

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

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

бакалавр

(назва освітнього ступеня)

на тему: *Комп'ютеризована система багатокамерного відеозапису
динамічних об'єктів*

Виконав: студент 4 курсу, групи СІ-42
спеціальності 123 «Комп'ютерна інженерія»

(шифр і назва спеціальності)

(підпис)

(прізвище та ініціали)

Керівник

(підпис)

(прізвище та ініціали)

Нормоконтроль

(підпис)

(прізвище та ініціали)

Завідувач кафедри

(підпис)

Осухівська Г.М.

(прізвище та ініціали)

Рецензент

(підпис)

(прізвище та ініціали)

Тернопіль
2025

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)

Кафедра комп'ютерних систем та мереж
(повна назва кафедри)

ЗАТВЕРДЖУЮ
Завідувач кафедри
Осухівська Г.М.
(підпис) (прізвище та ініціали)
«27» січня 2025 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

на здобуття освітнього ступеня бакалавр
(назва освітнього ступеня)

за спеціальністю 123 «Комп'ютерна інженерія»
(шифр і назва спеціальності)

студента Чорняк Олексій Вікторович
(прізвище, ім'я, по батькові)

1. Тема роботи Комп'ютеризована система багатокамерного відеозапису динамічних об'єктів

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

Затверджені наказом ректора від « 27 » січня 2025 року № 4/7-53

2. Термін подання студентом завершеної роботи 17.06.2025

3. Вихідні дані до роботи Технічне завдання

4. Зміст роботи (перелік питань, які потрібно розробити)

Вступ. 1. Аналіз технічного завдання

2. Проектна частина

3. Практична частина

4. Безпека життєдіяльності, основи охорони праці.

Висновки

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

1.

2.

3.

4.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
<i>Безпека життєдіяльності, основи охорони праці</i>			

7. Дата видачі завдання 27.01.2025 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1.	<i>Розробка технічного завдання.</i>	<i>27.01 – 02.02</i>	<i>Викон.</i>
2.	<i>Аналіз задачі</i>	<i>2.05 – 5.05</i>	<i>Викон.</i>
3.	<i>Ознайомлення з документацією ESP32 sam</i>	<i>8.05 – 10.05</i>	<i>Викон.</i>
4.	<i>Розробка та збір комп'ютеризованої системи</i>	<i>11.05 – 22.05</i>	<i>Викон.</i>
5.	<i>Безпека життєдіяльності, основи охорони праці</i>	<i>23.05 – 25.05</i>	<i>Викон.</i>
6.	<i>Оформлення пояснювальної записки і графічного матеріалу</i>	<i>26.05 – 02.06</i>	<i>Викон.</i>
7.	<i>Перевірка на академічний плагіат, перевірка керівником та консультантами</i>	<i>9.06 – 15.06</i>	
8.	<i>Попередній захист кваліфікаційної роботи бакалавра</i>	<i>16.06 – 22.06</i>	<i>Викон.</i>
9.	<i>Захист кваліфікаційної роботи бакалавра</i>	<i>24.06</i>	<i>Викон.</i>

Студент

(підпис)

(прізвище та ініціали)

Керівник роботи

(підпис)

(прізвище та ініціали)

АНОТАЦІЯ

Чорняк О.В. Комп'ютеризована система багатокамерного відеозапису динамічних об'єктів: робота на здобуття кваліфікаційного ступеня бакалавра: спец. 123 — комп'ютерна інженерія. Тернопіль: Тернопільський національний технічний університет імені Івана Пулюя, 2025.

Ключові слова: комп'ютерний зір, розпізнавання обличчя, OpenCV, глибока нейронна мережа (DNN), Haar Cascade, ESP32cam, Arduino Nano , Python.

У кваліфікаційній роботі втілено систему розпізнавання обличчя в реальному часі з застосуванням бібліотеки OpenCV. Зображення обробляються з відеопотоку, а знайдені обличчя аналізуються для подальшого управління положенням камери. Координати об'єкта надсилаються на мікроконтролери Arduino Nano та ESP32, котрі управляють сервомоторами. Зв'язок між програмою на Python та мікроконтролерами відбувається через послідовний інтерфейс. Система демонструє інтеграцію комп'ютерного зору з вбудованими пристроями для автоматизованого стеження об'єктів.

ANNOTATION

Chorniak O.V. Computerized Multi-Camera Video Recording System for Dynamic Objects: Bachelor's Graduation Thesis: speciality 123 — computer engineering. Ternopil: Ternopil Ivan Puluj National Technical University, 2025.

Keywords: computer vision, face recognition, OpenCV, deep neural network (DNN), Haar Cascade, ESP32-CAM, Arduino Nano, Python.

In this qualification work, a real-time face recognition system was implemented using the OpenCV library. Images are processed from a video stream, and the detected faces are analyzed to control the position of the camera. The coordinates of the object are transmitted to Arduino Nano and ESP32 microcontrollers, which operate the servo motors. Communication between the Python-based software and the microcontrollers is established via a serial interface. The system demonstrates the integration of computer vision with embedded devices for automated object tracking.

ЗМІСТ

ВСТУП	9
РОЗДІЛ 1 АНАЛІЗ ТЕХНІЧНОГО ЗАВДАННЯ	11
1.1 Аналіз області застосування комп'ютерної системи	11
1.2 Аналіз вимог до комп'ютерної системи	13
1.3 Аналіз існуючих аналогів та переваги комп'ютерної системи	15
РОЗДІЛ 2 ПРОЄКТНА ЧАСТИНА.....	19
2.1 Опис логіки роботи комп'ютерної системи	19
2.1.1 Передача відеопотоку	20
2.1.2 Обробка кадру за допомогою OpenCV та обчислення координат	20
2.1.3 Передача команди повороту	21
2.2 Обґрунтування вибору апаратного забезпечення компютеризованої системи.....	22
2.2.1 Обґрунтування вибору відеокамери	22
2.2.2 Обґрунтування вибору сервоприврду	25
2.2.3 Опис розробленого штативу	28
2.2.4 Структурна схема роботи компютеризованої системи	30
2.3 Обґрунтування вибору програмного забезпечення компютеризованої системи.....	32
2.3.1 Схема алгоритму роботи компютеризованої системи	34
РОЗДІЛ 3 ПРАКТИЧНА ЧАСТИНА.....	38
3.1 Реалізація практичних рішень	38

					КС КРБ 123.244.00.00 ПЗ							
Змн.	Арк.	№ докум.	Підпис	Дата	Комп'ютеризована система багатокамерного відеозапису динамічних об'єктів				Літ.	Арк.	Аркушів	
Розроб.		Чорняк О.В.									6	
Перевір.												
Реценз.												
Н. Контр.												
Затверд.		Осухівська Г.М.										
					ТНТУ, каф. КС, гр. СІ-42							

3.1.1	Налаштування середовища розробки для програмування ESP32 sam..	38
3.1.2	Налаштування середовища розробки для роботи з OpenCV.....	41
3.1.3	Налаштування середовища розробки для Arduino Nano.....	42
3.2	Тестування комп'ютеризованої системи	44
РОЗДІЛ 4 БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ		48
4.1	Менеджмент безпеки.....	48
4.2	Електробезпека на будівельному майданчику.....	50
ВИСНОВКИ.....		52
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....		54
Додаток А Технічне завдання		
Додаток Б Лістинг		

					КС КРБ 123.244.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		7

СПИСОК СКОРОЧЕНЬ

CV – Computer Vision

DNN – Deep Neural Network

Haar – Haar Cascade Classifier

FPS – Frames Per Second

UART – Universal Asynchronous Receiver-Transmitter

IDE – Integrated Development Environment

USB – Universal Serial Bus

					КС КРБ 123.244.00.00 ПЗ	Арк.
						8
Змн.	Арк.	№ докум.	Підпис	Дата		

ВСТУП

В умовах сучасного, стрімкого поступу цифрових технологій, комп'ютерне бачення та автоматизовані системи відеоспостереження набирають дедалі більшої ваги у забезпеченні безпеки, автоматизації виробничих процесів, телемедицині, освіті, кіберспорті, індустрії розваг та журналістиці. Серед головних задач, що розв'язуються такими системами, є розпізнавання та відстеження об'єктів, що змінюються в реальному часі, зокрема людського обличчя. Саме ця задача має надзвичайну актуальність, адже вона застосовується в широкому спектрі сфер: від систем відеоспостереження до інтерактивних мультимедійних рішень.

Зокрема, зростаючий інтерес до персоналізованих відеосистем вимагає розробки мобільних, недорогих та енергоощадних пристроїв, що можуть забезпечити високоякісний відеозапис з функцією динамічного фокусування на об'єкті. Стає очевидною необхідність у гнучких, адаптивних системах, які здатні відслідковувати положення обличчя чи будь-якого іншого об'єкта та відповідно коригувати напрямок зйомки без залучення користувача. Це дає змогу розширити функціональність систем відеоспостереження, покращити ефективність дистанційної фіксації подій, а також створювати нові формати мультимедійного контенту. Одним із перспективних напрямів є використання багатокамерних систем чи комп'ютеризованих платформ на базі мікроконтролерів, котрі можуть інтегруватися з класичними або мобільними камерами (наприклад, смартфонами) та самостійно стежити за об'єктами в кадрі. У контексті цієї дипломної роботи йдеться про розробку та реалізацію комп'ютеризованої системи, що забезпечує багатокамерний відеозапис та автоматичне відстеження руху об'єкта зйомки (людського обличчя) з можливістю повороту камери в напрямку об'єкта.

Особливої актуальності така система набуває для малих продакшн-студій, влогерів, освітніх платформ та наукових експериментів, де важливими є мобільність, автономність, легкість використання та відносно невелика

					КС КРБ 123.244.00.00 ПЗ	Арк.
						9
Змн.	Арк.	№ докум.	Підпис	Дата		

вартість компонентів. Зважаючи на це, впровадження інтелектуальної системи на базі модулів ESP32-CAM, Arduino та бібліотеки OpenCV є доцільним і своєчасним технічним рішенням, яке здатне задовольнити потреби у компактному, ефективному та адаптивному відеозаписі динамічних сцен. Отже, тема цієї кваліфікаційної роботи бакалавра є актуальною, враховуючи високий попит на інтелектуальні, доступні за ціною та енергоощадні системи відеозапису, здатні адаптуватися до змін положення об'єкта в реальному часі без втручання оператора.

					КС КРБ 123.244.00.00 ПЗ	Арк.
						10
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 1 АНАЛІЗ ТЕХНІЧНОГО ЗАВДАННЯ

1.1 Аналіз області застосування комп'ютерної системи

Сучасна інженерія, зосереджуючись на автоматизації, робототехніці, комп'ютерному зору та рішеннях IoT (Інтернету Речей), широко використовує вбудовані системи. Вони являють собою мікропроцесорні пристрої або мікроконтролерні платформи, спроектовані для виконання спеціалізованих задач у складі більш складних технічних комплексів. Ці системи об'єднують в собі апаратне забезпечення та програмне забезпечення і часто функціонують автономно. Їх основною перевагою є здатність взаємодіяти з навколишнім середовищем через сенсори, швидко реагувати на сигнали та управляти виконавчими пристроями без потреби в складному інтерфейсі для взаємодії з користувачем. Однією з ключових сфер застосування вбудованих систем є автоматизована відеозйомка та відстеження об'єктів у кадрі. У відповідь на зростаючий попит на збір, аналіз та трансляцію візуальної інформації, зокрема в освіті, на масових заходах, у сфері охорони та розваг, стають необхідними доступні та гнучкі рішення, які здатні функціонувати самостійно. Це можуть бути як стаціонарні, так і мобільні системи, розроблені для зйомки людей, тварин, транспортних засобів чи будь-яких інших рухомих об'єктів. У межах цієї кваліфікаційної роботи передбачається розробка комп'ютеризованої системи багатокамерного відеозапису динамічних об'єктів з функцією автоматичного стеження за обличчям людини. Суть системи полягає у розпізнаванні обличчя в режимі реального часу за допомогою вбудованої камери, визначенні положення обличчя в кадрі та керуванні виконавчим механізмом, що обертає знімальний пристрій у напрямку виявленого об'єкта.

					КС КРБ 123.244.00.00 ПЗ		
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>			
Розроб.		Чорняк О.В.			Аналіз технічного завдання		
Перевір.		Тиш Є.В.					
Реценз.							
Н. Контр.		Луцик Н.С.					
Затверд.		Осухівська Г.М.					
					<i>Літ.</i>	<i>Арк.</i>	<i>Аркушів</i>
						11	
					ТНТУ, каф. КС, гр. СІ-42		

Таким чином, реалізується функціональність, яка імітує дії відеооператора – камера постійно слідкує за людиною, утримуючи її в центрі кадру, що забезпечує плавний та інформативний відеозапис.

Конструктивно пристрій планується виконати у вигляді поворотного штатива зі закріпленням смартфона. Додатково на штативі буде розташовано модуль ESP32-CAM — вбудовану платформу з камерою та підтримкою Wi-Fi. Цей модуль здійснюватиме зйомку та передаватиме відео на персональний комп'ютер, де відбуватиметься основна обробка інформації. Вибір ESP32-CAM зумовлено її компактністю, низьким енергоспоживанням, вбудованою підтримкою бездротових протоколів передачі даних та можливістю обробки зображень на платі.

Важливо відзначити, що ESP32-CAM не є основним пристроєм для запису. Зйомка відео здійснюватиметься смартфоном, закріпленням на рухомій частині платформи. Камера на ESP32 виконує функцію "сенсора зору" – вона аналізує простір перед пристроєм та допомагає визначити положення об'єкта. Це дозволяє поєднати високоякісне зображення зі смартфона з аналітичними можливостями ESP32 та ПК.

Обробка кадрів здійснюватиметься на комп'ютері, де працюватиме Python-скрипт з використанням бібліотеки OpenCV. Цей скрипт отримуватиме відеопотік з ESP32-CAM, визначатиме обличчя в кадрі, обчислюватиме їх центр, а потім прийматиме рішення про необхідність повороту знімального пристрою. У разі виявлення зміщення обличчя відносно центру кадру, скрипт надсилатиме відповідні координати на мікроконтролер Arduino Nano, який керуватиме сервоприводом. В результаті знімальна платформа буде повертатися таким чином, щоб обличчя залишалося в центрі кадру. Це рішення дозволяє автоматизувати відеозйомку без постійного втручання людини. Система буде ідеальною для запису лекцій, онлайн-зустрічей, навчальних відео, стримінгу блогів, де необхідно зосередити увагу глядача на спікері, а також для домашнього відеоспостереження, відстеження руху дітей

					КС КРБ 123.244.00.00 ПЗ	Арк.
						12
Змн.	Арк.	№ докум.	Підпис	Дата		

або тварин у приміщенні.

Проектована система буде прив'язана до комп'ютера, який виконуватиме роль головного обчислювального вузла. Саме на ньому відбуватиметься обробка зображень, реалізація алгоритмів виявлення та прийняття рішень. Така архітектура обрана з огляду на те, що ESP32-CAM має обмежену продуктивність та не здатна ефективно працювати з нейронними мережами в реальному часі. Водночас, живлення Arduino Nano та ESP32-CAM буде здійснюватися безпосередньо через USB-порти ПК, що усуне потребу в додаткових джерелах живлення та сприятиме компактності. У перспективі ця система може бути розширена до багатокамерної. При підключенні кількох ESP32-CAM до одного ПК, можливо буде реалізувати сценарії, де кожна камера виконує окрему роль: одна – веде об'єкт, інша – фіксує загальний план, третя – знімає з іншого ракурсу. Така функціональність може бути корисною в освітніх закладах, на сценах, у студіях, лабораторіях, під час досліджень та ін. Серед переваг запропонованої архітектури – її модульність та гнучкість. Кожен компонент системи може бути замінений, доповнений або модифікований без змін загальної логіки роботи. Відкритість використаних технологій (ESP32, Arduino, OpenCV, Python) дозволяє користувачам легко змінювати або адаптувати систему під власні потреби, без потреби в дорогому ліцензійному ПЗ.

Отже, запропонована вбудована система поєднає сучасні технічні підходи, доступність реалізації, відкриту архітектуру та потенціал масштабування, що робить її особливо ефективною в освітніх, демонстраційних, побутових та експериментальних проектах.

1.2 Аналіз вимог до комп'ютерної системи

У процесі розробки комп'ютеризованої багатокамерної системи відеозапису динамічних об'єктів необхідно взяти до уваги ряд вимог, які

					КС КРБ 123.244.00.00 ПЗ	Арк.
						13
Змн.	Арк.	№ докум.	Підпис	Дата		

забезпечать її надійне функціонування в умовах освітнього, домашнього або експериментального середовища. Ці вимоги охоплюють функціональні, технічні та конструктивні аспекти. З функціональної точки зору, система має виявляти людину у полі зору камери та автоматично відстежувати її положення. Обличчя, розпізнане за допомогою алгоритмів комп'ютерного зору, фіксуватиметься у відеокадрі, після чого система визначить його координати відносно центру зображення. На основі цієї інформації обчислюватиметься кут, на який необхідно повернути знімальний пристрій. Передбачено, що поворотна платформа автоматично адаптуватиметься до положення обличчя без втручання людини, що забезпечить ефект "розумної камери".

Для реалізації цієї логіки буде використано модуль ESP32-CAM, який виконуватиме функцію відеоспостереження та надсилатиме потік даних у локальну мережу. Персональний комп'ютер, який виступатиме обчислювальним вузлом, зчитуватиме зображення, розпізнаватиме обличчя за допомогою бібліотеки OpenCV та здійснюватиме обробку координат. Далі інформація передаватиметься на мікроконтролер Arduino Nano, який керуватиме сервомеханізмом для фізичного повороту камери чи смартфона. У технічному плані система вимагатиме наявності PSRAM-пам'яті в ESP32 для якісної трансляції JPEG-кадрів, стабільного Wi-Fi-з'єднання, достатньої обчислювальної потужності комп'ютера для роботи з бібліотеками Python, а також точної роботи сервоприводу з великим кутом обертання та плавністю ходу. Дуже важливо забезпечити злагоджену взаємодію всіх компонентів, щоб уникнути зависань, затримок або втрат даних під час обміну інформацією між пристроями.

Загальний принцип функціонування передбачає таку послідовність: камера надсилає потік на ПК, де відбувається розпізнавання обличчя; якщо об'єкт вийшов за межі допустимого центру — система розраховує необхідний поворот; після чого Arduino отримує команду і активує механічний привід.

					КС КРБ 123.244.00.00 ПЗ	Арк.
						14
Змн.	Арк.	№ докум.	Підпис	Дата		

Очікувана затримка між виявленням обличчя та реакцією сервоприводу не перевищуватиме однієї секунди, що є прийнятним для задач освітнього рівня чи аматорського використання. Окрему увагу буде приділено конструктивному оформленню пристрою. Усі елементи планується розмістити у спеціально змодельованому корпусі, створеному в CAD-середовищі та надрукованому на 3D-принтері. Конструкція матиме відсіки для надійного кріплення основних модулів – камери, контролера, приводу – а також фіксоване місце для смартфона або іншого записуючого пристрою. Такий підхід забезпечить не лише зручність у збиранні та транспортуванні, а й дозволить легко модифікувати чи масштабувати систему в майбутньому, у разі розширення функціоналу або зміни конфігурації.

Загалом, сформульовані вимоги спрямовані на створення надійної, компактної та легко адаптованої системи, здатної ефективно виявляти та супроводжувати об'єкти без потреби зовнішнього управління. Врахування зазначених параметрів стане основою для побудови працездатного, функціонального та модернізованого рішення.

1.3 Аналіз існуючих аналогів та переваги комп'ютерної системи

Одним з найближчих до нашого винаходу за спектром можливостей комерційних продуктів є веб-камера OBSBOT Tail Air (рис.1.1). Її позиціюють як "інтелектуальну" систему, що автоматично відстежує об'єкт у кадрі. OBSBOT Tail Air – це сучасний гаджет з вмонтованим процесором, котрий опрацьовує картинку в реальному часі. Пристрій може автономно розпізнавати обличчя, автоматично обирати кадр, збільшувати або зменшувати зображення, слідкувати за об'єктами та вести зйомку в різноманітних форматах з високою роздільною здатністю. Найчастіше її використовують у відеоблогах, стрімінгу, під час відеоконференцій та для

					КС КРБ 123.244.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		15

створення креативного контенту, де потрібна динамічна, плавна та інтелектуальна робота камери, яка не потребує присутності оператора.



Рисунок 1.1 – OBSBOT Tail Air

Основна задача цієї камери, подібно до нашої розробки, полягає у захопленні рухомих об'єктів у полі зору, а далі – відстеженні їх траєкторії. OBSBOT Tail Air використовує штучний інтелект, заснований на моделях комп'ютерного зору, для точного розпізнавання об'єктів та плавного автоматичного керування камерою, завдяки інтегрованим моторам і високоточній системі стабілізації. Камера здатна розпізнавати не тільки обличчя, але й жести, рухи рук, змінювати фокус відповідно до руху голови, а також реагувати на попередньо задані команди користувача. Подібно до нашої системи, вона володіє можливістю автоматичного керування обертанням камери, щоб утримувати об'єкт у кадрі, незалежно від його переміщень.

Однак, попри технологічну досконалість OBSBOT Tail Air, наша пропозиція має низку переваг, особливо у контексті індивідуального

					КС КРБ 123.244.00.00 ПЗ	Арк.
						16
Змн.	Арк.	№ докум.	Підпис	Дата		

виробництва, наукових досліджень або навчального процесу. Перш за все, наша система є відкритою та модульною, що передбачає повну адаптацію до конкретних потреб користувача. Її серце складають недорогі та легкодоступні компоненти: ESP32-CAM для передавання відео через Wi-Fi, Arduino Nano для обробки координат та управління сервоприводом, а також персональний комп'ютер з бібліотекою OpenCV, на який покладено всю логіку розпізнавання об'єктів. Така структура дає змогу легко модифікувати систему, додавати нові функції або інтегрувати її з іншими цифровими чи апаратними пристроями.

Іншою важливою перевагою є низька собівартість виготовлення – вона в десятки разів менша за вартість промислових аналогів, що робить розробку доцільною для самостійного виготовлення, прототипування, лабораторних експериментів та хобі-проектів. Завдяки використанню технологій 3D-друку корпус пристрою можна створити під індивідуальні розміри та специфіку використання, включаючи можливість кріплення на штатив, створення поворотної платформи або розгортання багатокамерної конфігурації. Така гнучкість у фізичному дизайні відкриває нові перспективи щодо розміщення системи в обмежених просторах, на мобільних платформах, у робототехнічних пристроях та подібних рішеннях.

Функціональність також може бути значно розширена. У перспективі можлива інтеграція модулів нічного бачення, електронного масштабування (zoom), мультиоб'єктного трекінгу, а також компонентів для розпізнавання жестів, міміки або навіть емоцій. Крім того, можливо реалізувати голосове керування, взаємодію з користувачем через мобільні додатки або підключення до хмарних сервісів для віддаленого моніторингу. Подібні можливості в OBSBOT Tail Air є закритими або обмеженими на рівні прошивки і, як правило, не підлягають змінам чи кастомізації з боку користувача.

Щодо принципу роботи, в обох випадках камера орієнтована на активне стеження за об'єктом, що рухається. У нашій системі координати об'єкта

					КС КРБ 123.244.00.00 ПЗ	Арк.
						17
Змн.	Арк.	№ докум.	Підпис	Дата		

обчислюються на комп'ютері за допомогою Python-бібліотеки OpenCV, після чого передаються через послідовний порт (UART) на Arduino Nano, який керує сервоприводом, змінюючи положення платформи з камерою. Важливо, що вся система працює в реальному часі, з мінімальною затримкою, що робить її придатною для широкого спектру практичних застосувань: від особистих відео-блогів та запису онлайн-курсів до створення інтерактивних презентацій, наукових досліджень або прототипів у сфері людина-машинна взаємодія. Отже, незважаючи на значно нижчий рівень автоматизації та комерційного впровадження, система, що розробляється, не поступається OBSBOT Tail Air у базовій функції — автоматичному трекінгу — і одночасно має очевидні переваги у гнучкості налаштування, відкритості архітектури, адаптації під індивідуальні вимоги, а також значно нижчій вартості. Це робить її особливо актуальною для навчальних закладів, лабораторій, студентських проєктів, технічних гуртків, а також ентузіастів, які прагнуть вивчити роботу вбудованих систем, мікроконтролерів та алгоритмів комп'ютерного зору.

					КС КРБ 123.244.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		18

РОЗДІЛ 2 ПРОЄКТНА ЧАСТИНА

2.1 Опис логіки роботи комп'ютеризованої системи

Опис взаємодії складових комп'ютеризованої системи через протоколи комунікації має ключове значення для розуміння того, як окремі частини пристрою синхронізують свою роботу в реальному часі. У прикладі система містить три головні пристрої: ESP32-CAM, персональний комп'ютер з програмним забезпеченням на Python (з використанням бібліотеки OpenCV) та мікроконтролер Arduino, до якого підключено сервопривод. Кожен із цих елементів взаємодіє з іншими через власний тип цифрового або аналогового зв'язку, утворюючи цілісну логічну структуру системи (рис.2.1).

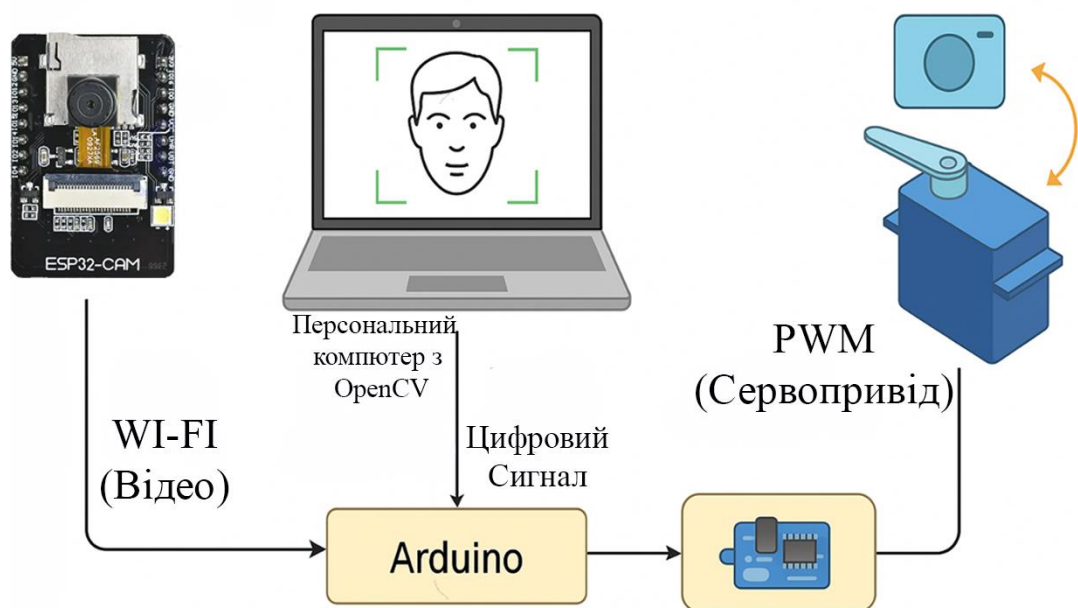


Рисунок 2.1 – Структура комп'ютеризованої системи багатоканального відеозапису динамічних об'єктів

					КС КРБ 123.244.00.00 ПЗ							
Змн.	Арк.	№ докум.	Підпис	Дата								
Розроб.		Чорняк О.В			Проектна частина				Літ.	Арк.	Аркушів	
Перевір.		Тиш Є.В									19	
Реценз.									ТНТУ, каф. КС, гр. СІ-42			
Н. Контр.		Луцик Н.С.										
Затверд.		Осухівська Г.М.										

2.1.1 Передача відеопотоку

На першому кроці, ESP32-CAM виступає джерелом зображення. Цей мікроконтролер з камерою здатний формувати відеопотік в реальному часі. Для його передавання на інший пристрій, ESP32 підключається до локальної бездротової мережі Wi-Fi. Прошивка пристрою запускає HTTP-сервер, який транслює MJPEG-потік за IP-адресою, наприклад, <http://192.168.X.X:81/stream>.

Передача MJPEG-потіку відбувається через TCP-протокол, що забезпечує послідовність і надійність доставки пакетів. Кожне JPEG-зображення надсилається як частина багатокомпонентної HTTP-відповіді з MIME-типом `multipart/x-mixed-replace`. Це дозволяє клієнтському боку (в нашому випадку – персональному комп'ютеру) приймати нові зображення без необхідності перезавантаження з'єднання. Таке рішення є простим, ефективним і не потребує складної програмної архітектури.

На ПК розроблено скрипт мовою Python, що використовує модуль `urllib.request` для підключення до потоку та `OpenCV` для обробки кадрів. Робочий процес базується на безперервному зчитуванні байтового потоку, визначенні початку (`0xFFD8`) та кінця (`0xFFD9`) JPEG-зображення і наступному його декодуванні у формат зображення.

2.1.2 Обробка кадру за допомогою OpenCV та обчислення координат

OpenCV – це бібліотека програмного забезпечення для комп'ютерного зору, що використовується в нашій системі для виявлення облич на відеопотоці, отриманому з камери ESP32-CAM. Вона функціонує на персональному комп'ютері, зчитуючи відеопотік через HTTP-протокол, декодуючи JPEG-кадри в зображення та обробляючи їх за допомогою нейронної мережі для аналізу. Розпізнавання облич ґрунтується на використанні попередньо навченого детектора, заснованого на алгоритмі SSD (Single Shot Multibox Detector) з моделлю Caffe, що дозволяє миттєво визначати координати облич на кожному кадрі. Для підготовки зображення до

					КС КРБ 123.244.00.00 ПЗ	Арк.
						20
Змн.	Арк.	№ докум.	Підпис	Дата		

обробки OpenCV генерує з нього blob – нормалізоване зображення, зменшене до розміру 300x300 пікселів і адаптовано до вимог вхідних даних нейронної мережі. Далі модель виконує прямий прохід (forward pass) зображенням та видає координати прямокутника навколо виявленого обличчя. Ці координати масштабуються до розміру оригінального кадру, після чого визначається центр обличчя та його позиція в полі зору. У разі, якщо обличчя знаходиться поза центральною "мертвою зоною", OpenCV формує команду з певним кутом повороту і відправляє її через UART на Arduino для механічного регулювання позиції камери.

Сама нейронна мережа влаштована таким чином, що розділяє зображення на сітку, де кожна комірка прогнозує прямокутник і ймовірність наявності обличчя. Завдяки цьому обробка здійснюється дуже швидко, дозволяючи виявляти обличчя практично в режимі реального часу, навіть на звичайних комп'ютерах без відеокарт. Ця особливість робить OpenCV ідеальним рішенням для інтеграції в компактні комп'ютерні системи з обмеженими обчислювальними ресурсами.

2.1.3 Передача команди повороту

Після розрахунку координат Python-програма надсилає значення кута (від 0 до 180 градусів) на мікроконтролер Arduino. Передача даних виконується через стандартний UART (універсальний асинхронний прийомопередавач) – найпоширеніший протокол для простого двонаправленого серійного зв'язку. У нашому випадку, оскільки Arduino підключено до ПК через USB, сигнал UART реалізується через віртуальний COM-порт.

На Arduino очікується текстовий рядок, який зчитується функцією Serial.readStringUntil('\n'), перетворюється на число та використовується для керування сервоприводом.

Завдяки простому текстовому протоколу передача є надійною та легко налагоджується. Також такий підхід дозволяє розширювати функціональність,

					КС КРБ 123.244.00.00 ПЗ	Арк.
						21
Змн.	Арк.	№ докум.	Підпис	Дата		

додаючи інші команди в майбутньому.

2.2 Обґрунтування вибору апаратного забезпечення компютеризованої системи

Вибір компонентів обладнання для комп'ютерної системи багатокамерного відеофіксування рухомих об'єктів обумовлений потребою забезпечення її злагодженої роботи в реальному часі, зважаючи на обмеженість доступних ресурсів. Проект розроблено з урахуванням потреб у компактності, незалежності від зовнішнього живлення, економному споживанні енергії та низькій вартості складових — що є критичним для розбудови багатокамерних систем, де ці характеристики множаться.

2.2.1 Обґрунтування вибору відеокамери

Ключовим компонентом системи є мікроконтролер ESP32-CAM. Він являє собою сукупність Wi-Fi модуля, процесора з архітектурою Xtensa LX6 та вмонтованої камери. Вибір саме цього модуля пояснюється декількома факторами.

ESP32-CAM (рис.2.2), має у своєму розпорядженні обчислювальні ресурси, необхідні для захоплення та первинної обробки зображень високої якості, з роздільною здатністю до UXGA (1600×1200 пікселів). Це забезпечує фіксацію всіх ключових елементів у кадрі, що є надзвичайно важливим при вирішенні задач комп'ютерного зору, зокрема, розпізнаванні облич. Модульна підтримка форматів JPEG та RGB дає змогу оптимізувати об'єм даних, які передаються, без суттєвого погіршення якості, тим самим прискорюючи аналіз відеопотоку. Також, інтегрований Wi-Fi модуль забезпечує можливість бездротової передачі відеопотоку безпосередньо на персональний комп'ютер або сервер для подальшої обробки. Це дозволяє значно спростити фізичне розташування камери у просторі — вона не прив'язана до жорсткого з'єднання через USB або інші провідні інтерфейси. Завдяки цьому система легко

					КС КРБ 123.244.00.00 ПЗ	Арк.
						22
Змн.	Арк.	№ докум.	Підпис	Дата		

масштабована: до неї можна підключити декілька ESP32-CAM, які працюватимуть одночасно в одній Wi-Fi мережі.



Рисунок 2.2 – ESP32 cam

Аналізуючи можливі варіанти для втілення системи комп'ютерного зору на базі вбудованих пристроїв, доречно розглянути камеру OV7670 як одну з розповсюджених альтернатив (рис.2.3). Цей модуль часто використовується в простих проєктах з Arduino, що пояснюється його невисокою вартістю та легкістю доступу. Але, попри його популярність у навчальних проєктах, він має значні функціональні обмеження, особливо якщо враховувати сучасні вимоги до відеозйомки, розпізнавання облич та передачі даних.



Рисунок 2.3 – OV7670

OV7670 використовує лише інтерфейс SCCB (аналог I²C), а для передачі відео потребує прямого підключення до мікроконтролера з достатньо великою кількістю GPIO. Проблема полягає в тому, що більшість плат, таких як Arduino Uno або Nano, не володіють потрібною обчислювальною потужністю або достатнім об'ємом оперативної пам'яті для обробки навіть низькорівневої відеоінформації. Зберігання цілого кадру в оперативній пам'яті цих пристроїв є фізично неможливим. Крім того, OV7670 не оснащена вбудованим JPEG-енкодером, тому обробку всіх даних необхідно виконувати в зовнішньому середовищі. На противагу цьому, ESP32-CAM має вбудований JPEG-кодер, PSRAM для буферизації відео та Wi-Fi-модуль, що дозволяє передавати відеопотік напряму на ПК або сервер, уникаючи потреби в додаткових хост-контролерах. Це дозволяє реалізувати повноцінний стримінг з мінімальними затримками, чого неможливо досягти з OV7670 без застосування складних обхідних рішень.

					КС КРБ 123.244.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		24

Таблиця 2.1 – Порівняння характеристик ESP32 cam та OV7670

Назва	ESP32	OV7670
Роздільна здатність	До 1600×1200 (JPEG)	До 640×480 (VGA, без JPEG)
Формат зображення	JPEG (апаратне кодування)	RAW RGB565/YUV (без стиснення)
Підключення до Wi-Fi	Вбудований Wi-Fi	Вимагає додатковий модуль
Підтримка потокового відео	Так (HTTP, MJPEG)	Ні (вимагає зовнішній буфер)
Застосування для OpenCV	Пряма інтеграція через Wi-Fi	Потрібен проміжний пристрій

Отже, ESP32-CAM представляє собою найкращий варіант для розробки автономних та доступних рішень у сфері комп'ютерного зору. Вона демонструє ефективний баланс між функціональністю, вартістю та простотою впровадження, суттєво переважаючи такі модулі, як OV7670, у реальних умовах експлуатації.

2.2.2 Обґрунтування вибору сервоприводу

Сервопривід Tower Pro MG996R (рис.2.4) один з найпопулярніших та надійних у своєму сегменті. З аналоговим сервомеханізмом та великим крутним моментом, що дозволяє переміщати об'єкти значної ваги з високою точністю.



Рисунок 2.4 – Tower Pro MG996R

Конструкція MG996R передбачає міцний пластиковий корпус і редуктор, виготовлений з металевих шестерень. Це забезпечує підвищену стійкість до зношування та дозволяє ефективно працювати під великими навантаженнями протягом тривалого часу. В порівнянні з більш простими моделями, MG996R може обертатися до 180° , також має двомісний підшипник, що гарантує плавність і стабільність обертання. Ще однією важливою особливістю є підтримка роботи в широкому діапазоні напруги (від 4.8В до 7.2В), що розширює можливості при проектуванні системи живлення. Крутний момент сягає 9.4 кг·см при 6В, що дозволяє використовувати його з камерою, закріпленою на важкій конструкції, без ризику втрати потужності або пошкодження.

Окремо слід відзначити наявність захисту від перегріву та перевантаження у MG996R, що значно знижує ймовірність поломки при тривалій або циклічній роботі. Цей фактор робить його оптимальним рішенням для систем з численними циклами відстеження, таких, де пристрій безперервно реагує на координати обличчя та коригує положення камери відповідно до руху об'єкта.

У рамках проєкту, цей сервопривід відповідає за поворот камери (або смартфона) згідно з координатами обличчя, отриманими в результаті обробки

					КС КРБ 123.244.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		26

відеопотоку. Варто також розглянути сервопривід TS90D (рис.2.5) компактний аналоговий сервомеханізм.



Рисунок 2.5 – TS90D

Хоча його редуктор виготовлено з пластику, що робить його менш стійким до зносу, цей недолік компенсується легкою вагою, швидкою реакцією та простотою інтеграції. TS90D підтримує стандартний кут обертання 0–180 градусів, що дозволяє застосовувати його для поворотів та позиціонування легких компонентів. Крутний момент TS90D становить приблизно 2.5 кг·см при 6В, чого достатньо для точного повороту об'єктів з невеликою інерцією, таких як датчики, лазери, невеликі камери або декоративні елементи. Низьке енергоспоживання робить цей сервопривід придатним для роботи з обмеженими джерелами живлення (наприклад, від акумулятора чи USB).

Втім, через пластикові шестерні, відсутність захисту від перевантажень та низький крутний момент, TS90D не рекомендується для систем, де привід повинен працювати під значним навантаженням або в умовах інтенсивного циклічного навантаження – як у випадку нашої автоматизованої поворотної камери. Його використання може призвести до передчасного зносу або зниження точності через люфти в редукторі.

Підсумовуючи, TS90D є добрим вибором для легких задач, але не

					КС КРБ 123.244.00.00 ПЗ	Арк.
						27
Змн.	Арк.	№ докум.	Підпис	Дата		

забезпечує необхідного рівня стабільності, надійності та сили для нашої системи з сервокерованим поворотом відеокамери на основі даних комп'ютерного зору. Для прийняття обґрунтованого рішення, зіставимо MG996R із широко відомим аналогом TS90D.

Таблиця 2.2 – Порівняння характеристик MG996R cam та TS90D

Назва	Tower Pro MG996R	TS90D
Тип шестерень	Металеві	Пластикові
Макс. Крутний момент	9.4 кг·см	9.4 кг·см
Швидкість обертання	0.17 сек/60°	0.12 сек/60°
Тип управління	Аналогове	Аналогове
Захист від перегріву/навантаження	Є	Відсутній

2.2.3 Опис розробленого штативу

У конструкції комп'ютеризованої системи відеоспостереження використано індивідуально розроблений штатив, що враховує особливості експлуатації ESP32-CAM та смартфона як знімальних пристроїв. Штатив спроектовано та виготовлено з використанням 3D-друку, що забезпечило максимальну гнучкість у розробці та адаптації до конкретних технічних вимог. Основне завдання штатива – надійна фіксація пристроїв із можливістю динамічного обертання в горизонтальній та вертикальній площинах завдяки встановленим сервоприводам (рис. 2.6).

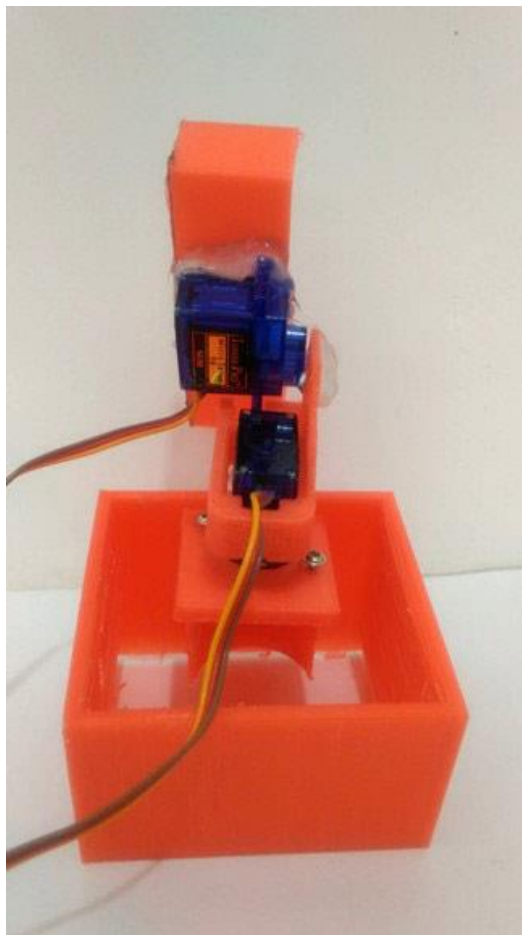


Рисунок 2.6 – Штатив розроблений для компютеризованої системи

Конструкція складається з базової платформи, двох поворотних вузлів і спеціального утримувача для знімального модуля. Нижній сервопривід відповідає за обертання по горизонталі, а верхній регулює кут нахилу по вертикалі. Всі компоненти з'єднані болтами та фіксуючим клеєм, що забезпечує необхідну міцність та, водночас, полегшує обслуговування пристрою. Застосування 3D-друку дозволило не тільки зменшити вартість виробництва корпусу, а й оптимізувати його форму, враховуючи розміри електронних компонентів, кабельних з'єднань та центру мас. Крім того, така технологія дає змогу швидко вносити зміни в конструкцію при необхідності, наприклад, адаптувати кріплення під інші камери або смартфони. Розміри штатива дозволяють легко розміщувати його на робочому столі або фіксувати на нерухомих об'єктах, що робить систему зручною для стаціонарного та

					КС КРБ 123.244.00.00 ПЗ	Арк.
						29
Змн.	Арк.	№ докум.	Підпис	Дата		

мобільного використання.

Надрукований на 3D-принтері штатив не тільки виконує функцію механічної підтримки, але й виступає інтегруючим елементом – саме він забезпечує точне спрямування камери відповідно до команд, отриманих від контролера. Це апаратне рішення вдало поєднує в собі доступність, функціональність та високий рівень адаптивності.

2.2.4 Структурна схема роботи комп'ютеризованої системи

Система реалізована у вигляді послідовного ланцюга взаємодіючих модулів, кожен з котрих виконує чітко визначену роль у процесі виявлення та супроводу об'єкта (обличчя) в кадрі (рис.2.7).

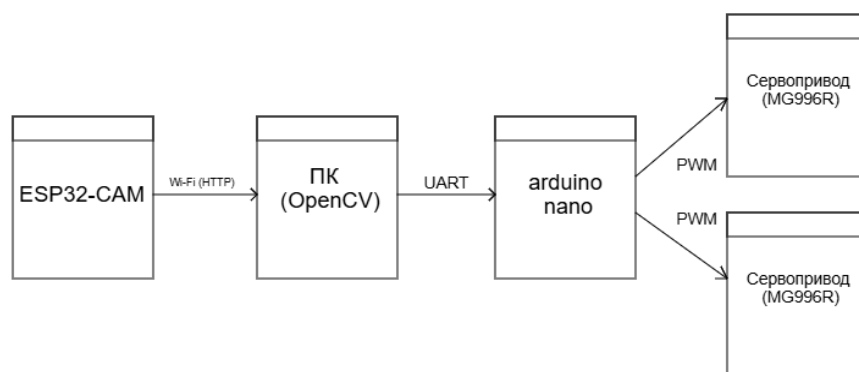


Рисунок 2.7 – Структурна схема

Першим елементом є ESP32-CAM, який виконує функцію візуального сенсора. Він оснащений камерою і модулем Wi-Fi, що дозволяє здійснювати безпроводну трансляцію відеопотоку на персональний комп'ютер за протоколом HTTP. Важливо підкреслити, що ESP32-CAM не виконує обробку зображення локально, а лише передає «сире» відео на зовнішній обчислювальний вузол, тим самим значно знижуючи навантаження на вбудований мікроконтролер.

Далі у ланцюжку знаходиться персональний комп'ютер, на котрому

					КС КРБ 123.244.00.00 ПЗ	Арк.
						30
Змн.	Арк.	№ докум.	Підпис	Дата		

реалізовано програмне забезпечення на базі бібліотеки OpenCV. Саме тут відбувається основна інтелектуальна обробка відео: детекція обличчя за допомогою алгоритмів машинного зору, обчислення координат обличчя у кадрі та визначення, наскільки воно відхилилося від центральної осі. Персональний комп'ютер виконує роль аналітичного ядра системи, приймаючи рішення про потребу корекції положення зйомного пристрою (смартфона, прикріпленого до сервоосі).

Результатом обробки є числове значення кута повороту, яке передається по дротовому інтерфейсу UART до Arduino Nano — недорогого та енергоефективного мікроконтролера. Arduino не займається обробкою зображень, а лише виконує команди, отримані з ПК. Це розділення функціональності дозволяє знизити навантаження на кожний модуль і збільшити загальну швидкодію системи. Вихідні сигнали з Arduino надходять на два сервоприводи MG996R, котрі керуються через PWM-сигнали (широтно-імпульсна модуляція). Завдяки високому крутному моменту ці приводи можуть обертати тримач смартфона, забезпечуючи горизонтальне або вертикальне позиціонування. У такій конфігурації можлива побудова двовісної платформи, що дозволяє системі адаптуватись до положення обличчя в просторі.

Таким чином, структурна схема відображає чітку розподілену архітектуру, в якій кожен модуль виконує спеціалізовану функцію — сенсорну, обчислювальну або виконавчу — і взаємодіє з іншими за допомогою уніфікованих інтерфейсів. Така модульність дозволяє легко замінювати компоненти, масштабувати систему або адаптувати її під інші задачі, наприклад — жестове керування або трекінг об'єктів у промисловості.

					КС КРБ 123.244.00.00 ПЗ	Арк.
						31
Змн.	Арк.	№ докум.	Підпис	Дата		

2.3 Обґрунтування вибору програмного забезпечення комп'ютеризованої системи

Вибір програмних засобів для втілення комп'ютеризованої системи відеонагляду базувався на практичності, ефективності та доступності інструментів, які здатні забезпечити як програмування апаратної складової, так і обробку даних, управління сервоприводом та створення корпусу пристрою. Усі програмні рішення були підібрані з урахуванням особливостей задачі: взаємодія вбудованих мікроконтролерів, комп'ютерний зір, передача даних у режимі реального часу та фізичне компонування пристрою.

Для прошивки модуля ESP32-CAM було обрано середовище розробки Arduino IDE. Воно надає зручний інтерфейс для програмування мікроконтролерів та має повну підтримку плати ESP32. Завдяки відкритості коду, Arduino IDE широко застосовується в галузі IoT та робототехніки, що гарантує наявність актуальних бібліотек та активної спільноти, здатної оперативно надавати допомогу у випадку труднощів. Зокрема, використання бібліотеки `esp_camera.h` дозволяє швидко реалізувати функції ініціалізації камери, підключення до Wi-Fi, обробки зображень та керування виводами ESP32.

Важливу роль для реалізації інтелектуальної частини проєкту відіграла мова програмування Python. Завдяки бібліотеці OpenCV, що містить широкий набір інструментів для обробки зображень, стало можливим здійснювати захоплення відеопотоку, виділення облич на кадрах за допомогою нейронної мережі, визначення координат об'єкта та передачу їх через серійний порт на Arduino. OpenCV підтримує різні попередньо натреновані моделі для детекції облич, зокрема, мережу `res10_300x300_ssd`, яка у нашому випадку забезпечила достатню точність та швидкодію. Бібліотека також дає змогу візуалізувати процес у реальному часі, що було корисним на етапі відлагодження алгоритмів.

					КС КРБ 123.244.00.00 ПЗ	Арк.
						32
Змн.	Арк.	№ докум.	Підпис	Дата		

Обмін даними між ПК та мікроконтролером реалізовано через інтерфейс UART, використовуючи бібліотеку pyserial. Такий підхід є простим, стабільним та добре підтримується апаратними платформами. Він дозволяє без затримок передавати значення кута обертання сервоприводу, що забезпечує безперервне стеження за об'єктом без помітної інерції.

Для керування механічним приводом, який обертає модуль камери, було обрано Arduino Nano. Це один з найнадійніших та найпопулярніших мікроконтролерів з великою базою прикладів і бібліотек. Зокрема, бібліотека Servo.h дає змогу здійснювати точне позиціонування сервоприводу за заданим кутом, що критично важливо для адекватної реакції на зміну координат об'єкта у відеопотоці. Arduino, завдяки простоті програмування та низькому енергоспоживанню, ідеально підходить для керування подібними механізмами.

Ще одним важливим етапом розробки проєкту стало створення фізичного корпусу пристрою. Для цього використовувалась система автоматизованого проєктування (CAD) Autodesk Fusion 360. Цей інструмент був обраний через його здатність поєднувати середовище тривимірного моделювання, інженерного аналізу та підготовки до 3D-друку в одному програмному продукті. З його допомогою було змодельовано штатив, який забезпечує надійне кріплення модуля ESP32-CAM та сервоприводу, а також дозволяє зручно позиціонувати пристрій у просторі. Fusion 360 підтримує експорт у формати STL, які є стандартом для 3D-друку, завдяки чому розроблена конструкція без проблем була реалізована за допомогою FDM-принтера.

Таким чином, вибір програмного забезпечення є обґрунтованим з точки зору функціональної повноти, доступності, підтримки спільноти та гнучкості для подальшого удосконалення системи. Arduino IDE, Python з OpenCV, а також Fusion 360 у комплексі забезпечили можливість повністю реалізувати комп'ютеризовану систему, яка включає обробку відео, автоматичне керування камерою та надійну конструкцію.

					КС КРБ 123.244.00.00 ПЗ	Арк.
						33
Змн.	Арк.	№ докум.	Підпис	Дата		

2.3.1 Схема алгоритму роботи компютеризованої системи

Алгоритмічна схема описує послідовність операцій, які виконуються всередині кожного з функціональних модулів системи. Вона умовно розділена на три головні блоки, що відповідають модулям: ESP32-CAM, ПК (OpenCV) та Arduino Nano. Усі процеси синхронізовані та функціонують в реальному часі, забезпечуючи безперервну реакцію на зміну положення обличчя в кадрі (див.рис.2.8-2.10).

Алгоритм для
ESP32 cam



Рисунок 2.8 – Схема алгоритму роботи коду для ESP32Cam

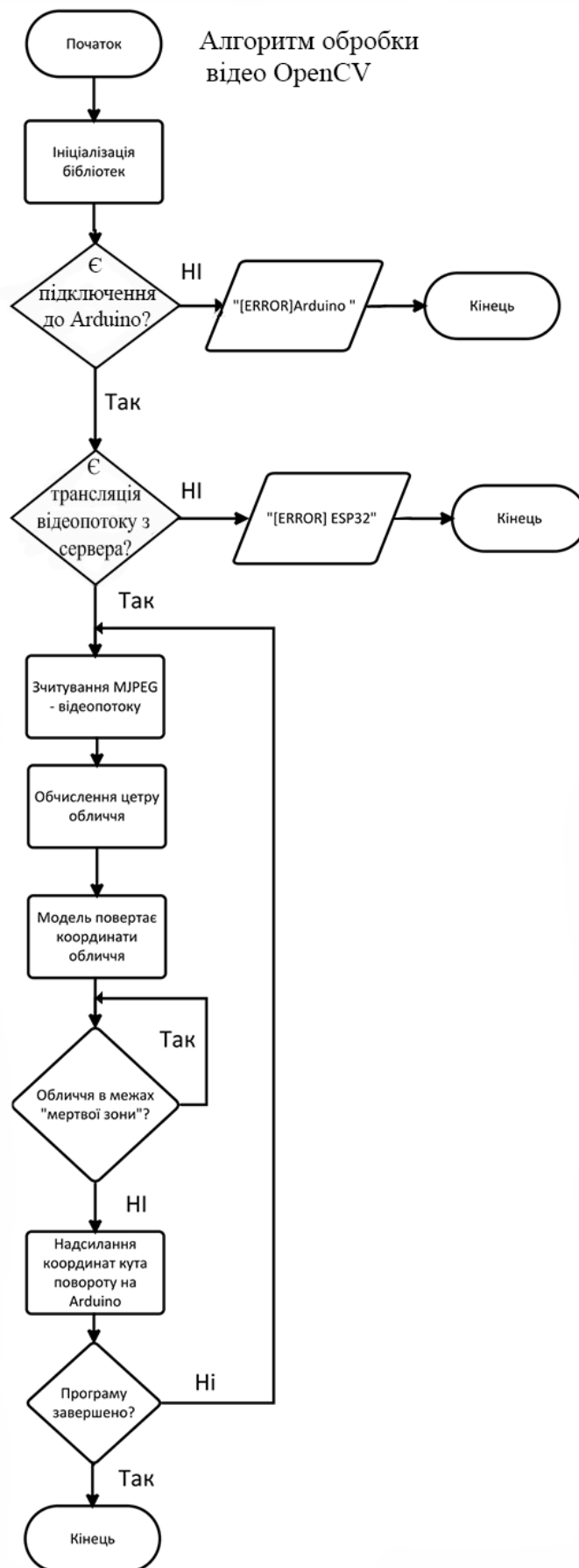


Рисунок 2.9 – Схема алгоритму роботи коду OpenCV



Рисунок 2.10 – Схема алгоритму керування сервоприводів

На початку ініціалізуються пінові налаштування камери та конфігурація Wi-Fi-з'єднання. При вдалому підключенні до мережі виконується налаштування параметрів зображення (роздільна здатність, формат, якість) і розпочинається трансляція MJPEG-потoku на вказану IP-адресу. Цей потік постійно доступний для зчитування зовнішніми клієнтами, зокрема персональним комп'ютером. Якщо з'єднання з мережею відсутнє, процес припиняється.

Центральним етапом є ініціалізація бібліотек, зокрема OpenCV, та перевірка наявності доступу до Arduino і ESP32. Якщо хоча б один з пристроїв недоступний, програма завершить роботу з відповідною діагностикою.

У разі успішного старту програма переходить у режим обробки потоку. З кожного зчитаного кадру виконується аналіз за допомогою DNN-моделі

обличчя (детектор глибокого навчання). Виявлене обличчя обробляється для визначення його центру по горизонтальній осі. На основі положення центру обличчя визначається, чи перебуває воно в межах так званої «мертвої зони» — центральної області кадру, в якій серво залишає положення незмінним. Якщо обличчя за її межами — виконується обчислення нового кута повороту і передача його до Arduino через UART.

Arduino працює в циклічному режимі, чекаючи на серійну команду. При отриманні числового значення кута, він перетворює його у PWM-сигнал відповідної тривалості та подає на сервопривід. Завдяки цьому обертання відбувається з мінімальною затримкою і плавно.

Уся система працює до моменту завершення програми користувачем або до виникнення помилки з'єднання. Такий підхід дозволяє забезпечити стабільність і надійність.

					КС КРБ 123.244.00.00 ПЗ	Арк.
						37
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 3 ПРАКТИЧНА ЧАСТИНА

3.1 Реалізація практичних рішень

3.1.1 Налаштування середовища розробки для програмування ESP32cam

Програмування модуля ESP32-CAM відбувається у середовищі Arduino IDE, яка встановлюється з офіційного веб-сайту (рис. 3.1).

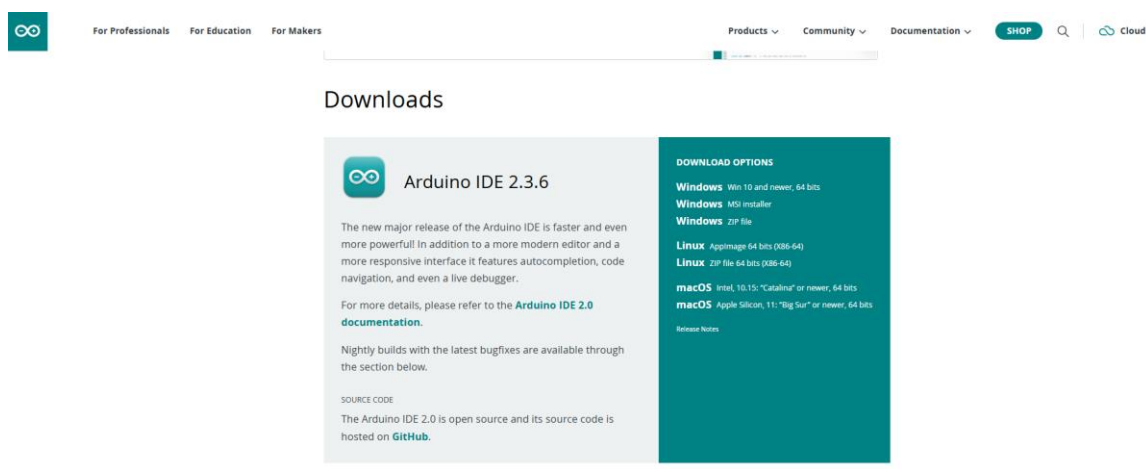


Рисунок 3.1 – Завантаження середовища розробки Arduino IDE

У конфігурації IDE, в полі "Додаткові URL для Менеджера плат" вставляється посилання на пакет підтримки ESP32 (див. рис. 3.2).

- Відкрити: Файл → Параметри;
- У полі «Додаткові URL-адреси для менеджера плат» додати “ https://dl.espressif.com/dl/package_esp32_index.json ”.

					КС КРБ 123.244.00.00 ПЗ		
Змн.	Арк.	№ докум.	Підпис	Дата	Практична частина		
Розроб.		Чорняк О.В					
Перевір.		Тиш Є.В					
Реценз.							
Н. Контр.		Луцик Н.С.					
Затверд.		Осухівська Г.М.					
					Лім.	Арк.	Аркушів
						38	
					ТНТУ, каф. КС, гр. СІ-42		

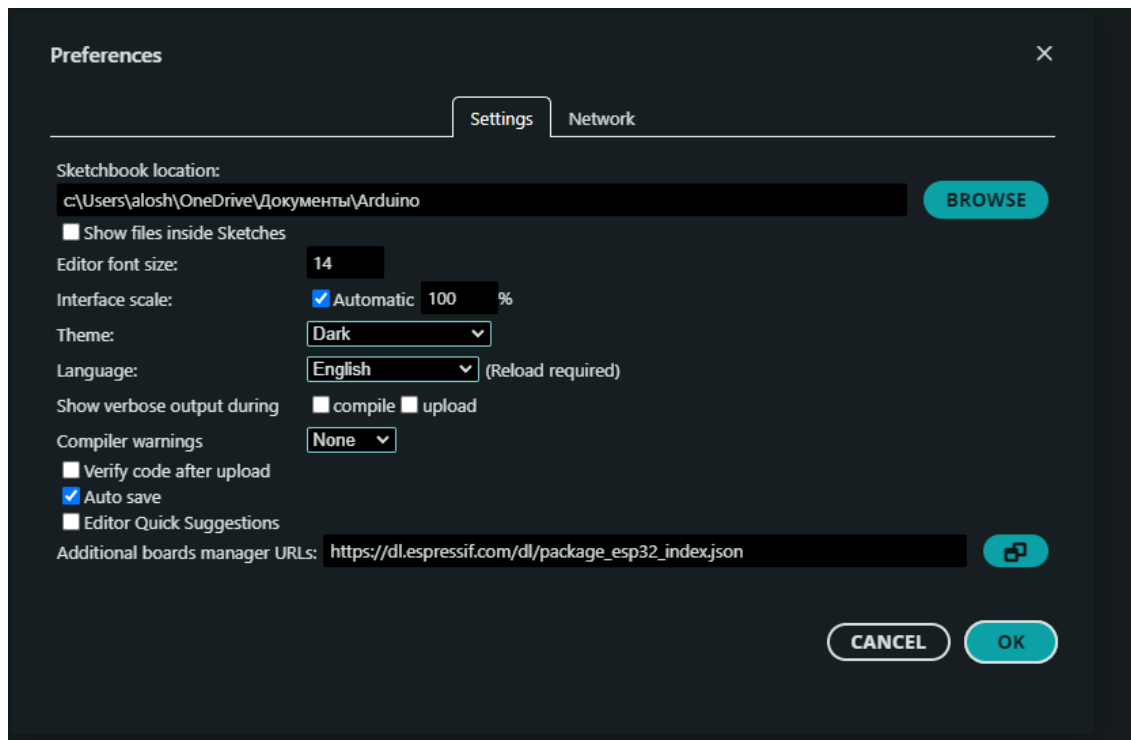


Рисунок 3.2 – Додаткові URL – адресі плат

Далі, в меню "Керування платами" відбувається встановлення офіційної бібліотеки.

Інструменти → Плати → Керування платами встановлено пакет ESP32 by Espressif Systems (рис3.3).

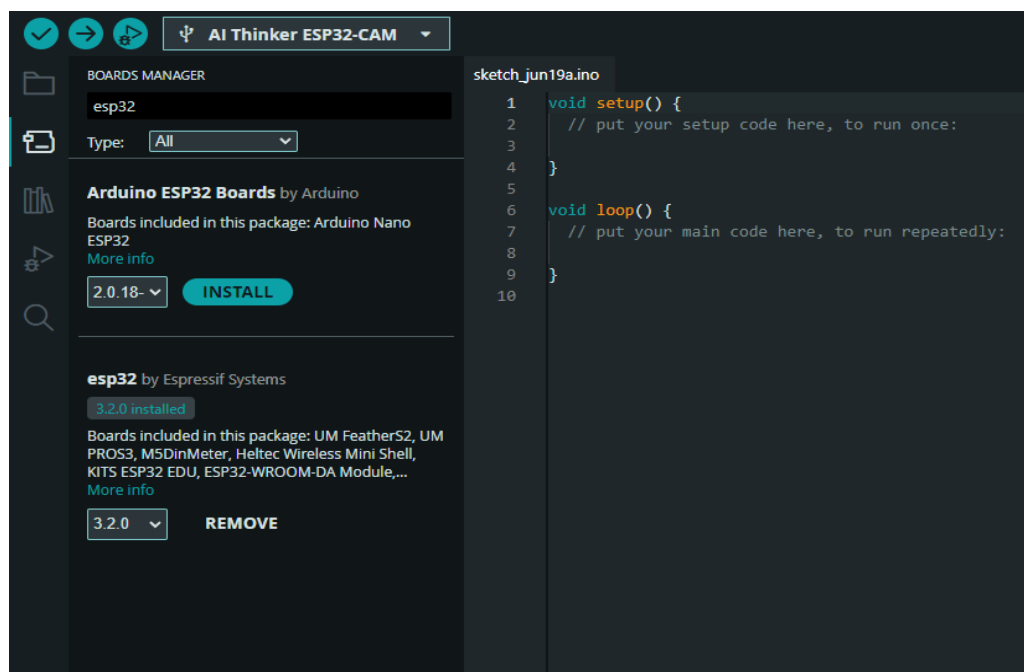


Рисунок 3.3 – Завантаження пакету ESP32 by Espressif Systems

У параметрах обирається плата AI Thinker ESP32-CAM. Ця плата вирізняється наявністю вбудованої камери та PSRAM, що забезпечує можливість передавати відео з високою чіткістю. Додатково встановлюється тактова частота на 240 MHz, обирається потрібний COM-порт, а також швидкість завантаження даних 115200 бод (див. рис. 3.4).

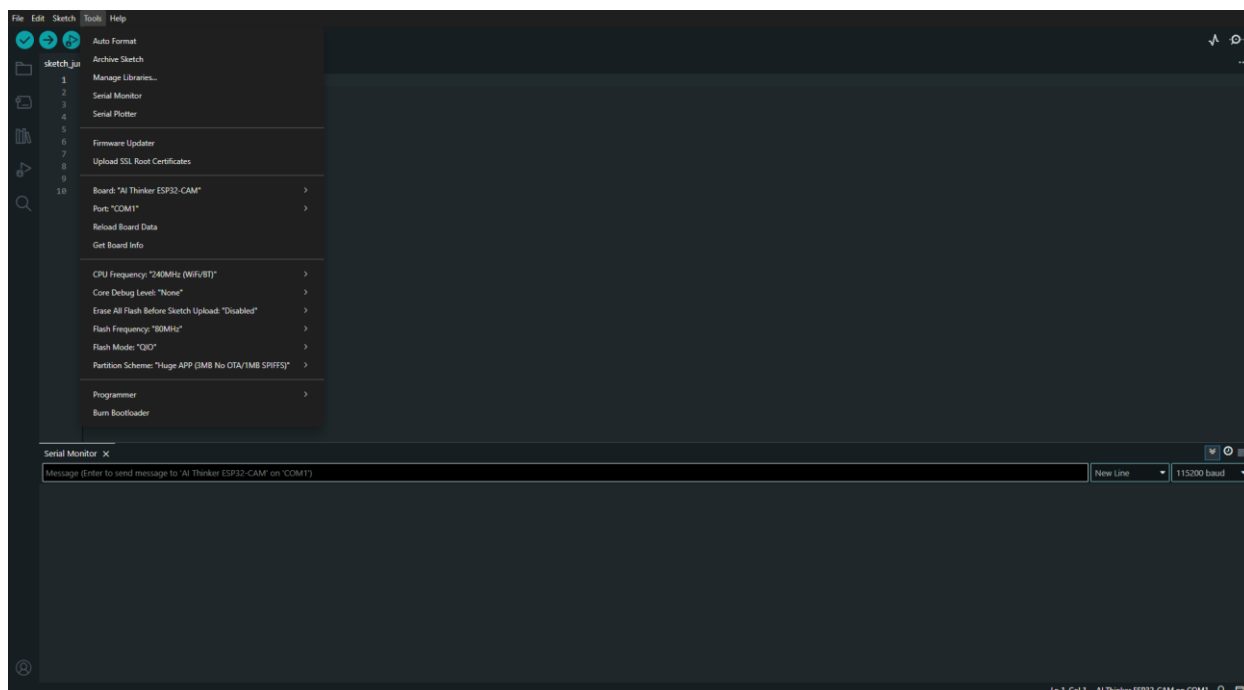


Рисунок 3.4 – Завершення налаштувань середовища розробки

Перед прошиванням було замкнено пін IO0 на GND для переходу в режим завантаження. Після натискання кнопки Upload в Arduino IDE відбулося компілювання та завантаження прошивки. Після прошивання було розімкнено IO0–GND та натиснуто Reset, щоб ESP32-CAM почала функціонувати у звичайному режимі лістинг коду програми (див.Додаток Б).

					КС КРБ 123.244.00.00 ПЗ	Арк.
						40
Змн.	Арк.	№ докум.	Підпис	Дата		

3.1.2 Налаштування середовища розробки для роботи з OpenCV

Система обробки зображень реалізована на Python 3.10+. Python потрібно встановити з офіційного сайту.

More resources • Online Documentation • PEP 719, 3.13 Release Schedule • Report bugs at https://github.com/python/cpython/issues. • Help fund Python directly (or via GitHub Sponsors), and support the Python community. Full Changelog							
Files							
Version	Operating System	Description	MD5 Sum	File Size	GPG	Signature	SBOM
Gzipped source tarball	Source release		88dc0b8317cab6e46e8336995bcc577f	28.1 MB	SIG	signature	SPDX
XZ compressed source tarball	Source release		dbaa833aa736eddb18a6a6e0c10fa	21.8 MB	SIG	signature	SPDX
macOS 64-bit universal2 installer	macOS	for macOS 10.13 and later	a0f126b757effdbf7e1d0863a6f176f	67.1 MB	SIG	signature	
Windows installer (64-bit)	Windows	Recommended	da9f24ae94e5b3491f3d92b07d34cc72	27.5 MB	SIG	signature	SPDX
Windows installer (32-bit)	Windows		d3898d8ea3a1524b043458311446c0b3	26.2 MB	SIG	signature	SPDX
Windows installer (ARM64)	Windows	Experimental	75b8a99cfd9fd5c15771b598bc067385	26.8 MB	SIG	signature	SPDX
Windows embeddable package (64-bit)	Windows		370a345d8ea8bbc1830a2385f24632d2	10.4 MB	SIG	signature	SPDX
Windows embeddable package (32-bit)	Windows		589d9d4938b4e19c9fa4de83aad3d425	9.2 MB	SIG	signature	SPDX
Windows embeddable package (ARM64)	Windows		0d2b5391a1df1319242f17a8339b8bc6	9.7 MB	SIG	signature	SPDX
About	Downloads	Documentation	Community	Success Stories	News		
Applications	All releases	Docs	Diversity	Arts	Python News		
Quotes	Source code	Audio/Visual Talks	Mailing Lists	Business	PSF Newsletter		
Getting Started	Windows	Beginner's Guide	IRC	Education	PSF News		
Help	macOS	Developer's Guide	Forums	Engineering	PyCon US News		

Рисунок 3.6 – Завантаження Python 3.15.5

Тепер потрібно встановити бібліотеки для роботи з OpenCV через pip, вони забезпечують обробку відео, роботу з масивами та зв'язок з Arduino.

```
pip install opencv-python
pip install numpy
pip install pyserial
```

На відміну від традиційних каскадів Хаара, у цьому проєкті використовується нейромережевий детектор обличчя на базі SSD (Single Shot Detector), що гарантує вищу точність та надійність. Для реалізації цього застосовуються два DNN-файли:

- deploy.prototxt — архітектура моделі;
- res10_300x300_ssd_iter_140000.caffemodel — ваги алгоритму.

Ці файли були поміщені в директорію з Python-скриптом (рис.3.7).

					КС КРБ 123.244.00.00 ПЗ	Арк.
						41
Змн.	Арк.	№ докум.	Підпис	Дата		

Ім'я	Стан	Дата змінення	Тип	Розмір
face_tracker	↻	11.06.2025 22:40	Папка файлів	
sketch_jun4b	↻	04.06.2025 20:02	Папка файлів	
detect_faces.py	↻	16.06.2025 22:37	Исходный файл ...	3 КБ
face_tracker.py	↻	16.06.2025 23:56	Исходный файл ...	4 КБ
haarcascade_frontalface_default.xml	↻	19.05.2025 23:23	Microsoft Edge Н...	909 КБ
res10_300x300_ssd_iter_140000.caffemodel	↻	16.06.2025 22:38	Файл CAFFEMODEL	10 417 КБ

Рисунок 3.7 – Директорія комп'ютерної системи

Python-скрипт зчитує MJPEG-потік з ESP32-CAM, виділяє JPEG-кадри, подає їх на обробку в нейронну мережу та обчислює координати обличчя у кадрі. Якщо об'єкт виявлено поза центральною зоною, визначається, у який бік слід повернути камеру, після чого обчислюється новий кут і надсилається на Arduino через COM-порт лістинг (див.Додаток Б).

3.1.3 Налаштування середовища розробки для Arduino Nano

В Arduino IDE у параметрах обирається плата Arduino Nano та COM port до якого підключений мікроконтролер (рис.3.9)

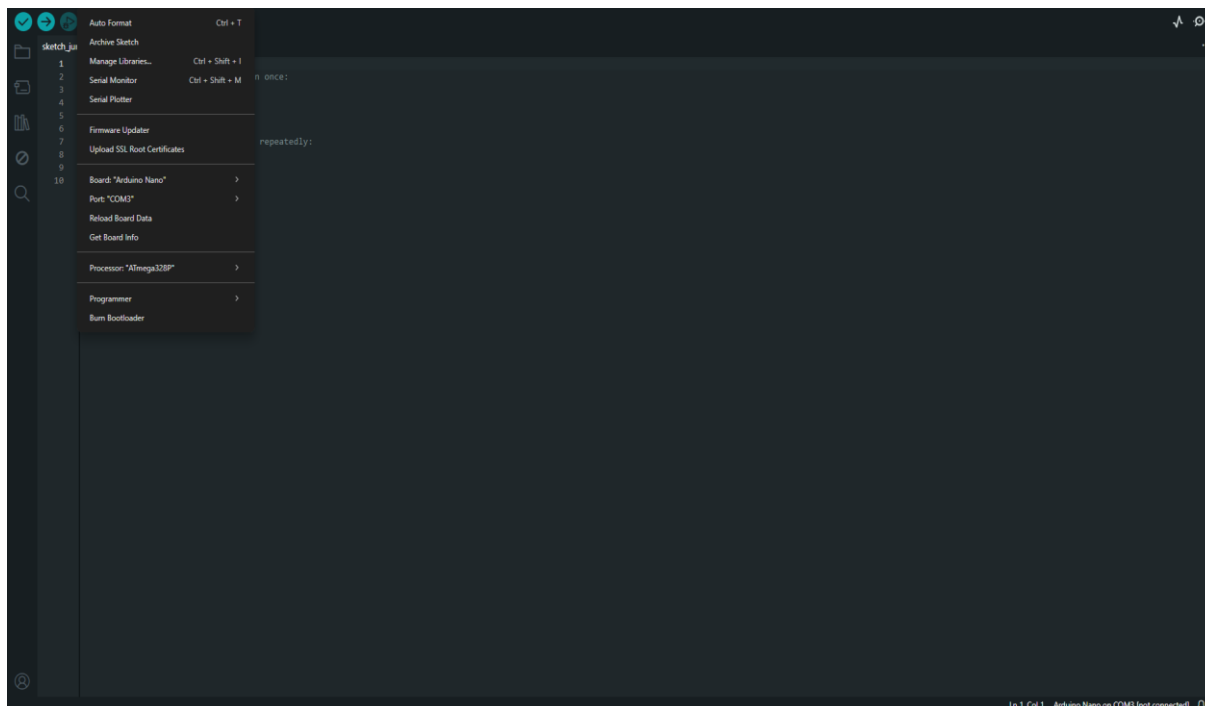


Рисунок 3.9 – Завершення налаштувань середовища розробки

					КС КРБ 123.244.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		42

Для створення керуючого сигналу використовується бібліотека Servo.h. Після отримання значення Arduino обмежує його діапазоном від 0 до 180° і надсилає відповідний PWM-сигнал на контакт, до якого приєднано сервомотор MG996R. Сервопривід обертає платформу зі смартфоном або камерою, забезпечуючи фізичне відстеження об'єкта в кадрі.

Уся логіка Arduino є простою та ефективною: обробка послідовного вводу, обчислення кута, передавання сигналу — усе функціонує з мінімальною затримкою, що дозволяє камері вчасно реагувати на зміни у положенні обличчя лістинг (рис.3.10).

```
#include <Servo.h>    // Підключення бібліотеки для роботи з
сервоприводом

Servo myservo;        // Створення об'єкту сервоприводу
int angle = 90;        // Початковий кут сервоприводу (90° – центральне
положення)

void setup() {
    Serial.begin(9600);    // Ініціалізація послідовного порту
    myservo.attach(9);    // Підключення сервоприводу до пін D9
    myservo.write(angle);    // Встановлення початкового положення
}

void loop() { // Перевірка наявності даних у послідовному порту
    if (Serial.available()) {
        String input = Serial.readStringUntil('\n'); // Читання рядка
з кутовим значенням

        int new_angle = input.toInt();    // Перетворення рядка в число
        // Обмеження кута в межах 0-180° (захист сервоприводу)
        new_angle = constrain(new_angle, 0, 180);

        myservo.write(new_angle);    // Встановлення нового кута
        delay(15);    // Невелика затримка для стабілізації
    }
}
```

Рисунок 3.10 – Лістинг системи повороту сервоприводу

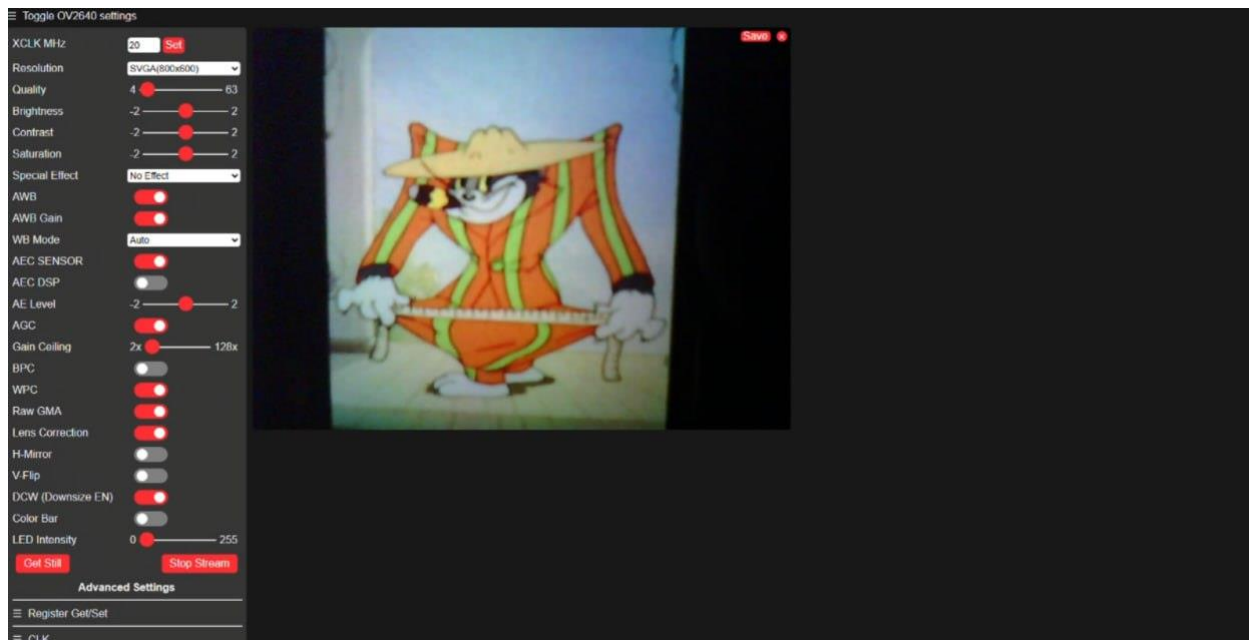


Рисунок 3.12 – Трансляція відопотоку на веб сервер

Аби переконатися в правильному функціонуванні Python-скрипту, що відповідає за ключову обробку відеоряду в реальному часі, потрібен його запуск. Після введення команди запуску у терміналі, система розпочинає відображати дані про свою діяльність: наприклад, поточні кути нахилу для сервоприводів, що надсилаються, а також інформацію про виявлення обличчя у кожному кадрі. За умови коректного налаштування всіх складників системи, Python-оточення забезпечує безперебійний обмін даними з мікроконтролерами, такими як ESP32 та Arduino Nano, використовуючи послідовний порт або інші протоколи зв'язку, вказані в програмному коді. Про це свідчить виведення в консоль переданих та отриманих значень, таких як підтвердження отримання даних, сигнали зворотного зв'язку або реакція на зміни в кадрі. Консоль слугує центральним пунктом для відстеження роботи системи дозволяє оперативно виявляти проблеми, збої у зв'язку або некоректну роботу алгоритмів. При безперебійній роботі всі повідомлення відображаються без затримок або попереджень про помилки, підтверджуючи успішну взаємодію програмної та апаратної складових. (рис. 3.13).

```

1 import cv2
2 from urllib.request import urlopen
3 import numpy as np
4 import serial
5 import time
6
7 # === 1. Параметри ===
8 ESP32_STREAM_URL = 'http://192.168.31.103:81/stream' # Заміни на свою IP-адресу
9 SERIAL_PORT = 'COM4' # Заміни на свій COM порт
10 BAUD_RATE = 9600
11 DEAD_ZONE_PERCENT = 0.15
12 current_angle = 90
13
14 # === 2. Ініціалізація DNN ===
15 net = cv2.dnn.readNetFromCaffe("deploy.prototxt", "res10_300x300_ssd_iter_140000.caffemodel")
16
17 # === 3. Підключення до Arduino ===
18 try:
19     arduino = serial.Serial(SERIAL_PORT, BAUD_RATE)
20     time.sleep(2)
21     print("[INFO] Arduino ready.")
22     arduino.write(f"{current_angle}\n".encode())
23 except Exception as e:
24     print(f"[ERROR] Arduino: {e}")

```

```

[Done] exited with code=1 in 0.534 seconds

[Running] python -u "c:\Users\alosh\OneDrive\Рабочий стол\diplom\face_tracker.py"
[INFO] Arduino ready.
[INFO] Connecting to ESP32-CAM...
[INFO] Stream connected.

```

Рисунок 3.13 – Перевірка справності підключаення до веб сервера та Arduino

Коли обличчя не потрапляє в центральну ділянку кадру ("мертву зону"), програма обчислює величину відхилення та генерує команду для сервоприводу. У цей момент Arduino приймає числовий вираз кута і переналаштовує положення привода. Тестування цього кроку потрібно, аби впевнитися, що передача інформації між комп'ютером та Arduino відбувається без затримок, і що серво реагує коректно та надійно (рис. 3.14).

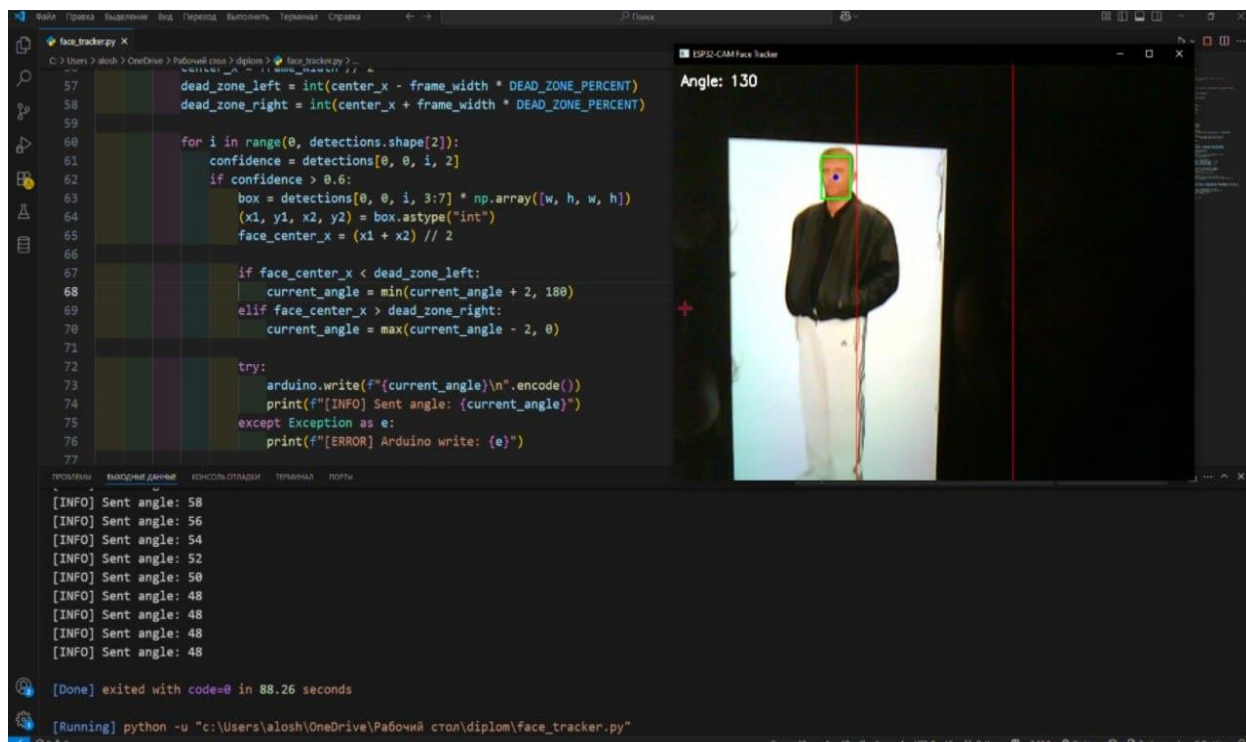


Рисунок 3.14 – Перевірка передачі інформації на Arduino

В ході тестування, що було здійснено, засвідчено надійну функціональність усіх компонентів системи. Це стосується як процесів обміну інформацією, так і активації виконавчих вузлів. Система демонструє здатність працювати в режимі реального часу, оперативно реагуючи на ідентифікацію об'єктів та правильно здійснюючи обчислення кутових параметрів.

РОЗДІЛ 4 БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ

4.1 Менеджмент безпеки

Менеджмент безпеки – це комплекс організаційних, технічних, аналітичних та превентивних заходів, метою яких є убезпечення життя, здоров'я, майна та інфраструктури. Він включає системний аналіз ризиків, оцінювання їх наслідків, розробку заходів реагування, а також впровадження систем моніторингу та постійного контролю. У ХХІ столітті важливість менеджменту безпеки зростає у зв'язку з цифровізацією, автоматизацією та розвитком інтелектуальних систем. У рамках розробки комп'ютеризованої системи багатокамерного відеозапису динамічних об'єктів, реалізованої на базі ESP32-CAM, OpenCV та Arduino, закладено значний потенціал для розширеного використання в сфері управління техногенною та виробничою безпекою. Система, хоч і призначалася для виявлення та стеження за обличчям, має архітектуру, яка легко адаптується та перепрограмовується під різні потреби безпеки:

- потенціал системи в контексті менеджменту безпеки;
- виявлення небезпечних подій в реальному часі.

Змінивши програмну логіку обробки відео, OpenCV-аналізатор може визначати не тільки обличчя, але й різноманітні аномалії: падіння людини, проникнення в заборонену зону, появу підозрілих об'єктів (залишені речі, що становлять загрозу). Такі події можуть автоматично передаватися до центру управління для подальших дій.

Традиційні системи управління ризиками базуються на звітності, аудитах та постфактумному аналізі інцидентів. Розроблена система додає активний елемент: вона не тільки фіксує ситуацію, а й приймає рішення в

					КС КРБ 123.244.00.00 ПЗ		
Змн.	Арк.	№ докум.	Підпис	Дата			
Розроб.		Чорняк О.В.			Безпека життєдіяльності, основи охорони праці		
Перевірив		Тиш Є.В					
Консульт.		Пилипкць М.І.					
Н. Контр.		Луцик Н.С.					
Затверд.		Осухівська Г.М.					
					Літ.	Арк.	Аркушів
						48	
					ТНТУ, каф. КС, гр. СІ-42		

реальному часі. Це дає змогу зменшити реактивність та перейти до проактивного реагування.

У багатьох випадках людський фактор (втома, неуважність, упередженість) зменшує ефективність заходів безпеки. Використання автономних відеосистем на основі штучного інтелекту дозволяє уніфікувати підхід до оцінки ситуацій, мінімізуючи суб'єктивність. Камера реагує тільки на те, що вона "бачить", та оцінює ситуацію строго згідно з попередньо заданими критеріями.

На відміну від стаціонарних рішень, система на базі ESP32-CAM є компактною, портативною та легко масштабованою. Це ідеально підходить для об'єктів, що змінюють конфігурацію (будівельний майданчик, зона ремонту, зона евакуації чи масового скупчення людей). Систему можна швидко розгорнути в новому місці без складної інфраструктури.

У разі підключення до центрального комп'ютера або сервера, кожна подія (виявлення людини, рух, відхилення від траєкторії) може бути зареєстрована в базі даних із зазначенням часу та координат. Це дає можливість створити цифрове відображення безпеки об'єкта та проводити аудит, який може використовуватися під час службових розслідувань, навчання або під час перевірок контролюючими органами.

Завдяки відкритій архітектурі, система легко інтегрується з іншими елементами управління безпекою: пожежною сигналізацією, контролем доступу, системами SCADA, базами даних персоналу. Така інтеграція дозволяє побудувати єдину платформу управління безпекою, де відеоаналіз буде одним із модулів.

Практичні приклади використання:

- у промисловості, моніторинг присутності персоналу у небезпечних зонах;
- на об'єктах критичної інфраструктури, виявлення сторонніх осіб біля генераторних блоків, електрощитових;

					КС КРБ 123.244.00.00 ПЗ	Арк.
						49
Змн.	Арк.	№ докум.	Підпис	Дата		

- у навчальних закладах або лікарнях: контроль поведінки в коридорах, запобігання скупченню людей, тривожна сигналізація при агресивних діях;
- у сільському господарстві; охорона техніки, фіксація руху в нічний час.

4.2 Електробезпека на будівельному майданчику

Забезпечення електробезпеки на будівельному майданчику — ключовий аспект охорони праці. Відкриті джерела напруги, рухома техніка, вологі поверхні та мінлива інфраструктура створюють серйозні ризики ураження електричним струмом як для працівників, так і для сторонніх осіб. Сучасні методи покращення безпеки будівельних робіт включають впровадження автоматизованих систем моніторингу, де особливе місце можуть займати компактні інтелектуальні системи на основі вбудованих модулів. Розроблена комп'ютеризована система багатокамерного відеозапису динамічних об'єктів, що базується на ESP32-CAM, Arduino та обробці відеопотоку через OpenCV, може бути адаптована для моніторингу доступу людей до зон підвищеної електробезпеки. Якщо переналаштувати програмну логіку з виявлення обличчя на відстеження присутності об'єкта у визначеній зоні кадру, пристрій зможе визначати присутність людей, що потрапили до забороненої або небезпечної області.

Завдяки вбудованій камері ESP32-CAM та бездротовій передачі відео, пристрій легко інтегрується у тимчасову інфраструктуру майданчика. Система може бути розміщена біля відкритих щитів живлення, трансформаторів, тимчасових розподільчих коробок або кабельних трас, де перебування без попередження — категорично неприпустиме. В такому разі система фіксує присутність людини у зоні ризику, проводить відеоаналіз та автоматично сповіщає чергового оператора або виконує запрограмовану дію — увімкнення сигнального освітлення, сирени чи надсилання push-сповіщення.

					КС КРБ 123.244.00.00 ПЗ	Арк.
						50
Змн.	Арк.	№ докум.	Підпис	Дата		

Окрім звичайного розпізнавання руху, система може бути вдосконалена для визначення осіб без засобів індивідуального захисту (ЗІЗ). Використовуючи попередньо навчену нейронну мережу, можливо реалізувати детекцію наявності каски, сигнального жилета або спеціального одягу. При виявленні людини без ЗІЗ у небезпечній зоні, система згенерує окреме сповіщення, яке вказуватиме на порушення правил техніки безпеки. Особливо важлива функція ідентифікації сторонніх осіб, котрі не мають дозволу до території. На відміну від традиційних охоронних систем, заснованих на RFID або пропускній системі, дана реалізація не вимагає носіння жетонів, браслетів чи радіопередавачів — контроль здійснюється візуально, за принципом "бачу — реаую". Це дозволяє виявити проникнення навіть у випадках вимкнення живлення іншого обладнання або при обході офіційного входу. В умовах динамічного та мінливого середовища будівельного майданчика (переміщення техніки, зміни планування, підключення/відключення ліній) розроблена система має перевагу в гнучкості та простоті налаштування. Один модуль можна встановити за лічені хвилини, живити від портативного акумулятора, передавати дані бездротовою мережею і не потребуватиме складного технічного обслуговування. У поєднанні з логуванням подій на ПК або сервері, така система стає частиною інтелектуального підходу до електробезпеки, де акцент робиться на превентивному контролі.

Комп'ютеризована система багатокамерного відеозапису динамічних об'єктів на базі ESP32-CAM з OpenCV — це не просто інструмент для стеження за динамічними об'єктами, а й потенційно універсальний пристрій контролю безпечного доступу до небезпечних зон.

					КС КРБ 123.244.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		51

ВИСНОВКИ

В результаті виконання кваліфікаційної роботи було виконано наступне

1. Створено функціональну комп'ютеризовану систему, що базується на ESP32-CAM, Arduino Nano та OpenCV, котра забезпечує безперервний відеозапис, обробку зображень в реальному часі та керування положенням камери залежно від координат виявленого обличчя. В основу розробки покладено застосування недорогих компонентів, що при цьому демонструють високу функціональність і здатність працювати автономно. Архітектура системи передбачає чітку взаємодію трьох компонентів: модуль відеозйомки (ESP32-CAM), обчислювальний модуль (персональний комп'ютер з Python та OpenCV) і виконавчий пристрій (Arduino з сервоприводом). Кожен модуль виконує свою власну функцію, що дозволяє досягти модульності, простоти оновлення та масштабування.

2. Реалізовано алгоритм розпізнавання обличчя у відеопотоці та автоматизоване управління поворотом камери, що гарантує стеження за об'єктом у реальному часі. Для розпізнавання обличчя використано нейромережеву модель SSD (Single Shot Detector), що дозволяє отримувати координати об'єкта на кожному кадрі з високою точністю. Далі координати обличчя зіставляються з центральною зоною кадру («мертвою зоною»), і при відхиленні формується команда на зміну положення платформи, що реалізується через Arduino та сервопривід MG996R. Такий підхід дає змогу динамічно стежити за переміщенням людини в полі зору камери без потреби ручного налаштування чи участі оператора. У процесі тестування система показала надійну роботу, швидке реагування, стабільність передачі відео та точність в обробці кадрів.

3. Оцінено практичні перспективи використання системи в контексті забезпечення безпеки, моніторингу доступу та автоматичного нагляду, особливо в умовах підвищеного ризику. Запропоноване рішення легко адаптується під інші завдання за рахунок відкритої архітектури та гнучкості

					КС КРБ 123.244.00.00 ПЗ	Арк.
						52
Змн.	Арк.	№ докум.	Підпис	Дата		

програмного забезпечення. Наприклад, шляхом модифікації логіки обробки відео можна реалізувати виявлення небезпечних ситуацій на будівельному майданчику, контроль наявності засобів індивідуального захисту у працівників (каска, жилети), виявлення несанкціонованого проникнення сторонніх осіб у контрольовані зони та інше. Таким чином, розроблена система може бути інтегрована у більші комплекси моніторингу безпеки, відеоаналітики або smart-системи нагляду за об'єктами інфраструктури.

У підсумку, результати кваліфікаційної роботи підтверджують доцільність створення подібних систем на базі вільного програмного забезпечення та доступного апаратного забезпечення. Це дозволяє вирішувати як навчальні, так і практичні задачі у сфері автоматизації, робототехніки та цифрового відеоспостереження, що є актуальним напрямом у контексті розвитку технологій штучного інтелекту та комп'ютерного зору.

					КС КРБ 123.244.00.00 ПЗ	Арк.
						53
Змн.	Арк.	№ докум.	Підпис	Дата		

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Жаровський Р.О., Луцик Н.С., Осухівська Г.М., Паламар А.М., Тиш Є.В. Методичні вказівки до виконання кваліфікаційної роботи бакалавра для здобувачів першого (бакалаврського) рівня вищої освіти за спеціальністю 123 «Комп'ютерна інженерія» усіх форм навчання. Тернопіль: ТНТУ, 2024. 39 с.
2. Espressif Systems. ESP32-CAM Technical Reference Manual. URL: <https://www.espressif.com/en/products/socs/esp32/resources> (дата затвердження 6.03.2025р).
3. Arduino. Огляд плати Arduino Nano. [Електронний ресурс] – Режим доступу [URL:https://docs.arduino.cc/hardware/nano](https://docs.arduino.cc/hardware/nano) (дата затвердження 6.03.2025р).
4. Микитишин А.Г., Митник М.М., Стухляк П.Д., Пасічник В.В. Комп'ютерні мережі. Книга 2. Львів: «Магнолія 2006», 2024. 328 с.
5. Брадскі, Г., та Келер, А. Вивчення OpenCV: Комп'ютерний зір з бібліотекою OpenCV. O'Reilly Media, 2008. 580 ст.
6. Паламар М.І., Стрембіцький М.О., Паламар А.М. Проектування комп'ютеризованих вимірювальних систем і комплексів. Навчальний посібник. Тернопіль: ТНТУ. 2019. 150 с.
7. Обласюк, А.М. Вбудовані системи керування: підручник. – Львів: Вид-во Львівської політехніки, 2019. 276 стор.
8. Kharchenko A., Bodnarchuk I., Yatsyn V. The Method for Comparative Evaluation of Software Architecture with Accounting of Trade-offs. American Journal of Information Systems. 2014. Vol. 2, No. 1. P. 20-25.
9. GitHub. ESP32-CAM Розпізнавання облич і приклади стрімінгу. [Електронний ресурс] Режим доступу: <https://github.com/espressif/esp32-camera>
10. Лупенко С.А., Пасічник В.В., Тиш Є.В. Комп'ютерна логіка. Навчальний посібник. Львів: Видавництво «Магнолія 2006», 2024. 354 с.
11. Lupenko S., Orobchuk O., Stadnik N., Zozulya A. Modeling and signals processing using cyclic random functions. In Proceedings of the 13th IEEE

					КС КРБ 123.244.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		54

International Scientific and Technical Conference on Computer Sciences and Information Technologies (CSIT). Lviv, Ukraine, 11–14 September, 2018. P. 360–366.

12. Тиш Є.В. Узагальнений алгоритм синтезу компонентів комп'ютерних систем на основі мікропрограмних автоматів. Вчені записки Таврійського національного університету імені В.І. Вернадського. Том 36 (75), № 1, 2025. С. 247-253.

13. Tysh Ie. Approach and method of evaluation of the general reliability indicator of computer systems. International scientific journal “Computer systems and information technologies”, 3 (5). Khmelnytskyi National University. 2021. P.74-80.

14. Тиш Є.В., Шалапай Р.І. Типи вимог до комп'ютерних систем і методи їх виявлення. Матеріали XII міжнародної науково-практичної конференції молодих учених та студентів «Актуальні задачі сучасних технологій» (6-7 грудня 2023 року). Тернопіль: ТНТУ. 2023. С. 437.

15. Palamar A., Karpinski M., Palamar M., Osukhivska H., Mytnyk M. Remote Air Pollution Monitoring System Based on Internet of Things. CEUR Workshop Proceedings, 2nd International Workshop on Information Technologies: Theoretical and Applied Problems (ITTAP 2022), Ternopil, Ukraine, November 22–24, 2022. Vol. 3309. P. 194-204.

16. Palamar A., Palamar M., Osukhivska H. Real-time Health Monitoring Computer System Based on Internet of Medical Things. CEUR Workshop Proceedings, 3rd International Workshop on Information Technologies: Theoretical and Applied Problems (ITTAP 2023), Ternopil, Ukraine, Opole, Poland, November 22–24, 2023. Vol. 3628. P. 106-115.

17. Скобло Ю. С, Соколовська Т. Б., Мазоренко Д. І., Б 40 Тіщенко Л. М., Троянов М. М. Безпека життєдіяльності, 2003. 424 с

					КС КРБ 123.244.00.00 ПЗ	Арк.
						55
Змн.	Арк.	№ докум.	Підпис	Дата		

Додаток А

Технічне завдання

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії

Кафедра комп'ютерних систем та мереж

“Затверджую”

Завідувач кафедри КС

_____ Осухівська Г.М.

“ ____ ” _____ 2025 р

Комп'ютеризована система багатокамерного відеозапису динамічних об'єктів

ТЕХНІЧНЕ ЗАВДАННЯ

на 10 листках

Вид робіт:

Кваліфікаційна робота

На здобуття освітнього ступеня «Бакалавр»

Спеціальність 123 «Комп'ютерна інженерія»

«УЗГОДЖЕНО»

«ВИКОНАВЕЦЬ»

Керівник кваліфікаційної роботи

Студент групи СІ-42

_____ к.т.н., доц. Тиш Є.В

_____ Чорняк О.В

« ____ » _____ 2025 р.

« ____ » _____ 2025 р.

Тернопіль 2025

1 Загальні відомості

1.1 Повна назва та її умовне позначення

Повна назва теми кваліфікаційної роботи: «Комп'ютеризована система багатокамерного відеозапису динамічних об'єктів».

Умовне позначення кваліфікаційної роботи: КС КРБ 123.244.00.00

1.2 Виконавець

Студент групи СІ-42, факультету комп'ютерно-інформаційних систем і програмної інженерії, кафедри комп'ютерних систем та мереж, Тернопільського національного технічного університету імені Івана Пулюя, Чорняк О.В.

1.3 Підстава для виконання роботи

Підставою для виконання кваліфікаційної роботи є наказ по університету (№4/7-53 від 27.01.2025 р.)

1.4 Планові терміни початку та завершення роботи

Плановий термін початку виконання кваліфікаційної роботи – 27.01.2025 р.

Плановий термін завершення виконання кваліфікаційної роботи – 23.06.2025 р.

1.5 Порядок оформлення та пред'явлення результатів роботи

Порядок оформлення пояснювальної записки та графічного матеріалу здійснюється у відповідності до чинних норм та правил ISO, ЕСКД, ЕСПД та ДСТУ.

Пред'явлення проміжних результатів роботи з виконання кваліфікаційної роботи здійснюється у відповідності до графіку, затвердженого керівником роботи. Попередній захист кваліфікаційної роботи відбувається при готовності роботи – наявності пояснювальної записки та графічного матеріалу.

Пред'явлення результатів кваліфікаційної роботи відбувається шляхом захисту на відповідному засіданні ЕК, ілюстрацією основних досягнень за допомогою графічного матеріалу.

2 Призначення і цілі створення системи

2.1 Призначення системи

Система, що розробляється, призначена для:

- автоматичного спостереження за рухомими об'єктами в полі зору;
- утримання об'єкта зйомки в центрі кадру за допомогою повороту смартфона чи камери.

2.2 Мета створення системи

Метою кваліфікаційної роботи є розробка системи відеозйомки з функцією автоматичного наведення на обличчя особи.

2.3 Характеристика об'єкту

Розробляювана система призначено для використання для зйомки відео без участі оператора.

3 Вимоги до системи

3.1 Вимоги до системи в цілому

3.1.1 Вимоги до структури та функціонування системи

Система має складатися з:

- вбудованого модуля ESP32-CAM;
- програмного забезпечення на персональному комп'ютері;
- контролера Arduino Nano;
- сервоприводу з платформою для відеокамери.

3.1.2 Вимоги до способів та засобів зв'язку між компонентами системи

Основна вимога, що висувається до методів та засобів інформаційного обміну – це їх відповідність.

Елементи системи мусять взаємодіяти за допомогою наступних каналів:

- ESP32-CAM передає відеопотік на персональний комп'ютер через Wi-Fi з використанням протоколу HTTP;
- персональний комп'ютер обробляє відеопотік і передає координати обличчя на мікроконтролер Arduino Nano через послідовний інтерфейс UART;
- мікроконтролер керує сервоприводом через цифровий сигнал PWM.

Усі засоби зв'язку повинні бути відповідні за швидкістю, форматом даних і логікою обміну, що забезпечує надійність і безперервність функціонування системи в реальному часі.

3.1.3 Вимоги до режимів функціонування системи

Для системи визначено два режими роботи:

- Нормальний режим роботи;

- Аварійний режим роботи.

Головним режимом роботи є звичайний режим. Для забезпечення звичайного режиму роботи системи необхідно виконувати вимоги та дотримуватись умов експлуатації програмного забезпечення та комплексу технічних засобів системи, зазначені у відповідних технічних документах

Аварійний режим функціонування системи визначається втратою з'єднання з одним чи кількома компонентами. При цьому система мусить утримувати теперішнє положення чи автоматично відновлювати з'єднання.

3.1.4 Вимоги по діагностуванню системи

Для діагностування системи використовуються:

- інструменти моніторингу програмного забезпечення (термінал, логи-файли);
- інструменти діагностики апаратної частини (послідовний монітор Arduino, індикатори приєднання);
- елементи самодіагностики, вмонтовані у середовище.

Інструменти мають забезпечувати зручний інтерфейс задля можливості перегляду діагностичних подій, моніторингу виконання головних процесів.

3.1.5 Перспективи розвитку, проектування системи

Цю систему можна вдосконалити через застосування додаткових програмних складників, а також підключення декількох камер до одного ПК задля створення багатокамерної системи з розширеними функціями спостереження.

3.2 Показники призначення

Система має передбачати можливість розширення. Здібності масштабування мають гарантуватися засобами використовуваного базового програмного та технічного забезпечення.

Система електроживлення має забезпечувати захисне вимикання при перевантаженнях і коротких замиканнях в колах навантаження, а також аварійне ручне вимкнення.

Загальні вимоги пожежної безпеки мають відповідати нормам на побутове електрообладнання. У разі пожежі не повинні виділятися отруйні гази та дим. Після зняття електроживлення має бути доступне застосування будь-яких засобів пожежогасіння.

3.2.1 Вимоги до надійності

Система повинна забезпечувати працездатність та відновлення своїх функцій при виникненні наступних ситуацій:

- при збоях в системі електропостачання апаратної частини;
- при помилках в роботі апаратних засобів (Сервопривід, камера);
- при помилках, пов'язаних з програмним забезпеченням (ОС і драйвери пристроїв).

Для захисту апаратури від стрибків напруги і комутаційних завад повинні стабілізатори живлення та конденсатори.

3.3 Вимоги до безпеки

Зовнішні елементи технічних засобів системи, що перебувають під напругою, повинні мати захист від випадкового дотику, а самі технічні засоби мати занулення або захисне заземлення .

Система електроживлення повинна забезпечувати захисне вимикання при перевантаженнях і коротких замиканнях в колах навантаження, а також аварійне ручне вимикання.

Загальні вимоги пожежної безпеки повинні відповідати нормам на побутове електрообладнання. У разі пожежі не мають виділятися отруйні гази і дим. Після зняття електроживлення має бути доступне застосування будь-яких засобів пожежогасіння.

3.3.1 Вимоги до експлуатації, технічного обслуговування, ремонту і зберігання компонентів системи

Вимоги до експлуатації, технічного обслуговування, ремонту та зберігання компонентів системи

Мікроклімат в приміщеннях повинен відповідати нормам виробничого мікроклімату за ДСН 3.3.6.042-99:

- температуру повітря в межах від $+10^{\circ}\text{C}$ до $+35^{\circ}\text{C}$;
- відносну вологість повітря при 25°C в межах від 30% до 80%;
- атмосферний тиск 760 ± 25 мм рт. ст.

Періодичне технічне обслуговування використаних технічних засобів має здійснюватися відповідно до вимог технічної документації, але не рідше ніж один раз на рік.

Періодичне технічне обслуговування і тестування технічних засобів мають містити обслуговування та тестування всіх використаних засобів, зокрема: модулів ESP32-CAM, Arduino Nano, сервоприводів, джерел живлення та елементів комунікації.

На підставі результатів тестування технічних засобів повинні проводитися аналіз причин появи виявлених дефектів і вживатися заходи щодо їх ліквідації.

3.4 Вимоги до захисту інформації від несанкціонованого доступу

Система мусить забезпечувати захист від несанкціонованого доступу на рівні не нижче визначеного вимогами, що висуваються до категорії 1Д за класифікацією документа, який чинний, “Автоматизовані системи. Захист від несанкціонованого доступу до інформації. Класифікація автоматизованих систем”.

Компоненти підсистеми захисту від НСД мусять забезпечувати:

- ідентифікацію користувача;
- перевірку повноважень користувача під час роботи з системою;
- розмежування доступу користувачів.

Рівень захищеності від несанкціонованого доступу засобів обчислювальної техніки, що виконують обробку конфіденційної інформації, повинен відповідати вимогам класу захищеності згідно з вимогами документу “Засоби обчислювальної техніки. Захист від несанкціонованого доступу до інформації. Показники захищеності від несанкціонованого доступу до інформації.”

3.4.1 Вимоги по збереженню інформації при аваріях

Інформація, у разі виникнення аварійних випадків, мусить бути збережена на запасних носіях, чи дубльована у вигляді журналів логування в межах наявної файлової системи комп'ютера.

3.4.2 Вимоги по стандартизації і уніфікації

Система мусить відповідати вимогам ергономіки та зручності використання за умови комплектування високоякісним обладнанням (ЕОМ, монітор та інше устаткування), яке має необхідні сертифікати відповідності та безпеки.

3.4.3 Вимоги до функцій (завдань), що виконуються системою:

- забезпечення зручного графічного інтерфейсу користувача;
- виявлення облич у відеопотоці;

- передача координат обличчя на мікроконтролер;

4 Вимоги до документації

Документація повинна відповідати вимогам ЄСКД та ДСТУ

Комплект документації повинен складатись з:

- пояснювальної записки;
- графічного матеріалу:
 - а) Структурна схема.
 - б) Блок схема роботи комп'ютеризованої системи.
 - в) Схема електрична принципова.
 - г) Схема електрична монтажна (з'єднань).

*Примітка: У комплект документації можуть вноситися зміни та доповнення в процесі розробки.

5 Стадії та етапи проектування

Таблиця 1 – Стадії та етапи виконання кваліфікаційної роботи бакалавра

№ етапу	Назва етапу виконання кваліфікаційної роботи	Термін виконання
1	Розробка технічного завдання.	27.01.2025р. – 02.02.2025р.
2	Аналіз задачі	02.05 – 05.05
3	Ознайомлення з документацією ESP32 cam	08.05 – 10.05
4	Розробка та збір комп'ютеризованої системи	11.05 – 22.05
5	Безпека життєдіяльності, основи охорони праці	23.05 – 25.05
6	Оформлення пояснювальної записки і графічного матеріалу	26.05 – 02.06
7	Перевірка на академічний плагіат, перевірка керівником та консультантами	9.06.2025р. – 15.06.2025р.
8	Попередній захист кваліфікаційної роботи бакалавра	16.06.2025р. – 22.06.2025р.
9	Захист кваліфікаційної роботи бакалавра	24.06

6 Додаткові умови виконання кваліфікаційної роботи

Під час виконання кваліфікаційної роботи у дане технічне завдання можуть вноситися зміни та доповнення.

Додаток Б

Лістинг програми (ESP32 cam)

```
#include "esp_camera.h"
#include <WiFi.h>

// Вибрана модель камери (AI Thinker з PSRAM)
#define CAMERA_MODEL_AI_THINKER
#include "camera_pins.h"

// Налаштування WiFi
const char *ssid = "Alexey";
const char *password = "wifi537386";

void startCameraServer();
void setupLedFlash(int pin);

void setup() {
    Serial.begin(115200);
    Serial.setDebugOutput(true);
    Serial.println();

    // Конфігурація камери
    camera_config_t config;
    config.ledc_channel = LEDC_CHANNEL_0;
    config.ledc_timer = LEDC_TIMER_0;
    config.pin_d0 = Y2_GPIO_NUM;
    config.pin_d1 = Y3_GPIO_NUM;
    config.pin_d2 = Y4_GPIO_NUM;
    config.pin_d3 = Y5_GPIO_NUM;
    config.pin_d4 = Y6_GPIO_NUM;
    config.pin_d5 = Y7_GPIO_NUM;
    config.pin_d6 = Y8_GPIO_NUM;
    config.pin_d7 = Y9_GPIO_NUM;
    config.pin_xclk = XCLK_GPIO_NUM;
    config.pin_pclk = PCLK_GPIO_NUM;
    config.pin_vsync = VSYNC_GPIO_NUM;
    config.pin_href = HREF_GPIO_NUM;
    config.pin_sccb_sda = SIOD_GPIO_NUM;
    config.pin_sccb_scl = SIOC_GPIO_NUM;
    config.pin_pwdn = PWDN_GPIO_NUM;
    config.pin_reset = RESET_GPIO_NUM;
    config.xclk_freq_hz = 20000000;
    config.frame_size = FRAMESIZE_SVGA; // Роздільна здатність
800x600
    config.pixel_format = PIXFORMAT_JPEG; // Формат зображення для
поточної передачі
    config.grab_mode = CAMERA_GRAB_WHEN_EMPTY;
    config.fb_location = CAMERA_FB_IN_PSRAM;
    config.jpeg_quality = 10; // Якість JPEG (1-63, де
1 – найкраща)
    config.fb_count = 1;
```

```

// Якщо PSRAM доступний, збільшуємо якість та буферизацію
if (config.pixel_format == PIXFORMAT_JPEG && psramFound()) {
    config.jpeg_quality = 5;
    config.fb_count = 2;
    config.grab_mode = CAMERA_GRAB_LATEST;
}

// Ініціалізація камери
esp_err_t err = esp_camera_init(&config);
if (err != ESP_OK) {
    Serial.printf("Помилка ініціалізації камери: 0x%x", err);
    return;
}

// Додаткові налаштування сенсора (яскравість, контраст тощо)
sensor_t *s = esp_camera_sensor_get();
if (s->id.PID == OV3660_PID) {
    s->set_vflip(s, 1); // Відзеркалення по
    // вертикалі
    s->set_brightness(s, 3); // Яскравість
    s->set_saturation(s, -2); // Насиченість кольорів
    s->set_contrast(s, 1); // Контрастність
    s->set_gainceiling(s, (gainceiling_t)3); // Обмеження
    // підсилення (зменшує шум)
}

// Підключення до WiFi
WiFi.begin(ssid, password);
WiFi.setSleep(false);

Serial.print("Підключення до WiFi...");
while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
}
Serial.println("\nWiFi підключено");

// Запуск сервера для потокової передачі з камери
startCameraServer();

Serial.print("Камера готова! Використовуйте URL: http://");
Serial.print(WiFi.localIP());
Serial.println();
}

void loop() {
    // Автоматичний перезапуск при втраті з'єднання з WiFi
    if (WiFi.status() != WL_CONNECTED) {
        Serial.println("[ПОМИЛКА] З'єднання з WiFi втрачено.
        Перезавантаження...");
        delay(3000);
        ESP.restart();
    }
}

```

```

    }
    delay(10000); // Пауза між перевітками
}

```

Лістинг програми (OpenCV)

```

import cv2
from urllib.request import urlopen
import numpy as np
import serial
import time

# Параметри підключення
ESP32_STREAM_URL = 'http://192.168.31.103:81/stream' # URL
відеопотоку з камери
SERIAL_PORT = 'COM4' # Порт для підключення Arduino
BAUD_RATE = 9600 # Швидкість передачі даних
DEAD_ZONE_PERCENT = 0.15 # Відсоток "мертвої зони" від центру
current_angle = 90 # Початковий кут сервоприводу

# Завантаження моделі для розпізнавання обличь
net = cv2.dnn.readNetFromCaffe("deploy.prototxt",
"res10_300x300_ssd_iter_140000.caffemodel")

# Ініціалізація підключення до Arduino
try:
    arduino = serial.Serial(SERIAL_PORT, BAUD_RATE)
    time.sleep(2) # Чекаємо ініціалізації
    print("[ІНФО] Arduino готовий.")
    arduino.write(f"{current_angle}\n".encode()) # Відправляємо
початковий кут
except Exception as e:
    print(f"[ПОМИЛКА] Arduino: {e}")
    exit()

# Підключення до відеопотоку
print("[ІНФО] Підключення до ESP32-CAM...")
try:
    stream = urlopen(ESP32_STREAM_URL)
    print("[ІНФО] Потік підключено.")
except Exception as e:
    print(f"[ПОМИЛКА] ESP32-CAM: {e}")
    exit()

bytes_data = b''

# Головний цикл обробки кадрів
while True:
    try:
        # Отримання та обробка кадру з потоку
        bytes_data += stream.read(1024)
        start = bytes_data.find(b'\xff\xd8') # Початок JPEG
        end = bytes_data.find(b'\xff\xd9') # Кінець JPEG

```

```

        if start != -1 and end != -1:
            jpg = bytes_data[start:end+2]
            bytes_data = bytes_data[end+2:]
            frame = cv2.imdecode(np.frombuffer(jpg,
dtype=np.uint8), cv2.IMREAD_COLOR)

            # Підготовка зображення для нейромережі
            (h, w) = frame.shape[:2]
            blob = cv2.dnn.blobFromImage(cv2.resize(frame, (300,
300)), 1.0,
                                           (300, 300), (104.0, 177.0,
123.0))

            net.setInput(blob)
            detections = net.forward()

            # Визначення "мертвої зони"
            center_x = w // 2
            dead_zone_left = int(center_x - w *
DEAD_ZONE_PERCENT)
            dead_zone_right = int(center_x + w *
DEAD_ZONE_PERCENT)

            # Обробка виявлених обличч
            for i in range(0, detections.shape[2]):
                confidence = detections[0, 0, i, 2] #
Впевненість у розпізнаванні
                if confidence > 0.6: # Якщо впевненість > 60%
                    box = detections[0, 0, i, 3:7] *
np.array([w, h, w, h])
                    (x1, y1, x2, y2) = box.astype("int")
                    face_center_x = (x1 + x2) // 2 # Центр
обличчя

                    # Корекція кута сервоприводу
                    if face_center_x < dead_zone_left:
                        current_angle = min(current_angle + 2,
180) # Рух вліво

                    elif face_center_x > dead_zone_right:
                        current_angle = max(current_angle - 2,
0) # Рух вправо

                    # Відправка нового кута на Arduino
                    try:

                        arduino.write(f"{current_angle}\n".encode())
                        print(f"[ІНФО] Відправлено кут:
{current_angle}")

                    except Exception as e:
                        print(f"[ПОМИЛКА] Помилка запису в
Arduino: {e}")

            # Візуалізація
            cv2.rectangle(frame, (x1, y1), (x2, y2), (0,

```

```

255, 0), 2)
        cv2.circle(frame, (face_center_x, (y1 + y2)
// 2), 4, (255, 0, 0), -1)
        break # Обробляємо тільки одне обличчя

        # Візуалізація "мертвої зони"
        cv2.line(frame, (dead_zone_left, 0),
(dead_zone_left, h), (0, 0, 255), 1)
        cv2.line(frame, (dead_zone_right, 0),
(dead_zone_right, h), (0, 0, 255), 1)

        # Відображення інформації
        cv2.putText(frame, f"Кут: {current_angle}", (10,
30),
                                cv2.FONT_HERSHEY_SIMPLEX, 0.7, (255,
255, 255), 2)
        cv2.imshow("Відстеження обличчя ESP32-CAM", frame)

        # Вихід при натисканні 'q'
        if cv2.waitKey(1) & 0xFF == ord('q'):
            break
    except Exception as e:
        print(f"[ПОМИЛКА] Помилка читання потоку: {e}")
        break

# Завершення роботи
arduino.close()
cv2.destroyAllWindows()

```