. Так як використовується навчання з учителем, то для цього нам потрібні вихідні дані  $Set_{out}$ . В цих даних поміщено інформацію про те, що знайшлося  $pos_{inf}$  чи не знайшлося співпадіння  $neg_{inf}$  з вхідною базою даних зображень.

Етап 3. Ініціалізація процесу розпізнавання та ідентифікації  $\delta$  об'єкту на зображенні, а саме виділення його на зображенні та обчислення мінімальної Евклідової відстані та у своєму складі буде мати декілька шагів.

Також на цьому етапі проводиться вибір топології нейронної мережі. Існує багато архітектур для створення нейронних мереж та розпізнавання об'єктів: багат шаровий перцептрон, згорткова нейрона мережа, рекурсивна, рекурентна, мережа довгої короткостроковій пам'яті, Sequence-to-sequence модель та неглибока мережа. Для нашого випадку вибір буде між перцептроном та згортковою нейронною мережею (ЗНМ). У порівнянні з перцептроном, згорткова мережа має набагато меншу кількість настроюються ваг. Тут використовується лише одне ядро ваг для всього зображення. В перцептроні для кожного пікселя вхідного зображення задаються персональні вагові коефіцієнти. Завдяки цьому нейронна мережа при навчанні запам'ятовує узагальнену інформацію і їй не потрібно попіксельно запам'ятовувати кожне зображення в міриадах вагових коефіцієнтів, як це робить перцептрон. Також ЗНМ має незначну реакцію на поворот зображення, на відміну від перцептронів

Етап 4. Зчитування набору даних  $NN_{set}$ , який буде складатись з певної кількості зображень, яка залежить від об'єкту, що повинен бути розпізнаний та проідентифікований.

На цьому етапі також повинен проводитись підбір основних характеристик мережі. До таких відносять – число шарів та нейронів, а також не треба забувати й про кількість зв'язків між нейронами. Ці параметри підбираються на основі функцій, які повинні вирішуватись. Частіше за все, спочатку обирають один проміжний шар з числом елементів в ньому, рівним напівсумі числа входів і числа виходів. Також ці параметри вже обираються на етапі навчання нейронної мережі. Всі вхідні і вихідні параметри можуть бути представлені у вигляді чисел з плаваючою точкою в діапазоні  $[0..1]$ , а також це можуть бути числа в довільному діапазоні, дати, символічні рядки. Таким чином дані про проблему можуть бути як кількісними, так і якісними

Етап 5. Приведення цього набору даних  $NN_{set}$  до набору потрібного розміру  $NN_{norm}$ . Якщо зображення з об'єктом дуже великі, то необхідно зробити так звану згортку зображення або матрицю нормалізації  $Mat_{norm}$  – отримання такої матриці, що використовується в обробці зображень як фільтр для розмиття, підвищення різкості, виділення границь тощо. Обробка зображення полягає у обчисленні нового значення обраного пікселя з врахуванням значення оточуючих його пікселів.

Спочатку обирається набір даних для тренування  $Tr_{data}$ , що буде складатися з набору параметрів зображень вже зміненого розміру.

Етап 6. На цьому наборі зображень знаходиться цільовий об'єкт і його основні параметри, які потім усереднюються  $Av_{img}$  для оптимального процесу розпізнавання та ідентифікації.

Найважливішим параметром, який використовується є визначення швидкості навчання мережі. Від цього параметру залежить, наскільки швидко результати навчання нейронної мережі будуть сходитися до еталонних результатів за якими проводиться навчання. Чим нижче швидкість навчання та час сходження, тим краще розпізнавання. При збільшенні швидкості навчання час збіжності може як збільшитись, так і зменшитись, в залежності від форми поверхні помилки

Етап 7. Отримання набору параметрів об'єкту розпізнавання та ідентифікації  $Dif_i$ , що унікально відрізняють цей об'єкт серед інших і формування матриці  $Mat_{obj}$  цих параметрів, а також навчання мережі.

По-перше, воно здійснюється на обраній вибірці зображень. По-друге, потрібно правильно обрати кількість епох – ітерація в процесі навчання, що включає перебір всіх елементів з навчальної множини один раз. А вже далі проводити навчання

Етап 8. Формування коваріаційної матриці  $Cov$ , за допомогою якої розраховуються основні показники випадкового розсіювання оцінок

навколо відповідних істинних значень параметрів, що аналізуються, а також характеристики взаємозв'язків отриманих оцінок.

Етап 9. Обчислення власних векторів  $Vec_{Cov}$ , власних об'єктів Eigenfaces, що представляють собою набір основних рис об'єкту розпізнавання та ідентифікації. Розмір цих векторів і об'єктів досить великий, тому потрібно зробити його зменшену версію.

Етап 10. Створення зменшеного простору набору основних рис об'єктів у просторі  $Eigenfaces_s$ .

Етап 11. Створення зменшеного простору набору основних рис цільового об'єкту  $Eigenfaces_{obj}$ .

Етап 12. Обчислення евклідової відстані між цими наборами рис  $Ecd_{Dist}$  та якщо ця відстань мінімальна  $\min|Ecd_{Dist}| \rightarrow 0$ , то об'єкт розпізнано та проідентифіковано.

Етап 13. Отримання інформації про якість розпізнавання та помилки  $Param$ . Для цього на тестовій множині здійснюється перевірка побудованої моделі. Далі отримується результат:

- істинопозитивний – розроблений класифікатор вирішив, що об'єкт розпізнано і він справді розпізнаний;
- хибнопозитивний – розроблений класифікатор вирішив, що об'єкт розпізнано але він не є розпізнаним;
- псевдонегативний – розроблений класифікатор вирішив, що об'єкт не розпізнано, а він розпізнаний;
- істинонегативний – розроблений класифікатор вирішив, що об'єкт не розпізнано і він справді не розпізнаний.

Крім прямої оцінки достовірності у відсотках існують такі метрики, як точність і повнота, засновані на вищенаведених результатах бінарної класифікації.

На основі цієї інформації можна зробити висновок про те, чи вірно працює нейронна мережа і процес розпізнавання та ідентифікації.

Етап 14. Корекція основних характеристик. В разі, якщо виникають помилки при розпізнаванні, повертаємося до етапу 15. Якщо таких помилок немає – переходимо до етапу 6.

Етап 15. Опис роботи нейронної мережі у вигляді функцій  $Rcg_{Obj} = \{Set_{in}, Set_{out}, \delta, Param\}$ . Цей Етап потрібен для того, щоб мінімізувати такий параметр навчання, як час. Після того, як нейронна мережа навчилася, перший та останні кроки порівнюються, а також беруться до уваги дії, які використовувались для навчання мережі і формується математичний чи вербальний опис алгоритму навчання за допомогою функцій, або словесного опису.

Етап 16. Створення пустого проекту. Для цього треба обрати мову програмування. Нейронні мережі підключаються до будь-якої мови програмування та їх IDE. Кращим та найпростішим варіантом є мова Python. Обґрунтування вибору мови буде проводитись у наступній главі.

Етап 17. Підключення модулів нейронної мережі до програмного забезпечення. На цьому етапі потрібно завантажити та підключити основні бібліотеки, які потрібні для навчання нейронної мережі та створення програмного забезпечення.

Етап 18. Підключення модулів комп'ютерного зору до програмного забезпечення. Для того, щоб програмне забезпечення працювало з потоковим відео для розпізнавання потрібно завантажити і підключити модулі бібліотеки комп'ютерного зору.

Етап 19. Створення функцій та методів для ідентифікації деталей. Для використання нейронної мережі треба розробити методи і функції, за допомогою яких по-перше, відбувається покадрова обробка відео, по-друге, використовується функціонал розробленої мережі.

Етап 20. Запуск на виконання розробленого програмного забезпечення. На цьому етапі проводиться компіляція проекту та виправлення помилок програмного забезпечення.

Етап 21. Отримання ідентифікованих деталей для процесу сортування  $Rcg_{Obj}$ . Результатом виконання є ідентифіковані деталі для процесу сортування за допомогою обмежуючої рамки з написаним над ним класом, до якого ці деталі відносяться.

## **2.4 Висновки до другого розділу**

Під час виконання атестаційної роботи у другому розділі були розглянуті такі питання:

- основні технології ідентифікації деталей на базі комп'ютерного зору;
- розробили модель ідентифікації деталей;
- ґрунтуючись на модель розробили метод ідентифікації

# **3 РОЗРОБКА ПРОГРАМНОГО ЗАСОБУ ДЛЯ ІДЕНТИФІКАЦІЇ ДЕТАЛЕЙ ДЛЯ ПРОЦЕСУ СОРТУВАННЯ НА БАЗІ КОМП'ЮТЕРНОГО ЗОРУ**

## **3.1 Вибір і обґрунтування технологій для створення програмного забезпечення**

На основі розробленого методу та моделі для ідентифікації деталей для процесу сортування на базі комп'ютерного зору було прийнято рішення розробити програмне забезпечення.

В якості мови програмування для даного програмного забезпечення було обрано Python. Основними перевагами цієї мови програмування є:

- величезний вибір бібліотек для будь-яких напрямлень розробки;
- можливість інтеграції через використання мов C, C++ та Java як основ для створення програм, тому що Python є інтерпретатором мови C;
- підвищення продуктивності під час програмування завдяки легкості написання коду;
- збільшення швидкості розробки програмного забезпечення;
- дуже сильна підтримка мови.

Для створення програмного забезпечення потрібно використовувати версію Python 3.6, тому що більш нові версії ще не підтримують використання бібліотек нейронних мереж.

Наступною бібліотекою, яка буде використовуватись є cvlib завдяки своїй простоті, також вона є високорівнева і зручна у використанні з відкритим вихідним кодом, що побудована для мови для Python. Основні параметри і принципи цієї бібліотеки були взяті в значній мірі з бібліотеки Keras, яка використовується для глибокого навчання та входить в пакет Tensorflow. Вона розроблена щоб можна було легко і швидко експериментувати. Можливість перейти від ідеї до прототипу з найменшою затримкою. Основні переваги:

- простота;
- зручність для користувача;
- модульність;
- розширюваність.

Наступна бібліотека, без якої не обійдеться розробка програмного забезпечення є бібліотека OpenCV. Вона використовується для отримання кадрів з відеокамери та виведення цих кадрів у створені вікна.

### 3.2 Алгоритм роботи програми

Алгоритм роботи програми представлено на рисунку 3.1

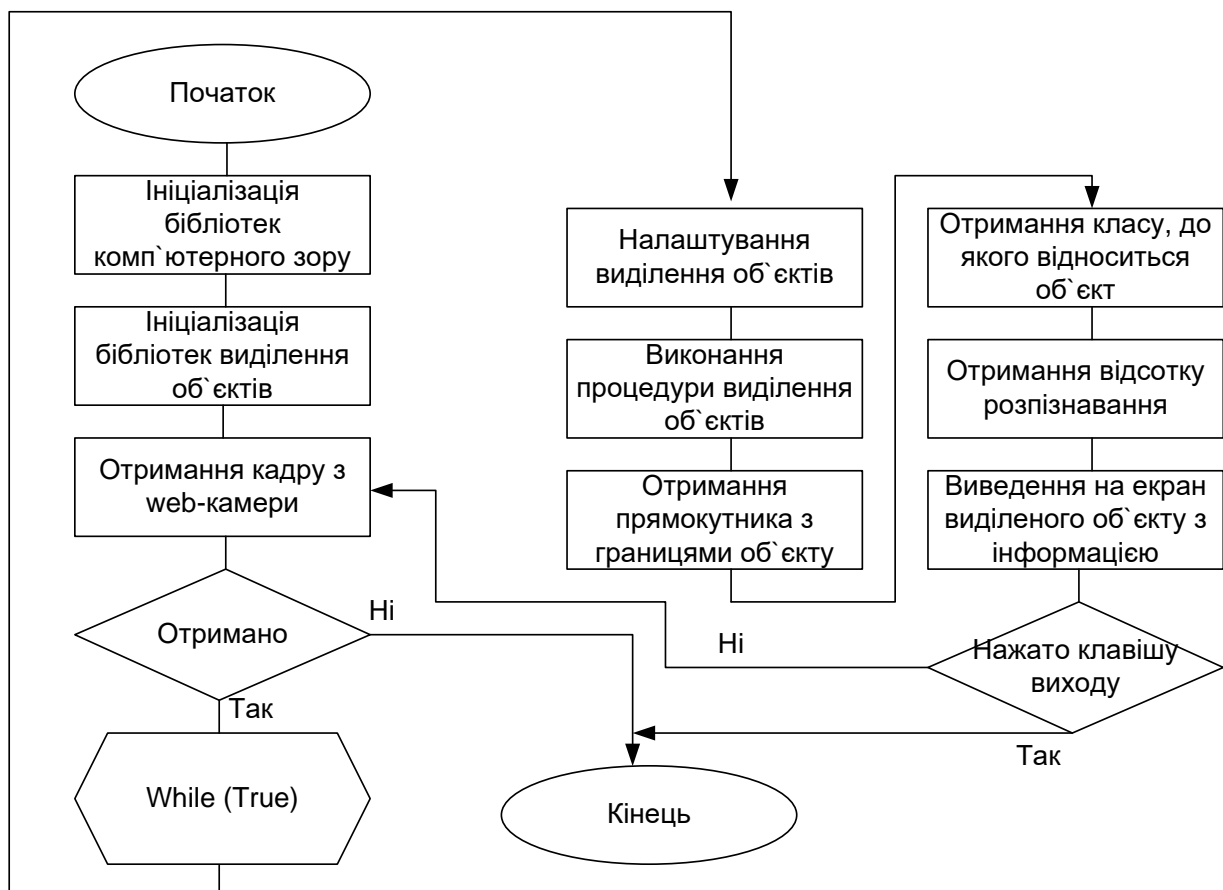


Рисунок 3.1 – Алгоритм роботи програмного забезпечення

### 3.3 Розробка програмного забезпечення

Програмне забезпечення було розроблене у середовищі програмування PyCharm за допомогою Python 3.6.

Для того, щоб створити програмне забезпечення потрібно завантажити та інсталювати декілька модулів (рис. 3.2-3.3).

Package	Version	Latest version
Keras	2.4.3	2.4.3
Keras-Preprocessing	1.1.2	1.1.2
Markdown	3.3.2	▲ 3.3.3
Pillow	8.0.0	▲ 8.0.1
PyYAML	5.3.1	5.3.1
Werkzeug	1.0.1	1.0.1
absl-py	0.10.0	0.10.0
astunparse	1.6.3	1.6.3
cachetools	4.1.1	4.1.1
certifi	2020.6.20	2020.6.20
chardet	3.0.4	3.0.4
cvlib	0.2.5	0.2.5
cycler	0.10.0	0.10.0
flatbuffers	1.12	1.12
gast	0.4.0	0.4.0
google-auth	1.22.1	1.22.1
google-auth-oauthlib	0.4.1	0.4.1
google-pasta	0.2.0	0.2.0
grpcio	1.32.0	▲ 1.33.1
h5py	2.10.0	2.10.0
idna	2.10	2.10
imageio	2.9.0	2.9.0
importlib-metadata	2.0.0	2.0.0
imutils	0.5.3	0.5.3
kiwisolver	1.2.0	1.2.0

Рисунок 3.2 – Перша частина модулів

Package	Version	Latest version
matplotlib	3.3.2	3.3.2
numpy	1.19.2	1.19.2
oauthlib	3.1.0	3.1.0
opencv-python	4.4.0.44	4.4.0.44
opt-einsum	3.3.0	3.3.0
pip	10.0.1	▲ 20.2.4
progressbar	2.5	2.5
protobuf	3.13.0	3.13.0
pyasn1	0.4.8	0.4.8
pyasn1-modules	0.2.8	0.2.8
yparsing	2.4.7	2.4.7
python-dateutil	2.8.1	2.8.1
requests	2.24.0	2.24.0
requests-oauthlib	1.3.0	1.3.0
rsa	4.6	4.6
scipy	1.5.3	1.5.3
setuptools	50.3.2	50.3.2
six	1.15.0	1.15.0
tb-nightly	2.4.0a20201020	
tensorboard	2.3.0	2.3.0
tensorboard-plugin-wit	1.7.0	1.7.0
tensorflow	2.3.1	2.3.1
tensorflow-estimator	2.3.0	2.3.0
termcolor	1.1.0	1.1.0
tf-estimator-nightly	2.4.0.dev2020102001	

Рисунок 3.3 – Друга частина модулів

Всі ці модулі пов'язані між собою, тому окремо ставить їх не має можливості.

В коді програмного забезпечення потрібно підключити бібліотеки розпізнавання об'єктів, бібліотеку комп'ютерного зору. Для цього використовують наступні команди:

```
import cvlib as cv
from cvlib.object_detection import draw_bbox
import cv2
import sys
```

Команда `draw_bbox` потрібна для того, щоб виділяти розпізнані об'єкти за допомогою прямокутників.

Далі відкриваємо доступ до камери та створюємо об'єкт відезахвату з параметром 0 – тобто камера вибирається за змовчуванням:

```
video = cv2.VideoCapture(0)
```

Далі потрібно розробити нескінченний цикл `while (True)`, у нас це:

```
while video.isOpened() ,
```

де `video.isOpened` перевіряє, чи доступний кадр з відеокамери.

В разі, якщо ні, то програма зупиняється:

```
if not status:
    break
```

Читання кадру з камери відбувається за допомогою команди:

```
status, frame = video.read(),
```

де кадр буде зберігатися у змінну status.

Для застосування функції виділення об'єктів використовуються наступні команди:

```
bbox, label, conf = cv.detect_common_objects(frame, confidence=0.25,  
model='yolov3')
```

Тут отримуються 3 параметри:

- `bbox` – прямокутник, яким обмежений виявлений об'єкт;
- `label` – клас, до якого відноситься виявлений об'єкт
- `conf` – параметр впевненості в розпізнаванні; впевненість записує параметри мережі у всіх локальних мінімумах, відвіданих на етапі оптимізації однієї нейронної мережі і служить для отримання набору базових учнів (тобто знімків) з порівняльною в середньому прогностичної сили, які, однак, будуть генерувати кілька різних прогнози для даного випадку.

Також тут використовується функція детектування об'єктів `cv.detect_common_objects`, параметрами якої є:

- `frame` – отриманий кадр з камери;
- `confidence=0.25` – мінімальний коефіцієнт довіри складає 25%;
- `model='yolov3'` – використовується модель yolov3 для розпізнавання.

Потім виводимо інформацію за допомогою команди `print(bbox, label, conf)`.

Для виводу рамки з границями об'єктів використано функцію

```
out = draw_bbox(frame, bbox, label, conf, write_conf=True)
```

Далі виводимо результати на екран за допомогою команди бібліотеки комп'ютерного зору:

```
cv2.imshow("Real-time object detection", out)
```

Далі прописуємо команду виходу з нескінченного циклу по натисканню клавiші Q на клавіатурі:

```
if cv2.waitKey(1) & 0xFF == ord('q'):  
    break
```

Останньою дією є звільнення джерел:

```
video.release()  
cv2.destroyAllWindows()
```

Результат роботи програми представлено на рисунку 3.4

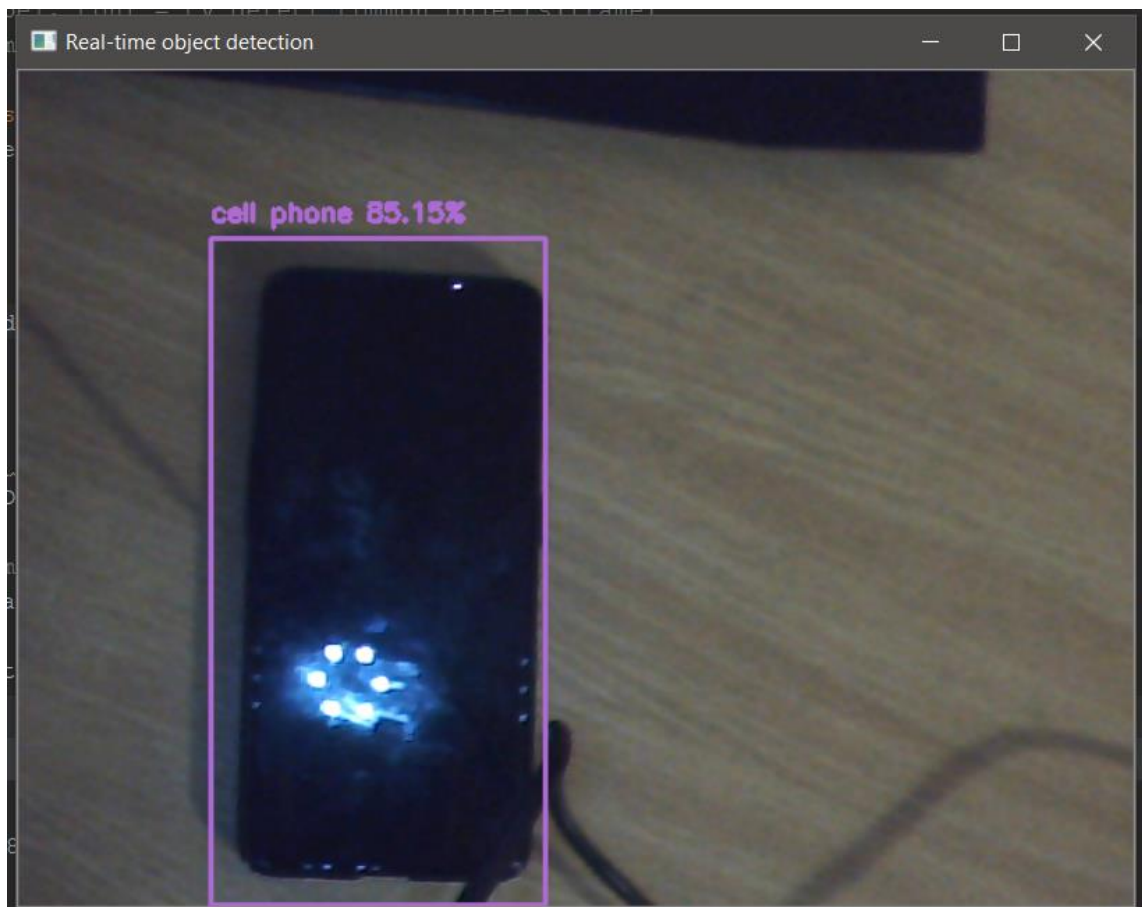


Рисунок 3.4 – Результат роботи програми

### **3.4 Висновки до третього розділу**

Під час виконання атестаційної роботи у данному розділі було виконано:

- обрано технології для створення програмного забезпечення;
- створено алгоритм роботи;
- розроблено програмне забезпечення.

## 4 ПРОВЕДЕННЯ ЕКСПЕРИМЕНТАЛЬНИХ ДОСЛІДЖЕНЬ З РОЗРОБЛЕНИМ ПРОГРАМНИМ ЗАБЕЗПЕЧЕННЯМ

### 4.1 Експеримент зі швидкості навчання класифікатору

Для проведення експериментів з якості ідентифікації було розроблено ще один класифікатор для виділення ознак на основі методів Віоли-Джонса з виділенням ознак Хаара.

В якості об'єкту ідентифікації було обрано деталь типу корпус на прикладі мобільного телефону.

Було зібрано 300 зображень і проведено навчання класифікаторів методами нейронних мереж та методом Хаара. Це навчання було розроблено для 30, 150 та 300 зразків. З результатами ідентифікації можна ознайомитися в таблиці 4.1 та рис. 4.1.

Таблиця 4.1 Результати швидкості навчання класифікатору

Кількість зображень	Нейронна мережа	Класифікатор Хаара
30	18 хв.	17 хв.
150	43 хв.	41 хв.
300	92 хв.	78 хв.

Як видно з графіку, навчання класифікатору за ознаками Хаара швидше для будь-якої кількості зображень, але якість ідентифікації за допомогою нейронної мережі має кращу якість.

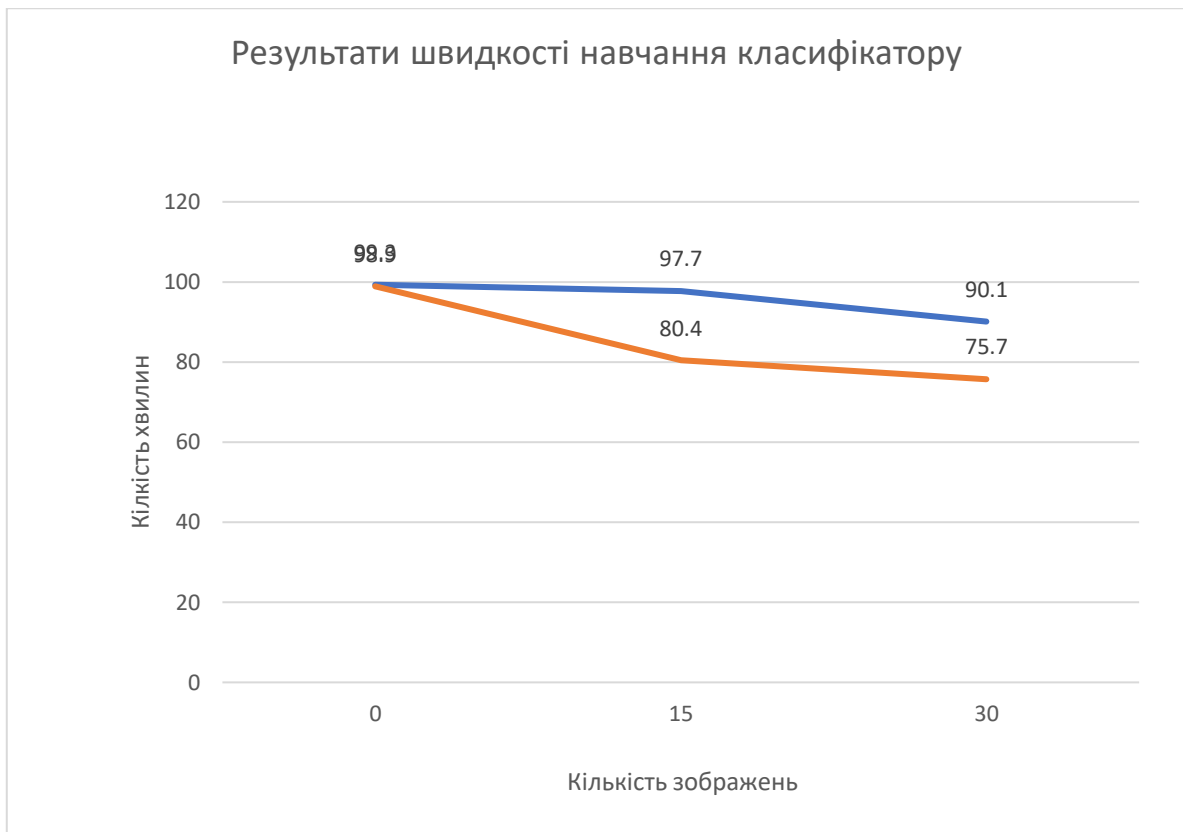


Рисунок 4.1 – Результати швидкостей навчання класифікаторів

#### 4.2 Експеримент з ідентифікації деталей типу корпус для різного рівня освітлення

Проведемо експеримент з ідентифікації корпусу мобільного телефону. По-перше, проведемо такий з різним рівнем освітлення (без, 1 та 3 лампи). Результати роботи класифікаторів наведено у таблиці 4.2, рис. 4.2.

Таблиця 4.2 – Результат ідентифікації корпусу за допомогою класифікаторів для різного рівня освітлення

Рівень освітлення	Нейронна мережа	Класифікатор Хаара
Без ламп	35%	0
1 лампа	87,5%	69,5%
3 лампи	98,7%	97,3%

Як бачимо з результатів, при нормальному рівні освітлення (ідеальні умови) обидва класифікатори показують дуже високий рівень ідентифікації.

При зміні показників яскравості з 3 ламп на 1 обидва класифікатори знизили свої показники, при чому класифікатор нейронної мережі знизив показники на 11,2%, а класифікатор Хаара на цілих 27,8%.

Зміна показників яскравості з відсутністю ламп призвела до того, що за допомогою природнього освітлення класифікатор Хаара не виявив жодного об'єкту, а класифікатор нейронної мережі отримав деякий результат, який дорівнює 35%.

З цього можна зробити висновок, що класифікатор, створений за допомогою нейронної мережі показує кращі результати при низькому рівні освітлення.

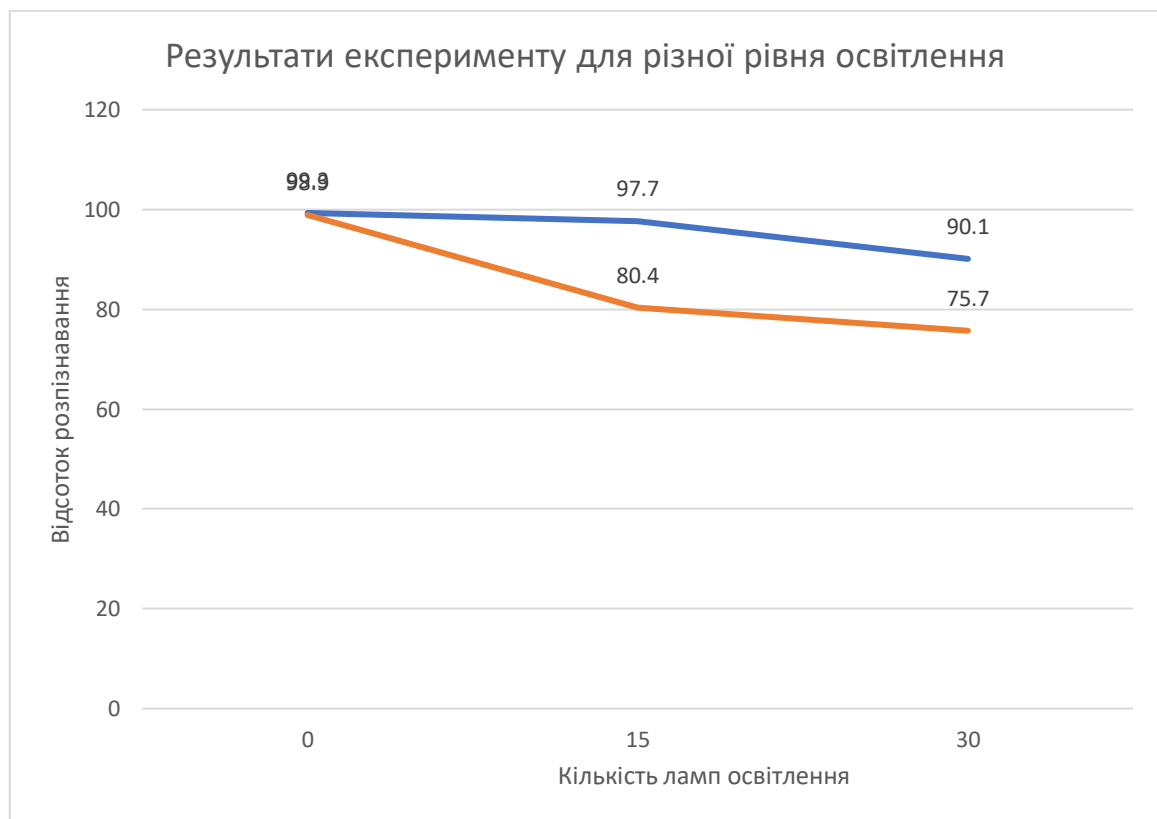


Рисунок 4.2 – Результат ідентифікації корпусу за допомогою класифікаторів для різного рівня освітлення

### 4.3 Експеримент з ідентифікації деталей типу корпус для різного куту повороту

Проведемо експеримент з ідентифікації корпусу мобільного телефону для різних кутів нахилу, так як на конвеєрній лінії деталі не завжди можуть йти в горизонтальному чи вертикальному положенні. Експеримент буде проводитися для ідеального рівня освітлення. Його результати записані в таблицю 4.3 та представлено на рис. 4.3.

Таблиця 4.3 – Експерименти з ідентифікації при різних кутах повороту деталей

Кут нахилу об'єкту	Нейронна мережа	Класифікатор Хаара
0	99.3	98.9
15	97.7	80.4
30	90.1	75.7
45	80.9	30.7
60	90.4	0
75	80.4	40.8
90	98.2	90.2

Як видно з результатів проведення експериментів, для завдань, коли об'єкт ідентифікації знаходиться під різними кутами повороту, якість розпізнавання класифікатором Хаара не є доцільним, тому що втрати в якості ідентифікації та розпізнавання доходять майже до 100%, а також показало, що результати ідентифікації та розпізнавання за допомогою класифікатору нейронної мережі майже завжди показують високі результати при ідеальних умовах для розпізнавання.



Рисунок 4.3 – Результати експерименту для різного кута повороту деталі

#### 4.4 Висновки до другого розділу

Під час виконання атестаційної роботи у четвертому розділі були розглянуті такі питання:

- експеримент зі швидкості навчання класифікатору;
- експеримент з ідентифікації деталей типу корпус для різного рівня освітлення;
- експеримент з ідентифікації деталей типу корпус для різного кута повороту.

## 5 ОХОРОНА ПРАЦІ

### 5.1 Аналіз умов праці на робочому місці

Атестаційна робота розроблялася в лабораторії розміром: довжина 7 м., ширина 6,6 м., висота 3 м., площа вікон 6 м<sup>2</sup>., площа двері 6 м<sup>2</sup>. У приміщенні працюють 6 осіб, розташовано 6 комп'ютерів. Площа приміщення становить 46,2 м<sup>2</sup>, об'єм – 138,6 м<sup>3</sup>. На одне робоче місце припадає 7,7 м<sup>2</sup> і 23,1 м<sup>3</sup>.

Схематичне представлення системи «Л-М-С» (рис 5.1) дозволяє проаналізувати вплив шкідливих факторів на організм людини. Елементи системи можна розділити на такі функціональні частини:

«Людина» - працівник виробництва:

– Л1 – людина, керуюча “машиною” для виконання основного завдання системи - написання програми для робота;

– Л2 – людина, як біологічний об'єкт, який безпосередньо впливає на виробниче середовище (споживання кисню, тепло і волого виділення);

– Л3 – людина, яка розглядається з точки зору його психофізіологічного стану.

«Машина» – обладнання в приміщенні:

– М1 – машина, що виконує основну технологічну функцію (ПЕОМ);

– М2 – машина, що виконує функції аварійного захисту - (обладнання, що запобігає виникненню небезпечних факторів);

– М3 – машина, що впливає на стан виробничого середовища і людини - (вплив, обумовлене шкідливими виробничими факторами).

«Середовище» – виробниче середовище.

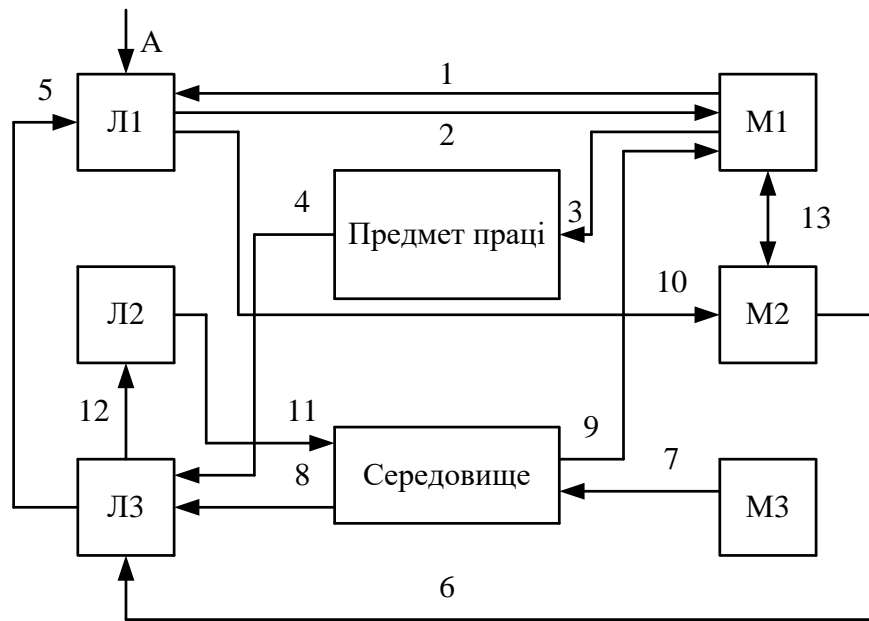


Рисунок 5.1 – Схема системи «Людина-Машина-Середовище»

В системі ЛМС утворюються зв'язку, які наведені в таблиці 5.1.

Таблиця 5.1 – Опис структури ЛМС

Номер зв'язку	Напрямок зв'язку	Зміст зв'язку
1	2	3
1	Л1 – М1	Вплив людини на основне технологічне обладнання (включення комп'ютера, написання програми для керування мобільним роботом)
2	М1 – Л1	Вплив шкідливих факторів, які виробляє обладнання, на психофізіологічний стан працівника (наприклад, підвищений рівень шуму викликає перенапруження слухових аналізаторів, що може призвести до приглухуватості; впливає на серцево-судинну систему, викликає роздратування і швидку стомлюваність; Підвищений рівень шуму на робочому місці несприятливо впливає не тільки на органи слуху, а й на весь організм людини в процесі роботи через центральну нервову систему.

Продовження таблиці 5.1 – Опис структури ЛМС

1	2	3
		У людини послаблюється увага, погіршується пам'ять, що призводить до значного зниження продуктивності праці і зростання кількості помилок у роботі)
3	М1 – ПП	Вплив машини на предмет праці (на технологічному обладнанні безпосередньо проводиться розробка програмного забезпечення для робота)
4	ПП – ЛЗ	Вплив предмета праці на психофізіологічний стан людини (якість предмета праці може викликати як позитивні, так і негативні емоції, останнє може призвести до нервового зриву чи інших негативних наслідків)
5	ЛЗ – Л1	Вплив психофізіологічного стану людини на працездатність людини (якщо людина втомилася, то працездатність падає, втрачається увага до виконуваної роботи, збільшення помилок під час роботи)
6	М2 – ЛЗ	Вплив функцій аварійного захисту на психофізіологічний стан людини (якщо в приміщенні відсутній захист людини від дії небезпечних факторів, то людина може отримати травму, наприклад поразка людини електричним струмом може викликати порушення життєдіяльності функцій організму, тобто втрату свідомості, зупинку дихання або припинення роботи серця)
7	М3 – С	Вплив машини на виробниче середовище (зміна параметрів мікроклімату, підвищений рівень ЕМВ, що опосередковано негативно впливає на психофізіологічний стан людини)
8	С – ЛЗ	Вплив середовища на психофізіологічний стан організму (підвищена вологість або температура повітря в приміщенні, погане або дуже насичене освітлення викликає роздратування, швидку стомлюваність і в результаті зниження працездатності)
9	С – М1	Вплив середовища на роботу машини (збільшена вологість та температура погано впливають на працездатність машини)

Продолження таблиці 5.1 – Опис структури ЛМС

1	2	3
10	Л1 – М2	Вплив людини на функції аварійного захисту (безпечний стан робочого місця - застосування методів захисту від впливу небезпечних виробничих факторів)
11	Л2 - С	Вплив людини як біологічного об'єкта на виробниче середовище (виділення вуглекислого газу, збільшення вологості та температури)
12	Л3 – Л2	Вплив психофізіологічного стану на ступінь інтенсивності біологічних процесів в організмі людини (підвищене емоційне напруження веде до зміни швидкості протікання біохімічних процесів в організмі, що призводить до збільшення потовиділення, прискорене дихання)
13	М1 – М2	Інформація про несправність технологічного обладнання, необхідна для спрацьовування функцій аварійного захисту
	М2 – М1	Аварійний керуючий вплив
А	Зовнішня система управління - ЛІ	Керуюча інформація про технологічний процес з зовнішньої системи управління

## 5.2 Промислова безпека

За ступенем небезпеки ураження електричним струмом згідно з НПАОП 40.1-1.21-98 приміщення належить до класу приміщень без підвищеної небезпеки ураження електричним струмом. Умови , які створюють підвищену і особливу небезпеку (підвищена вологість, струмопровідна пил, струмопровідні підлоги, можливість одночасного дотику до заземлених металоконструкціям будівлі і металевих поверхнях електроприладів) , відсутні.

З метою зниження небезпеки ураження людини електричним струмом проектом передбачається використання таких технічних засобів захисту:

– необхідно проводити контроль ізоляції відповідно до вимог НПАОП 40.1-1.21-98. Контроль проводити між нульовим і фазним провідниками і між фазами. Опір ізоляції не менше 500 кОм на фазу. Контроль проводити не рідше

1 разу на рік при відключеному електроживленні;

– тому що використовується електрична мережа з заземленою нейтраллю напругою до 1000В, то згідно НПАОП 40.1-1.32-01 всі металеві конструкції і частини обладнання, доступні для дотику людини і не мають інших видів захисту, що забезпечують електробезпеку, були занулені. Всі корпуси ЕОМ з'єднані з глухозаземленою нейтраллю джерела живлення за допомогою нульового захисного провідника. Занулення перетворює замикання на корпус в однофазне коротке замикання, в результаті чого спрацьовує максимальна струмовий захист, яка селективно відключає пошкоджену ділянку мережі. Автомат захисту вибирається по струму короткого замикання, час відключення дорівнює 0,2 с. Додатково застосовується повторне заземлення нульового проводу з метою зниження потенціалу занулених корпусів і напруги дотику у випадках обриву нульового проводу.

– проведення інструктажів з техніки безпеки здійснюється відповідно до НПАОП 0.00-4.12-05

Список інструктажів, які необхідно проводити на робочому місці:

– вступний інструктаж – проводиться інженером з охорони праці з усіма працівниками незалежно від освіти та стажу роботи. Даний інструктаж містить у собі ознайомлення з режимом праці та відпочинку даного підприємства, а також з правилами промислової та пожежної безпеки;

– первинний на робочому місці – проводиться на початку виробничої діяльності керівником структурного підрозділу. Включає в себе ознайомлення з небезпечними і шкідливими факторами, які можуть виникати на даному робочому місці, з індивідуальними засобами захисту, що застосовуються на робочому місці, з безпечними прийомами роботи;

– повторний – проводиться один раз на півроку з усіма працівниками за програмою первинного інструктажу.

– позаплановий – проводиться при заміні та модернізації обладнання, при зміні технологічного процесу, при введенні в дію нових стандартів і

правил з охорони праці. Включає в себе ознайомлення з новим обладнанням і повторенням правил промислової безпеки, якщо стався нещасний випадок на виробництві.

– цільовий – проводиться при виконанні разових робіт, не пов'язаних з основним видом діяльності, при ліквідації аварій, надзвичайних ситуацій, катастроф.

## ВИСНОВКИ

Слід зазначити, що використання технологій нейронних мереж в наш час є дуже актуальним завданням, окрема при використанні в Industry 4.0 – це все дозволить значно підвищити продуктивність сучасного приладобудівного виробництва.

Інтеграція машинного навчання та нейронних мереж в різних секторах приладобудівного виробництва зіграла ключову роль у відкритті нових можливостей для підвищення якості життя і забезпечення стійкості нашої екосистеми.

В результаті виконання атестаційної роботи магістра було вирішено наступні питання:

- проведено аналіз технологій, які використовуються на сучасному приладобудівному виробництві;
- розроблено модель ідентифікації деталей для процесу сортування на базі комп'ютерного зору;
- на основі моделі розроблено метод ідентифікації деталей для процесу сортування на базі комп'ютерного зору;
- на основі методу та моделі розроблено програмне забезпечення ідентифікації деталей для процесу сортування на базі комп'ютерного зору;
- проведено експерименти з розпізнавання деталей типу корпус при різних початкових умовах;
- розглянуто питання охорони праці;
- оформлено пояснювальну записку згідно [1] та [2].

Як видно з експериментів, класифікатори Хаара мають гарний ступінь ідентифікації та розпізнавання лише при ідеальних умовах. На відміну від них, технології нейронних мереж справляються зі своїми завданнями краще, хоча на навчання мережі тратиться і більший час.

Виконана робота може знайти продовження при невеликому удосконаленні і використовуватися для розпізнавання.

Данне впровадження на виробництві допоможе скоротити витрати на виробництво та автоматизувати процеси.

Завдяки тому, що воно може виконувати ідентифікацію різних деталей після початку його можна використовувати на різних підприємствах з різними типами та розмірами, якими обумовлені задані параметри.

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. ДСТУ 3008: 2015 Інформація та документація. Звіти у сфері науки і техніки. Структура і правила оформлення. ДП «УкрНДНЦ», 2016. – 31 с.

2. Методичні вказівки з «Розробки й оформлення магістерської атестаційної роботи» для студентів другого (магістерського) рівня вищої освіти галузі знань 15 Автоматизація та приладобудування за спеціальністю 151 Автоматизація та комп'ютерно-інтегровані технології освітні програми: «Автоматизоване управління технологічними процесами», «Комп'ютерно-інтегровані технологічні процеси і виробництва», «Комп'ютеризовані та робототехнічні системи» / Упоряд. І.Ш. Невлюдов, В.В. Косенко, В.В. Євсєєв. – Харків: ХНУРЕ, 2019. – 55 с.

3. A guide to modern factory automation and Industry 4.0 in manufacturing. [Електронний ресурс]. – Режим доступу : <https://www.information-age.com/guide-modern-factory-automation-industry-4-0-manufacturing-123490992/>(дата звернення: 20.10.2020)

4. Industry 4.0: Current Status and Future Trends. [Електронний ресурс]. – Режим доступу : <https://www.intechopen.com/books/industry-4-0-current-status-and-future-trends/industry-4-0-current-status-and-future-trends> (дата звернення: 20.10.2020)

5. 11 Trends That Will Dominate Manufacturing in 2021 [Електронний ресурс]. – Режим доступу : <https://global.hitachi-solutions.com/blog/top-manufacturing-trends> (дата звернення: 20.10.2020)

6. Wang, Lin & He, Jinfeng & Xu, Songjie. The Application of Industry 4.0 in Customized Furniture Manufacturing Industry. *MATEC Web of Conferences*. 100. 03022. 10.1051/matecconf/201710003022. Viena, Austria, 2017

7. The Role Of Conveyors In Industry 4.0 Manufacturing Environments. [Електронний ресурс]. – Режим доступу : <https://www.conovey.com/the-role-of-conveyors-in-industry-4-0-manufacturing-environments/> (дата звернення: 21.10.2020)

8. Fedorko, Gabriel.. Implementation of Industry 4.0 in the belt conveyor transport. MATEC *Web of Conferences*. 263. 01001. 10.1051/mateconf/201926301001. Podbanske, Slovakia, 2019

9. Sortation System Types, Applications & Benefits. [Электронный ресурс]. – Режим доступа : <https://www.conveyco.com/pros-cons-popular-sortation-systems/> (дата звернення: 21.10.2020)

10. Dabade, Sanjay. Automatic Sorting Machine Using Conveyor Belt / Sanjay Dabade. International Journal of Innovative and Emerging Research in Engineering. 2. 66-70.

11. Peršak Tadej Vision-Based Sorting Systems for Transparent Plastic Granulate / Tadej Peršak,, Branka Viltušnik , Jernej Hernavs and Simon Klančnik, University of Maribor, Velenje, Slovenia, 2019.

12. The difference between computer vision and machine vision . [Электронный ресурс]. – Режим доступа : <https://www.clearviewimaging.co.uk/blog/the-difference-between-computer-vision-and-machine-vision> (дата звернення: 23.10.2020)

13. Top computer vision trends for the modern enterprise [Электронный ресурс]. – Режим доступа : <https://www.analyticsinsight.net/top-computer-vision-trends-modern-enterprise/> (дата звернення: 23.10.2020)

14. Ing. Kuric I Application of artificial intelligence for the implementation of Industry 4.0 concept / prof. Dr. Ing. Kuric I., Ing. Zajačko I., PhD., Ing. Císar M., PhD., Tomáš Gál International scientific journal "Industry 4.0" year III, Issue 3, P.P. – 2018– pp.120-123

15. Hernavs, Deep learning in Industry 4.0 / Hernavs, Jernej & Ficko, Mirko & Berus, Lucijano & Rudolf, Rebeka & Klančnik, Simon – BRIEF. Journal of Production Engineering. 21. 1-5. 10.24867/JPE-2018-02-001. – 2018

16. Overview of modern computer vision tools [Электронный ресурс]. – Режим доступа : <https://svitla.com/blog/overview-of-modern-computer-vision-tools> (дата звернення: 24.10.2020)

17. Гарсия Глория Буэно. Обработка изображений с помощью OpenCV/ Глория Буэно Гарсия, Оскар Дениз Суарес, Хосе Луис Эспиноса Аранда, Хесус Салидо Терсеро, Исмаэль Серрано Грасиа, Ноэлия Валлез Энано // пер. с англ. Слинкин А. А. – М.: ДМК Пресс, 2016. – 210 с.: ил.

18. Hands-On Machine Learning with Scikit-Learn and TensorFlow by Aurélien Géron Copyright Aurélien Géron. Published by O'Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472 – 2017

19. CUDA: Compiling and optimizing for a GPU platform – International Conference on Computational Science, ICCS 2012

20. CUDA: Compiling and optimizing for a GPU platform./ [Chakrabarti, Gautam & Grover, Vinod & Aarts, Bastiaan & Kong, Xiangyun & Kudlur, Manjunath & Lin, Yuan & Marathe, Jaydeep & Murphy, Mike & Wang, Jian-Zhong]. Procedia Computer Science. 9. – 2012 – 340 pp.

21. Theano: a CPU and GPU math expression compiler./ [Bergstra, James & Breuleux, O. & Bastien, Frederic & Lamblin, Pascal & Pascanu, Razvan & Desjardins, G. & Turian, J. & Warde-Farley, David & Bengio, Y.. (2010)] Proceedings of the Python for Scientific Computing Conference (SciPy). Vol. 4.

22. Джулли Антонио. Библиотека Keras – инструмент глубокого обучения. Реализация нейронных сетей с помощью библиотек Theano и Tensorflow / Антонио Джулли, Суджит Пал. – М.: ДМК Пресс, 2018. – 294 с.

23. Theano: A Python framework for fast computation of mathematical expressions (The Theano Development Team) arXiv:1605.02688v1 [cs.SC] 9 May 2016

24. Leea Hagyeong, Introduction to convolutional neural network using Keras; an understanding from a statistician / Hagyeong Leea , Jongwoo Song //Communications for Statistical Applications and Methods 2019, Vol. 26, No. 6, 591–610

25. Deep Learning with TensorFlow 2 and Keras Second Edition / Antonio Gulli Amita Kapoor Sujit Pal /Published by Packt Publishing Ltd. Livery Place 35 Livery Street Birmingham B3 2PB, UK

26. Ceren Melek, (2019). Object Detection in Shelf Images with YOLO. /  
Melek, Ceren & Sönmez, Elena & Varlı Albayrak, Songul.  
10.1109/EUROCON.2019.8861817. – 2019 127 pp.