

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет _____ Комп'ютерних наук
(повна назва)
Кафедра _____ Штучного інтелекту
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти _____ перший (бакалаврський)

_____ Інтелектуальна система визначення стану водія для
_____ підвищення безпеки руху
(тема)

Виконав:
здобувач _____ четвертого _____ року навчання,
групи _____ ІТШ-22-2

_____ Артур НІКІТЕНКО
(власне ім'я, прізвище)

Спеціальність _____ 122 Комп'ютерні науки

(код і повна назва спеціальності)

Тип програми _____ освітньо-професійна

Освітня програма _____ Штучний інтелект

(повна назва освітньої програми)

Керівник _____ проф. Марина КУДРЯВЦЕВА
(посада, власне ім'я, прізвище)

Допускається до захисту

Завідувач кафедри ШІ

(підпис)

_____ Лариса ЧАЛА
(власне ім'я, прізвище)

2026 р.

Харківський національний університет радіоелектроніки

Факультет _____ Комп'ютерних наук _____

Кафедра _____ Штучного інтелекту _____

Рівень вищої освіти _____ перший (бакалаврський) _____

Спеціальність _____ 122 Комп'ютерні науки _____
(код і повна назва)

Тип програми _____ освітньо-професійна _____

Освітня програма _____ Штучний інтелект _____
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____

(підпис)

« _____ » _____ 20 ____ р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві _____ Нікітенку Артуру Глібовичу _____
(прізвище, ім'я, по батькові)

1. Тема роботи _____ Інтелектуальна система визначення стану водія для підвищення безпеки руху _____

затверджена наказом університету від 5 червня 2026 р. № 488Ст

2. Термін подання здобувачем роботи до екзаменаційної комісії 9 липня 2026 р.

3. Вихідні дані до роботи Науково-технічні публікації з методів комп'ютерного зору, офіційна документація Python, OpenCV та Dlib, попередньо навчена модель ідентифікації ключових точок обличчя, набір текстових відеоданих

4. Перелік питань, що потрібно опрацювати в роботі _____

1) Аналіз предметної галузі та постановка задачі _____

2) Формування вимог до програмної системи _____

3) Архітектура та проектування системи _____

4) Опис програмної реалізації _____

5) Експериментальні дослідження та тестування системи _____

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Строк / термін виконання етапів роботи	Примітка
1	Отримання завдання на кваліфікаційну роботу	08.06.2026	виконано
2	Аналіз предметної галузі	09.06.2026	виконано
3	Дослідження алгоритмів комп'ютерного зору та біометричних маркерів	11.06.2026	виконано
4	Проектування архітектури системи та модуля дистанційної фотоплетизмографії	13.06.2026	виконано
5	Програмна реалізація системи	15.06.2026	виконано
6	Проведення експериментальних досліджень	17.06.2026	виконано
7	Написання пояснювальної записки	18.06.2026	виконано
8	Захист перед ЕК	09.07.2026	

Дата видачі завдання 8 червня 20 26 р.

Здобувач 
(підпис)

Керівник роботи _____
(підпис)

проф. Марина КУДРЯВЦЕВА
(посада, власне ім'я, прізвище)

РЕФЕРАТ

Пояснювальна записка: 63 с., 18 рис., 3 табл., 2 дод., 9 джерел.

БІОМЕТРІЯ, КОМП'ЮТЕРНИЙ ЗІР, МАШИННЕ НАВЧАННЯ, МОНІТОРИНГ СТАНУ ВОДІЯ, НЕЙРОННІ МЕРЕЖІ, ОБРОБКА ВІДЕОПОТОКУ, DLIB, NUMPY, OPENCV, PYTHON, rPPG.

Об'єкт дослідження – процес автоматизованого моніторингу психофізіологічного стану людини в режимі реального часу.

Предмет дослідження – моделі, методи та алгоритми комп'ютерного зору для безконтактної оцінки втоми та рівня концентрації уваги водія.

Мета роботи – реалізація методів комп'ютерного зору для безконтактного контролю втоми водія та підготовка бази для виконання кваліфікаційної роботи.

Методи дослідження – методи цифрової обробки зображень та відеопотоку, алгоритми машинного навчання для локалізації антропометричних точок обличчя, геометричний аналіз, спектральний аналіз біосигналів.

У результаті практики проаналізовано існуючі системи моніторингу. Обґрунтовано вибір технологічного стеку на базі мови програмування Python та бібліотек комп'ютерного зору OpenCV і Dlib. Сформульовано архітектуру мультимодального програмного комплексу.

У ході індивідуального завдання практично розроблено алгоритми обчислення коефіцієнтів відкритості очей (EAR) та рота (MAR) для метричної детекції мікросну й позіхання. Запропоновано та імплементовано інноваційний модуль дистанційної фотоплетизмографії (rPPG) для безконтактного вимірювання серцевого ритму з відеопотоку. Отримані результати, написаний програмний код та зібрані матеріали є повноцінним підґрунтям для написання та успішного захисту кваліфікаційної роботи.

ABSTRACT

Bachelor's thesis contains: 63 pp., 18 fig., 3 tabl., 2 ann., 9 references.

BIOMETRICS, COMPUTER VISION, DLIB, DRIVER CONDITION MONITORING, MACHINE LEARNING, NEURAL NETWORKS, NUMPY, OPENCV, PYTHON, rPPG, VIDEO STREAM PROCESSING.

Object of research – the process of automated monitoring of a person's psychophysiological state in real time.

Subject of research – computer vision models, methods, and algorithms for non-contact assessment of driver fatigue and concentration levels.

Purpose of the work – implementation of computer vision methods for non-contact monitoring of driver fatigue and preparation of the basis for the qualification work.

Research methods – methods of digital image and video stream processing, machine learning algorithms for the localization of facial anthropometric landmarks, geometric analysis, and spectral analysis of biosignals.

As a result of the practical training, existing monitoring systems were analyzed. The choice of a technology stack based on the Python programming language and the OpenCV and Dlib computer vision libraries was justified. The architecture of a multimodal software complex was formulated.

During the individual task, algorithms for calculating the Eye Aspect Ratio and Mouth Aspect Ratio coefficients for the metric detection of microsleep and yawning were practically developed. An innovative remote photoplethysmography module was proposed and implemented for non-contact heart rate measurement from a video stream. The obtained results, written source code, and collected materials serve as a full-fledged foundation for the writing and successful defense of the qualification work.

ЗМІСТ

Перелік умовних позначень, символів, одиниць, скорочень і термінів	8
Вступ.....	9
1 Аналіз предметної галузі та постановка задачі.....	11
1.1 Опис предметної галузі та аналіз факторів ризику	11
1.2 Порівняльний аналіз існуючих систем та методів моніторингу	12
1.3 Огляд комерційних систем моніторингу водія	14
1.4 Аналіз методів розпізнавання обличчя та виділення ознак.....	15
1.5 Обґрунтування вибору технологій комп'ютерного зору.....	16
1.6 Постановка задачі.....	18
2 Формування вимог до програмної системи.....	19
2.1 Мета проєкту	19
2.2 Припущення та залежності	19
2.3 Функціональні вимоги	21
2.4 Нефункціональні вимоги.....	22
2.5 Вимоги до інтерфейсу користувача	23
3 Архітектура та проектування системи.....	24
3.1 UML проектування пз.....	24
3.1.1 Діаграма варіантів використання	24
3.1.2 Діаграма класів.....	25
3.1.3 Діаграма діяльності.....	25
3.1.4 Діаграма послідовностей.....	27
3.2 Архітектурна схема обчислювального конвеєра та математична модель локалізації обличчя	27
3.3 Математична модель оцінки стану очей.....	29
3.4 Математична модель детекції позіхання	30
3.5 Алгоритм дистанційного вимірювання пульсу.....	31
3.6 Логіка статистичного аналізу частоти кліпання	33
3.7 Кінцевий автомат прийняття рішень	33

4	Опис програмної реалізації	35
4.1	Вибір та обґрунтування інструментарію розробки	35
4.2	Структура програмного комплексу та основні модулі	36
4.3	Алгоритмічна реалізація функціональних модулів аналізу	37
4.3.1	Аналіз стану очей та лічильник моргань.....	37
4.3.2	Алгоритм часової фільтрації та таймінгів.....	38
4.3.3	Детекція відволікання та позіхання	39
4.4	Реалізація модуля безконтактного пульсометра.....	40
4.5	Система візуалізації та інтерфейс користувача	41
5	Експериментальні дослідження та тестування системи	43
5.1	Методика проведення експериментів та апаратне забезпечення.....	43
5.2	Математичні метрики оцінки якості алгоритмів	43
5.3	Тестування базового моніторингу: стан норми та відволікання.....	45
5.4	Дослідження реакції системи на мікросон та втому	46
5.5	Тестування модуля довгострокового аналізу.....	48
5.5.1	Тестування детекції стану стресу.....	48
5.5.2	Тестування детекції стану інтоксикації.....	50
5.6	Емпірична оцінка точності пульсометра	51
5.7	Оцінка впливу зовнішніх факторів та обмеження системи.....	52
5.8	Аналіз швидкодії та продуктивності конвеєра	53
	Висновки	54
	Перелік джерел посилання	56
	Додаток А Код програми.....	57
	Додаток Б Відомість кваліфікаційної роботи.....	62

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

ДТП – дорожньо-транспортна пригода;

ПЗ – програмне забезпечення;

ЦНС – центральна нервова система;

ADAS – Advanced Driver Assistance Systems – удосконалені системи
допомоги водієві;

AI – Artificial Intelligence – штучний інтелект;

CV – Computer Vision – комп'ютерний зір;

DMS – Driver Monitoring System – система моніторингу водія;

EAR – Eye Aspect Ratio – коефіцієнт відкритості ока;

FFT – Fast Fourier Transform – швидке перетворення Фур'є;

FPS – Frames Per Second – кадри на секунду;

GUI – Graphical User Interface – графічний інтерфейс користувача;

MAR – Mouth Aspect Ratio – коефіцієнт розкриття рота;

ROI – Region of Interest – область інтересу;

rPPG – Remote Photoplethysmography – дистанційна
фотоплетизмографія.

ВСТУП

Сучасний етап розвитку інформаційних технологій характеризується широким впровадженням методів штучного інтелекту та комп'ютерного зору в індустрію інтелектуальних транспортних систем. Забезпечення безпеки дорожнього руху залишається однією з найгостріших проблем суспільства, де людський фактор, зокрема втома, втрата концентрації або погіршення фізичного стану водія, є причиною переважної більшості дорожньо-транспортних пригод. Традиційні засоби моніторингу стану водія часто потребують використання контактних датчиків, що викликає дискомфорт, або аналізують поведінку автомобіля, реагуючи з великою затримкою. Тому розробка програмного забезпечення для безконтактного превентивного моніторингу психофізіологічного стану людини на основі відеопотоку є вкрай актуальним завданням.

Мета роботи – реалізація методів комп'ютерного зору для безконтактного контролю втоми водія та підготовка бази для виконання кваліфікаційної роботи.

Для детального розкриття поставленої мети в ході виконання індивідуального завдання було проаналізовано існуючі системи моніторингу (DMS) та обґрунтовано вибір технологічного стеку на базі мови програмування Python і бібліотек комп'ютерного зору OpenCV та Dlib. Сформульована архітектура мультимодального програмного комплексу передбачає практичну розробку алгоритмів обчислення коефіцієнтів відкритості очей (EAR) та рота (MAR) для метричної детекції мікросну й позіхання. Крім того, запропоновано та імплементовано інноваційний модуль дистанційної фотоплетизмографії (rPPG) для безконтактного вимірювання серцевого ритму з відеопотоку. Отримані результати, написаний програмний код та зібрані матеріали утворюють повноцінне теоретичне та практичне підґрунтя для подальшого написання та успішного захисту дипломного проєкту.

Об'єкт дослідження – процес автоматизованого моніторингу психофізіологічного стану людини в режимі реального часу.

Предмет дослідження – моделі, методи та алгоритми комп'ютерного зору для безконтактної оцінки втоми та рівня концентрації уваги водія.

Методи дослідження – методи цифрової обробки зображень та відеопотоку, алгоритми машинного навчання для локалізації антропометричних точок обличчя, геометричний аналіз, спектральний аналіз біосигналів.

Пояснювальна записка складається зі вступу, чотирьох розділів, висновків, переліку джерел посилання та додатків. У першому розділі проведено аналіз предметної галузі, розглянуто існуючі системи DMS, визначено ключові фактори ризику та сформульовано постановку задачі. У другому розділі визначено функціональні та нефункціональні вимоги до програмної системи, а також описано припущення й залежності її функціонування. У третьому розділі наведено математичний апарат системи, зокрема моделі EAR, MAR, алгоритми rPPG та логіку аналізу частоти кліпання. У четвертому розділі представлено результати експериментальних досліджень, наведено оцінку швидкодії та продемонстровано роботу інтерфейсу в різних сценаріях.

1 АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ ТА ПОСТАНОВКА ЗАДАЧІ

1.1 Опис предметної галузі та аналіз факторів ризику

Індустрія інтелектуальних транспортних систем (ITS) спрямована на мінімізацію ризиків дорожнього руху шляхом впровадження систем допомоги водієві (ADAS – Advanced Driver Assistance Systems). Згідно зі статистичними даними щодо безпеки дорожнього руху в Україні та світі, людський фактор залишається головною причиною переважної більшості дорожньо-транспортних пригод [1]. У цьому контексті системи моніторингу стану водія (Driver Monitoring Systems – DMS) набувають статусу обов'язкового стандарту безпеки для сучасних автомобілів [3].

Процес керування транспортним засобом вимагає безперервної концентрації уваги та швидкої реакції на динамічні зміни дорожньої обстановки. Дослідження у сфері безпеки руху показують, що критичними факторами, які призводять до втрати контролю над автомобілем, є [7]:

- втома та мікросон: фізіологічний стан, що супроводжується мимовільним закриттям очей на період від кількох десятих секунди до кількох секунд. Візуальними маркерами цього стану є зміна частоти кліпання, зниження м'язового тону обличчя та позіхання [6]. На швидкості дев'яносто км/год за 2 секунди мікросну автомобіль проїжджає понад 50 метрів у некерованому стані, що робить цей фактор найбільш смертоносним;

- відволікання уваги: перенесення погляду від дорожньої обстановки на побічні об'єкти (використання смартфона, налаштування мультимедійної системи, спілкування з пасажиром). Системи моніторингу класифікують цей стан за кутом повороту голови та напрямком погляду [4];

- стрес та інтоксикація: стани, що викликають критичне порушення когнітивних функцій. Вони супроводжуються аномальними змінами

вегетативних показників (тахікардія або брадикардія), а також порушенням природної частоти кліпання (тахікінезія або «застиглий погляд»).

Виявлення цих факторів ризику вимагає створення комплексних систем моніторингу, здатних аналізувати як геометричні характеристики обличчя, так і біометричні сигнали в режимі реального часу.

1.2 Порівняльний аналіз існуючих систем та методів моніторингу

Для вирішення задачі контролю стану водія сучасна індустрія пропонує кілька технологічних підходів. Для обґрунтування вибору технології розробки доцільно провести їх порівняльний аналіз. Існуючі методи моніторингу класифікуються на три основні категорії: аналіз поведінки автомобіля, контактні біометричні датчики та системи комп'ютерного зору (таблиця 1.1).

Таблиця 1.1 – Порівняльний аналіз методів моніторингу стану водія

Метод моніторингу	Принцип роботи	Переваги	Недоліки
Аналіз поведінки автомобіля	Відстеження траєкторії руху, фіксація перетину ліній розмітки, аналіз кута повороту керма.	Низька вартість, використання вбудованих датчиків автомобіля (ABS, ESP).	Висока інерційність (реакція настає, коли автомобіль вже почав аварійний маневр).

Продовження таблиці 1.1

Метод моніторингу	Принцип роботи	Переваги	Недоліки
Контактні біометричні датчики	Зчитування ЕКГ та пульсу через датчики, вбудовані у кермо, сидіння або смарт-годинник.	Медична точність вимірювання фізіологічних показників.	Дискомфорт для водія, чутливість до вологості рук, ризик втрати контакту при маневруванні.
Комп'ютерний зір (Безконтактний)	Аналіз відеопотоку обличчя та верхньої частини тулуба за допомогою камери.	Рання (превентивна) діагностика втоми, відсутність фізичного контакту, багатовекторність аналізу.	Залежність від умов освітлення, вимагає значних обчислювальних потужностей.

Детальний розгляд фізіологічних та апаратних особливостей функціонування кожного методу наведено нижче:

– аналіз поведінки автомобіля. Даний підхід базується на зборі телеметричних даних з CAN-шини автомобіля (кут повороту керма, сила натискання на педалі, датчики перетину ліній розмітки LDW). Незважаючи на простоту реалізації та відсутність дискомфорту для водія, цей метод має критичний недолік – надзвичайно високу інерційність. Система фіксує небезпеку лише тоді, коли автомобіль вже почав виконувати аварійний маневр, що унеможливорює превентивне попередження;

– контактні біометричні датчики. Використання електрокардіографічних (ЕКГ) сенсорів на кермі або смарт-годинників дозволяє з медичною точністю вимірювати пульс та провідність шкіри. Однак цей метод є непрактичним для повсякденного керування: він викликає дискомфорт, датчики чутливі до вологості рук, а при активному маневруванні (наприклад, перехопленні керма) контакт втрачається, що призводить до збою системи;

– безконтактний комп'ютерний зір (Computer Vision). Цей метод полягає в аналізі відеопотоку з камери, встановленої в салоні. Як зазначається у роботі [4], відеоаналітика дозволяє превентивно виявляти ознаки втоми за мімікою обличчя ще до того, як водій втратить контроль над траєкторією руху автомобіля.

Аналіз показує, що безконтактні відеосистеми є найбільш перспективним напрямком, оскільки вони дозволяють виявляти ознаки втоми превентивно – ще до того, як водій втратить контроль над траєкторією руху автомобіля [3].

1.3 Огляд комерційних систем моніторингу водія (DMS)

Для повноти аналізу предметної галузі доцільно розглянути існуючі комерційні рішення у сфері DMS, які вже інтегруються в сучасні автомобілі провідними автоконцернами. Ринок систем моніторингу стрімко розвивається, і сьогодні можна виділити кілька лідируючих платформ:

– система Seeing Machines (Guardian). Це одне з найбільш передових рішень для комерційного транспорту (вантажівок, автобусів). Система використовує інфрачервону (IR) камеру, встановлену на приладовій панелі. Інфрачервоне підсвічування дозволяє системі працювати в умовах повної темряви та сканувати обличчя навіть через сонцезахисні окуляри. Алгоритми відстежують рух зіниць (Gaze Tracking) та мікрорухи голови.

Головним недоліком є висока вартість та орієнтованість на корпоративний сектор [3];

- системи на базі Tobii Eye Tracking. Компанія спеціалізується на високоточному трекінгу очей для преміального сегмента автомобілів. Система створює 3D-модель голови водія та розраховує вектор погляду з точністю до градуса, що дозволяє виявляти «ефект тунельного зору». Недоліком є необхідність використання спеціалізованих стереокамер та потужних процесорів обробки;

- Tesla Cabin Camera (Tesla Autopilot). Використовує стандартну RGB-камеру. Система інтегрована з автопілотом і базується на згорткових нейронних мережах (CNN), які обробляються фірмовим процесором Tesla.

Аналіз комерційних рішень показує, що для масового впровадження у вигляді програмного забезпечення для бортових ПК бюджетного класу необхідні більш оптимізовані алгоритми, які не потребують спеціалізованих апаратних прискорювачів.

1.4 Аналіз методів розпізнавання обличчя та виділення ознак

Основою будь-якої відеосистеми моніторингу водія є алгоритм локалізації обличчя та виділення ключових мімічних точок. Вибір правильного математичного апарату є критичним для забезпечення роботи системи в режимі реального часу (Real-Time). Існують три основні підходи:

- метод Віюлі-Джонса (Каскади Хаара). Базується на використанні прямокутних ознак Хаара та алгоритмі машинного навчання AdaBoost. Переваги: надзвичайно висока швидкість роботи. Недоліки: чутливість до освітлення, велика кількість хибнопозитивних спрацювань при повороті голови, а також нездатність виділяти конкретні антропометричні точки: куточки очей, губ, що унеможливорює розрахунок метрик EAR чи MAR;

- глибинне навчання (Deep Learning). Використання багатопарових згорткових нейронних мереж (наприклад, Google MediaPipe, YOLO,

ResNet). Переваги: найвища точність та стійкість до будь-яких умов перекриття. Недоліки: висока ресурсомісткість. Для роботи з частотою 30 FPS необхідна наявність графічного прискорювача (GPU). Запуск складних CNN на звичайному центральному процесорі (CPU) призводить до падіння швидкодії до 5–10 FPS, що є неприпустимим для системи превентивної безпеки [4];

– гістограми спрямованих градієнтів (HOG) та Ансамбль дерев регресії. На першому етапі для кожного блоку пікселів розраховується гістограма градієнтів, яка передається в лінійний класифікатор опорних векторів (Linear SVM). На другому етапі застосовується метод ансамблю дерев регресії, який ітеративно «підганяє» шаблон із 68 точок під знайдене обличчя [5]. Переваги: алгоритм чудово оптимізований під архітектуру x86/x64 і здатний обробляти відеопотік виключно силами звичайного CPU з високою точністю.

1.5 Обґрунтування вибору технологій комп'ютерного зору

Незважаючи на доведену ефективність методів комп'ютерного зору на базі глибинного навчання (Deep Learning), їх реалізація у вбудованих системах автомобіля стикається з жорсткими апаратними обмеженнями, оскільки вони вимагають наявності спеціалізованих тензорних процесорів.

Для реалізації даної дипломної роботи було обрано альтернативний, апаратно-оптимізований стек технологій на базі мови програмування Python та відкритої бібліотеки комп'ютерного зору OpenCV [2]. Базовим алгоритмом детекції обличчя та локалізації ключових точок (Facial Landmarks) обрано метод ансамблю дерев регресії (Ensemble of Regression Trees), запропонований В. Каземі та Дж. Салліваном [5]. Даний алгоритм реалізований у кросплатформній бібліотеці Dlib. Особливістю цього методу є використання каскаду регресорів, які ітеративно уточнюють позиції 68

антропометричних точок обличчя на основі аналізу піксельних інтенсивностей (рисунок 1.1).

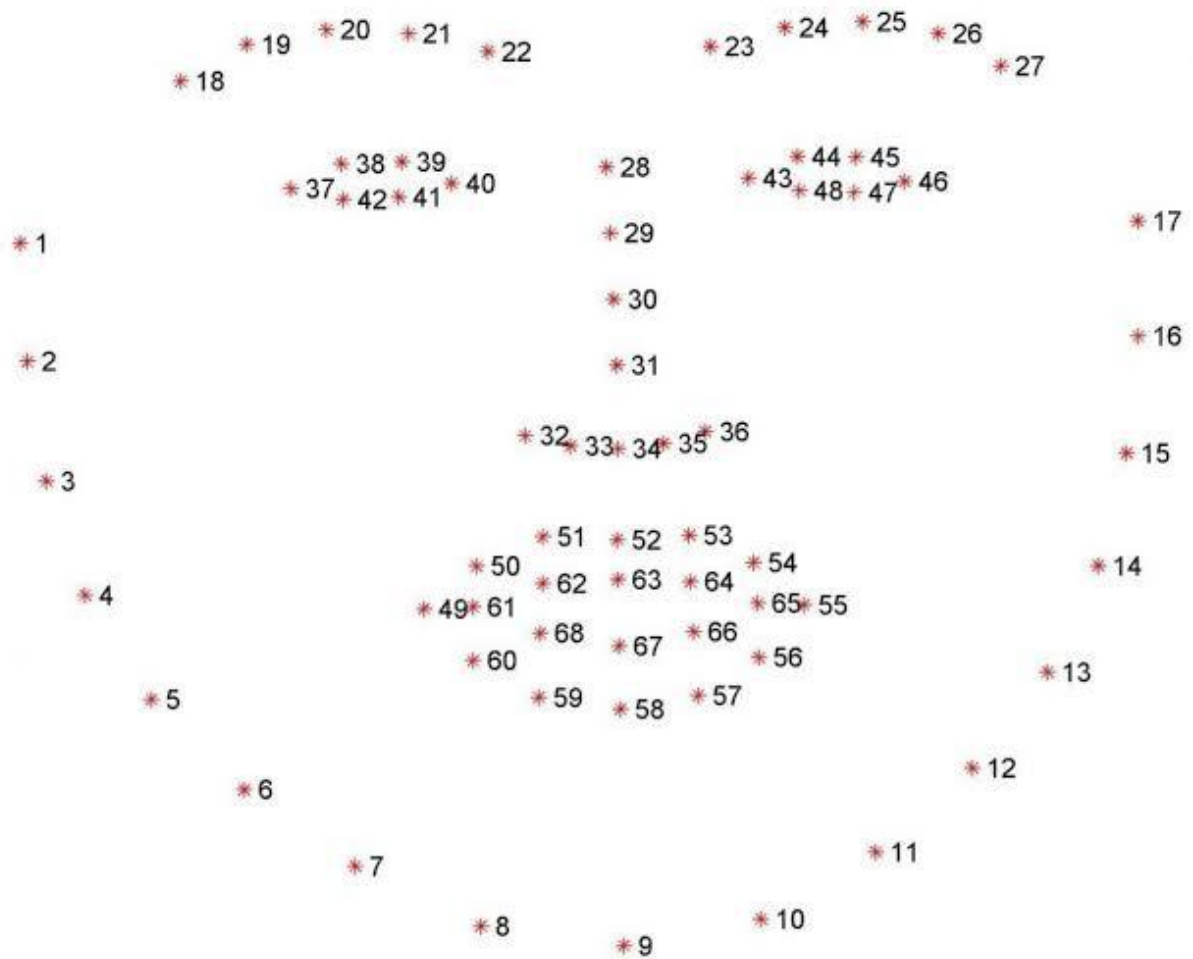


Рисунок 1.1 – Розташування 68 антропометричних точок обличчя за стандартом iBUG 300-W

Цей підхід дозволяє локалізувати риси обличчя (очі, ніс, контури губ) з мілісекундною затримкою, використовуючи виключно обчислювальні потужності стандартного центрального процесора (CPU).

Крім того, для усунення недоліків стандартних оптичних систем, які зазвичай здатні фіксувати лише геометрію обличчя, у роботі прийнято рішення інтегрувати алгоритми дистанційної фотоплетизмографії (rPPG). Це дозволить розширити функціонал системи шляхом спектральної обробки

мікрозмін кольору шкіри, викликаних серцевим ритмом водія, за допомогою математичного апарату швидкого перетворення Фур'є.

1.6 Постановка задачі

На основі проведеного аналізу предметної галузі та виявлених недоліків існуючих рішень, загальною метою дипломної роботи є розробка інтелектуальної системи безконтактного моніторингу водія, що поєднує геометричний аналіз міміки та спектральний аналіз біосигналів.

Для досягнення поставленої мети необхідно вирішити такі завдання:

- дослідити методи комп'ютерного зору та спроектувати обчислювальний конвеєр дистанційної фотоплетизмографії (rPPG) для безконтактного розрахунку частоти серцевих скорочень;
- розробити математичні моделі та алгоритми обчислення коефіцієнтів відкритості очей і рота для ідентифікації станів сну та позіхання;
- розробити алгоритми часової фільтрації (таймінгу) для відсіювання природного мікрокліпання, а також алгоритм оцінки частоти кліпання у часових вікнах для визначення станів когнітивного відхилення: стрес, інтоксикація;
- програмно імплементувати модуль захоплення та попередньої обробки відеопотоку, інваріантний до змін освітлення, та реалізувати механізм стабільного трекінгу 68 антропометричних точок обличчя в режимі реального часу;
- розробити графічний інтерфейс користувача з інтуїтивно зрозумілою системою попереджень (візуальних алертів) та провести експериментальне тестування продуктивності й точності розробленого програмного забезпечення.

2 ФОРМУВАННЯ ВИМОГ ДО ПРОГРАМНОЇ СИСТЕМИ

2.1 Мета проєкту

Головною метою проєкту є дослідження, проектування та програмна реалізація інтелектуальної відеосистеми, здатної в режимі реального часу відстежувати психофізіологічний стан водія без використання контактних датчиків. Розроблюване програмне забезпечення орієнтоване на превентивне (випереджальне) виявлення деструктивних фізіологічних станів людини за кермом, які безпосередньо загрожують безпеці дорожнього руху [3].

Програмна система має забезпечувати автоматичне виявлення ознак втоми, мікросну, тривалого відволікання уваги від дорожньої обстановки, а також непрямих біометричних ознак гострого фізіологічного стресу чи когнітивного гальмування на основі комп'ютерного аналізу міміки та капілярного кровотоку обличчя [6]. Кінцевим результатом проєкту є створення надійного, апаратно оптимізованого прототипу програмного комплексу. Цей комплекс, шляхом миттєвої генерації візуальних і текстових алертів, дозволить вчасно інформувати водія або оператора системи про критичне зниження концентрації, що безпосередньо сприятиме мінімізації ризиків виникнення дорожньо-транспортних пригод [1].

2.2 Припущення та залежності

Процес архітектурного проектування, алгоритмізації та подальшої експлуатації розроблюваної системи DMS (Driver Monitoring System) базується на визначеному переліку експлуатаційних припущень та системних залежностей.

У ході розробки було прийнято такі припущення:

- відеокамера жорстко зафіксована перед водієм (наприклад, на рульовій колонці, приладовій панелі або дзеркалі заднього виду) таким чином, щоб обличчя користувача знаходилося переважно в положенні анфас (допускаються незначні динамічні повороти голови під час керування автомобілем);

- у салоні транспортного засобу забезпечується мінімально необхідний рівень освітленості (природного денного, штучного або інфрачервоного підсвічування в темну пору доби), який є достатнім для коректної фіксації градієнтів яскравості обличчя оптичним сенсором [4];

- водій не використовує масивні сонцезахисні окуляри з дзеркальним або повним ультрафіолетовим покриттям, медичні маски чи інші предмети, що повністю перекривають ключові антропометричні зони обличчя (орбітальні області та контур губ) або повністю блокують видимий спектр світла.

Ефективне функціонування програмного комплексу визначається такими залежностями:

- програмні залежності: працездатність аналітичного ядра системи критично залежить від доступності локального файлу переднавченої моделі регресії `shape_predictor_68_face_landmarks.dat` [5], а також від коректної ініціалізації сторонніх бібліотек комп'ютерного зору та математичної обробки сигналів (OpenCV, Dlib, NumPy, SciPy) [2];

- апаратні залежності: точність і стабільність роботи модуля дистанційної фотоплетизмографії (rPPG) для безконтактного вимірювання пульсу прямо залежить від здатності відеокамери забезпечувати фіксовану частоту дискретизації кадрів на рівні не менше 25–30 кадрів на секунду без пропусків фреймів;

- обчислювальні залежності: швидкість обробки кожного окремого кадру конвеєра (включаючи каскадний аналіз та швидке перетворення Фур'є) залежить від наявності центрального обчислювального процесора

рівня не нижче середнього, здатного виконувати векторизовані матричні операції без критичних затримок.

2.3 Функціональні вимоги

Функціональні вимоги формалізують поведінку програмної системи та визначають чіткий перелік операцій, які вона повинна виконувати в автоматичному режимі під час обробки вхідного медіапотoku.

Розроблюваний програмний комплекс повинен забезпечувати виконання таких функцій:

- захоплення та препроцесинг відеопотоку: безперервне отримання поточних кадрів із підключеної цифрової камери, динамічне визначення параметрів матриці кадру, масштабування та оптимізоване перетворення зображення у градації сірого (grayscale) для первинного детектора;

- локалізація та трекінг обличчя: стабільне виявлення геометричних меж обличчя в кадрі та високоточне виділення двовимірних координат 68 антропометричних ключових точок (landmarks) згідно із затвердженим індексаційним стандартом [5];

- обчислення метрик міміки: безперервний математичний розрахунок скалярних коефіцієнтів відкритості очей (EAR – Eye Aspect Ratio) та розкриття рота (MAR – Mouth Aspect Ratio) для кожного кадру обчислювального циклу;

- часова фільтрація та таймінг станів: інтеграція програмних таймерів для точного вимірювання тривалості падіння коефіцієнта EAR нижче критичного порогу, що необхідно для диференціації рефлекторного моргання від фази мікросну;

- дистанційна пульсометрія (rPPG): автоматичне виділення області інтересу (ROI) в зоні лоба, формування часового ряду коливань інтенсивності зеленого колірної каналу, його фільтрація, нормалізація та

спектральний аналіз за допомогою швидкого перетворення Фур'є для розрахунку частоти серцевих скорочень водія;

- статистичний аналіз частоти кліпання: накопичення та циклічний аналіз даних про кількість завершених моргань у межах фіксованого 30-секундного ковзного часового вікна;

- контроль просторової орієнтації: фіксація критичного зміщення ключових точок носа відносно центральної осі симетрії обличчя для детекції тривалого відволікання погляду від дороги;

- багаторівневе сповіщення: обробка поточних станів за допомогою логічного автомата та виведення на екран відповідної кольорової індикації та текстових попереджень.

2.4 Нефункціональні вимоги

Нефункціональні вимоги визначають критерії якості, атрибути надійності, обмеження продуктивності та вимоги до безпеки, яким повинен відповідати розроблюваний програмний продукт.

До системи висуваються такі нефункціональні вимоги:

- швидкодія та час реакції: система повинна функціонувати в режимі жорсткого реального часу, забезпечуючи швидкість обробки відеопотоку на рівні не менше 20–25 FPS на стандартному CPU, а час від моменту фіксації небезпечного стану (наприклад, мікросну) до моменту активації тривожного алерта не повинен перевищувати 100 мілісекунд;

- надійність та відмовостійкість: програмне забезпечення має демонструвати стійкість до короткочасних шумів відеопотоку або перекриття обличчя (наприклад, під час маневрування чи почухування обличчя). У разі тимчасової втрати обличчя з кадру система повинна автоматично відновлювати трекінг протягом не більше ніж 3 кадрів після його повторної появи;

- конфіденційність біометричних даних: оскільки система обробляє персональні біометричні параметри людини (координати точок обличчя, частоту серцевих скорочень), всі обчислення мають виконуватися виключно локально (Edge Computing) без передачі медіаданих у мережу Інтернет, що унеможливорює витік конфіденційної інформації;
- портативність та сумісність: архітектура коду на базі мови Python має бути кросплатформною та забезпечувати можливість запуску системи як у середовищі ОС Windows, так і в Unix-подібних операційних системах.

2.5 Вимоги до інтерфейсу користувача

Людино-машинний інтерфейс системи моніторингу орієнтований на роботу в умовах високого когнітивного навантаження, тому до нього висуваються особливі вимоги щодо ергономіки та швидкості сприйняття інформації:

- принцип HUD (Hears-Up Display): графічні елементи, метрики та рамки детекції повинні накладатися безпосередньо поверх оригінального кольорового відеопотоку з камери без створення додаткових модальних вікон, які перевантажують екран;
- чітке кольорове кодування: візуальна індикація станів системи повинна однозначно інтерпретуватися водієм за допомогою стандартної палітри кольорів (зелений – нормальний стан, помаранчевий – відволікання від дороги, жовтий – стрес, червоний – критична небезпека засинання);
- візуальне дублювання алертів: кожне критичне фізіологічне відхилення повинно супроводжуватися одночасним виведенням великого контрастного текстового сповіщення та динамічною зміною графічної маски (наприклад, напівпрозорою фіолетовою заливкою кадру у разі детекції признаков інтоксикації). Текст повинен залишатися читабельним при будь-якому колірному фоні відеозображення.

3 АРХІТЕКТУРА ТА ПРОЕКТУВАННЯ СИСТЕМИ

3.1 UML проектування ПЗ

Для формалізації вимог, визначення логічних зв'язків та візуалізації архітектури розроблюваного програмного забезпечення було використано уніфіковану мову моделювання UML (Unified Modeling Language). Проектування проводилося на рівні об'єктно-орієнтованої архітектури та поведінкових алгоритмів.

3.1.1 Діаграма варіантів використання (Use Case Diagram)

Діаграма прецедентів визначає функціональне призначення системи з точки зору зовнішніх суб'єктів взаємодії. У системі виділено двох основних акторів: «Водій» (користувач, фізіологічний стан якого безперервно аналізується) та «Камера» (апаратний оптичний сенсор, що виступає ініціатором потоку даних). Головними прецедентами є процеси безперервного розрахунку біометричних показників та генерація багаторівневих попереджувальних сигналів у разі фіксації критичних відхилень від норми (рисунок 3.1).

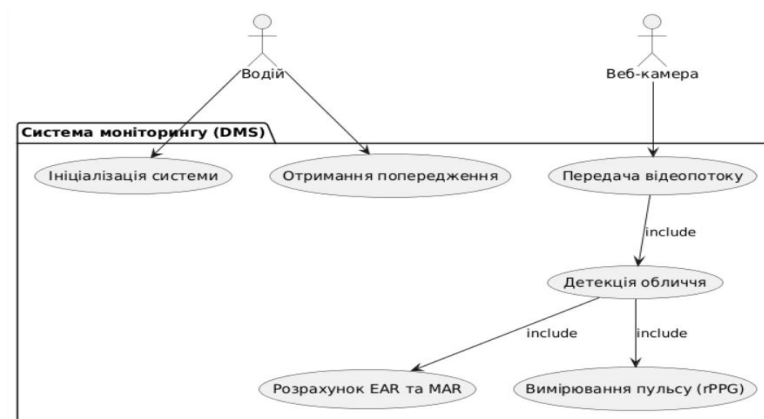


Рисунок 3.1 – Use Case діаграма

3.1.2 Діаграма класів (Class Diagram)

Логічна архітектура програмного комплексу спроектована з використанням патернів об'єктно-орієнтованого підходу. Діаграма класів (рисунок 3.2) демонструє статичну структуру системи, що розділена на вузькоспеціалізовані модулі. Основний клас `DriverMonitorApp` виконує роль головного контролера (Controller), який ініціалізує та керує екземплярами класів `FaceDetector` (відповідає за локалізацію за допомогою `Dlib`), `MetricsCalculator` (математичний апарат обчислення коефіцієнтів) та `AlertSystem` (логіка перемикання станів та генерації сповіщень).

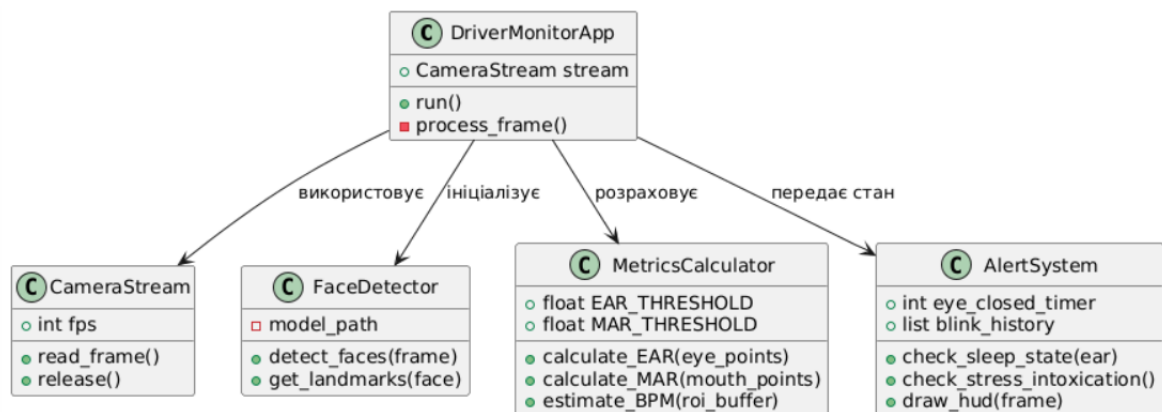


Рисунок 3.2 – Діаграма класів

3.1.3 Діаграма діяльності (Activity Diagram)

Для опису динаміки роботи головного обчислювального конвеєра системи побудовано діаграму діяльності (рисунок 3.3). Вона деталізує покроковий алгоритм обробки кожного окремого кадру відеопотоку. Логіка передбачає циклічне виконання операцій захоплення, препроцесингу шляхом переведення у градації сірого, локалізації антропометричних точок та перевірки розрахованих метрик (EAR, MAR, BPM) на відповідність безпечним пороговим значенням.

3.1.4 Діаграма послідовностей (Sequence Diagram)

Діаграма послідовностей (рисунок 3.4) відображає взаємодію об'єктів у часі для одного повного циклу аналізу стану водія. Вона показує, як ініціатор CameraStream передає кадр до DriverMonitorApp, який послідовно викликає методи FaceDetector для отримання координат точок. Далі ці точки передаються до MetricsCalculator, який повертає значення EAR, MAR та пульсу. На фінальному етапі AlertSystem приймає ці дані, перевіряє таймери та повертає статус для відмальовування на екрані.

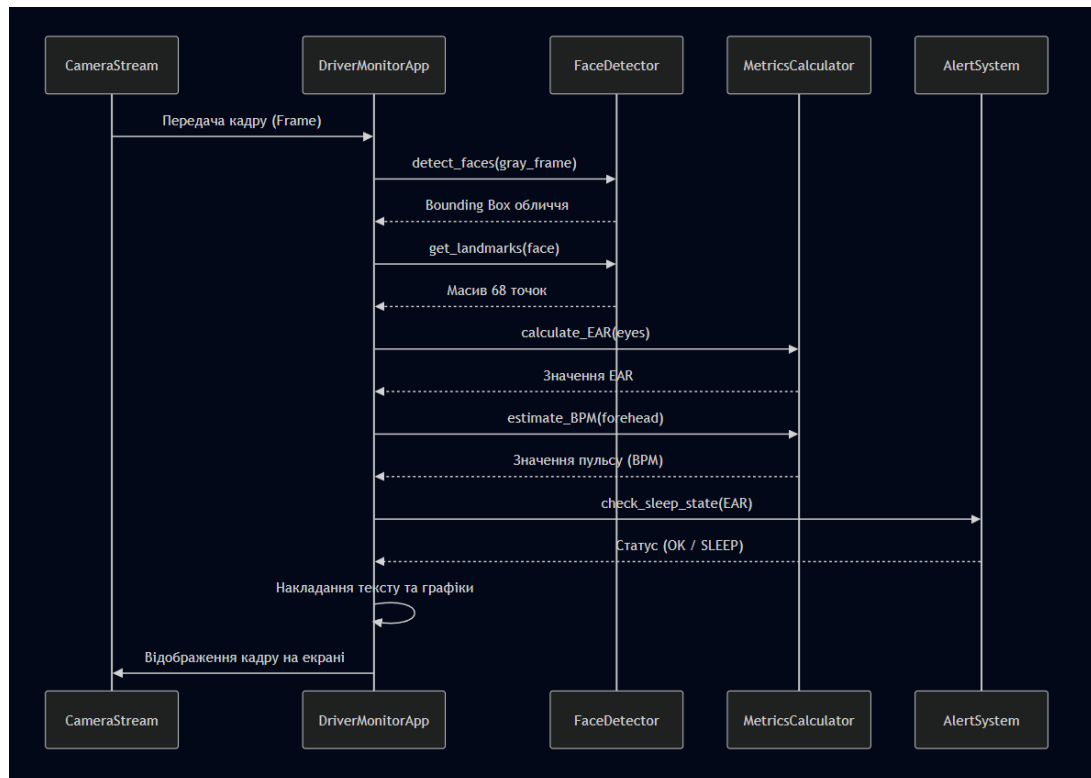


Рисунок 3.4 – Діаграма послідовностей системи моніторингу

3.2 Архітектурна схема обчислювального конвеєра та математична модель локалізації обличчя

Система побудована за модульним принципом з використанням парадигми поділу відповідальності. Це дозволяє оптимізувати

обчислювальні ресурси та послідовно обробляти потік даних з одного джерела відеосигналу без використання ресурсомісткої багатопотоковості, що є критичною вимогою для вбудованих пристроїв з обмеженими апаратними характеристиками (наприклад, бортових комп'ютерів автомобіля). Основні етапи обробки даних утворюють єдиний конвеєр і включають:

- модуль захоплення та препроцесингу: отримання кадру з камери через API бібліотеки OpenCV [2], його білінійне масштабування для зменшення обчислювального навантаження та конвертація у градації сірого. Перетворення у відтінки сірого є обов'язковою вимогою для коректної роботи детектора на базі гістограм спрямованих градієнтів (HOG – Histogram of Oriented Gradients);

- модуль детекції та локалізації (Facial Landmark Detection): ідентифікація контурів обличчя у кадрі за допомогою методу опорних векторів (SVM) та застосування математичного апарату ансамблю дерев регресії (Ensemble of Regression Trees) [5] для виділення 68 антропометричних ключових точок згідно зі стандартом датасету iBUG 300-W;

- аналітичний модуль: розрахунок геометричних (евклідові відстані між точками очей та рота) і частотних (аналіз зеленого спектра відеокадру) характеристик для отримання показників EAR, MAR та BPM;

- інтерфейсний модуль (HMI): візуалізація метрик на оригінальному кольоровому відеопотоці, накладання геометричних масок стану та генерація попереджувальних сигналів.

В основі роботи модуля FaceDetector (бібліотека Dlib) лежить алгоритм ітеративного уточнення форми, запропонований В. Каземі [5]. Замість повного сканування зображення важкими нейромережами, алгоритм використовує каскад дерев регресії. Нехай S_t – поточна оцінка координат 68 точок обличчя на кроці t . Наступна, більш точна позиція точок S_{t+1} обчислюється за рекурентною формулою:

$$S_{t+1} = S_t + r_t(I, S_t), \quad (3.1)$$

де I – інтенсивність пікселів вхідного зображення;

r_t – вектор оновлення, який генерується деревом регресії на кроці t .

Такий підхід дозволяє досягти стабільної локалізації точок за 1–3 мілісекунди, що робить його ідеальним для систем реального часу.

3.3 Математична модель оцінки стану очей (EAR)

Для метричної ідентифікації процесу закриття очей використовується алгоритм обчислення коефіцієнта Eye Aspect Ratio (EAR), адаптований для задач моніторингу втомі [4], [7]. Головна математична перевага цього методу полягає в його повній інваріантності до масштабу обличчя (відстані водія до камери) та незначних поворотів голови, оскільки метрика базується на відносних пропорціях, а не на абсолютних значеннях у пікселях.

Згідно з топологією карти 68 ключових точок, праве око описується точками з індексами 36...41, а ліве – 42...47. Для кожного ока розраховується відношення суми двох вертикальних відстаней між верхньою та нижньою повіками до горизонтальної відстані між куточками ока за формулою:

$$EAR = \frac{\|p_2 - p_6\| + \|p_3 - p_5\|}{2 \cdot \|p_1 - p_4\|}, \quad (3.2)$$

де $p_1, \dots, p_6$ – двовимірні координати точок відповідного ока;

$\|p_i - p_j\|$ – евклідова відстань між точками i та j , яка обчислюється як $\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$.

За результатами емпіричних спостережень:

- при відкритих очах значення EAR є відносно стабільним (близько 0.25...0.35);
- при закритті очей (під час кліпання або сну) значення стрімко наближається до нуля (падає нижче 0.15).

Для нівелювання природної асиметрії обличчя розрахунок ведеться як середнє арифметичне EAR для обох очей одночасно: $EAR_{avg} = (EAR_{left} + EAR_{right}) / 2$. Щоб уникнути хибних спрацювань на звичайне рефлекторне кліпання (яке триває 100...400 мілісекунд), система вмикає таймер небезпеки лише тоді, коли поточний EAR_{avg} падає нижче експериментально встановленого порогу ($Th_{ear} = 0.23$) і залишається в такому стані протягом критичного часу $\Delta T \geq 1.5$ сек.

3.4 Математична модель детекції позіхання (MAR)

Аналогічно до аналізу очей, для ротової порожнини проектується розрахунок коефіцієнта Mouth Aspect Ratio (MAR). Для мінімізації впливу стандартної міміки (наприклад, усмішки або розмови) алгоритм використовує координати виключно внутрішнього контуру губ (точки з індексами 61...68).

Формула розрахунку базується на трьох вертикальних відстанях між верхньою та нижньою губою в центрі та по краях, а також одній горизонтальній відстані між куточками рота:

$$MAR = \frac{\|p_{62} - p_{68}\| + \|p_{63} - p_{67}\| + \|p_{64} - p_{66}\|}{3 \cdot \|p_{61} - p_{65}\|}, \quad (3.3)$$

де $\|p_i - p_j\|$ – евклідова відстань між точками.

Коли водій глибоко позіхає, вертикальна відстань між щелепами різко зростає, що призводить до аномального перевищення встановленого порогу. В рамках розроблюваної системи критичним порогом позіхання визначено

значення $MAR > 0.5$. Повторюване перевищення цього порогу алгоритмом розцінюється як ранній біометричний індикатор настання втоми та кисневого голодування мозку, що передує проявам мікросну [6].

3.5 Алгоритм дистанційного вимірювання пульсу (rPPG)

Проектування модуля дистанційної фотоплетизмографії (remote Photoplethysmography) базується на виявленні та аналізі мікроскопічних варіацій інтенсивності зеленого колірного каналу відеозображення. Вибір саме зеленого спектра світла (довжина хвилі $\lambda \approx 500 \dots 550$ нм) математично та біологічно обґрунтований його найвищим коефіцієнтом поглинання оксигемоглобіном (HbO_2) крові у підшкірних капілярах під час систоли – фази скорочення серцевого м'яза.

Програмний конвеєр спектральної обробки біосигналу складається з наступних математичних кроків:

- виділення області інтересу (ROI): алгоритм динамічно розраховує координати прямокутної зони на лобі водія (область вище брів). Ця ділянка обрана через густу капілярну сітку та мінімальну схильність до впливу мимічних м'язів;
- формування одновимірного часового ряду: для кожного відеокадру t обчислюється середнє значення інтенсивності пікселів зеленого каналу I_g у заданій зоні лоба (ROI):

$$S(t) = \frac{1}{N} \sum_{i=1}^N I_g(x_i, y_i, t), \quad (3.4)$$

де N – загальна кількість пікселів у зоні лоба;

I_g – інтенсивність зеленого кольору.

Дані накопичуються у динамічному кільцевому буфері ємністю 150 кадрів (що відповідає 5 секундам запису при частоті 30 FPS).

Після формування часового ряду йде нормалізація та детрендинг. Для усунення впливу фонового освітлення (адитивного шуму) отриманий сигнал $S(t)$ стандартизується. Від кожного значення віднімається математичне сподівання μ і результат ділиться на середньоквадратичне відхилення σ :

$$\hat{S}(t) = \frac{S(t) - \mu}{\sigma}, \quad (3.5)$$

де $S(t)$ – отриманий сигнал;

μ – математичне сподівання;

σ – середньоквадратичне відхилення.

Наступним кроком обробки є швидке перетворення Фур'є (FFT). Часовий сигнал переводиться у частотну область для пошуку гармонічних коливань. Дискретне перетворення Фур'є обчислюється за формулою:

$$X(k) = \sum_{n=0}^{L-1} \hat{S}(n) \cdot e^{-j\frac{2\pi}{L}kn}, \quad (3.6)$$

де $L=150$ – розмір вікна;

k – індекс частотної складової.

На фінальному етапі застосовується смугова фільтрація. Отриманий спектр програмно очищається шляхом обнулення амплітуд частот, які виходять за межі діапазону $[0.75, 2.5]$ Гц. Цей діапазон відповідає фізіологічно можливій частоті серцевих скорочень людини у стані від спокою до сильного стресу – приблизно 45...150 ударів на хвилину. Частота $f_{\max}$, що відповідає максимальному піку амплітуди у відфільтрованому спектрі, конвертується у підсумкове значення пульсу: $\text{BPM} = f_{\max} \times 60$.

3.6 Логіка статистичного аналізу частоти кліпання (Blink Rate Analysis)

Для детекції непрямих ознак когнітивного гальмування (сп'яніння) або сильного психоемоційного стресу спроектовано модуль довгострокового статистичного аналізу динаміки повік. Замість миттєвої реакції на окремий кадр, система застосовує алгоритм ковзного часового вікна.

Система реєструє кожен факт завершеного кліпання (подія повного циклу: $EAR < Th_{ear} \rightarrow EAR > Th_{ear}$) та зберігає його часову мітку (timestamp). Алгоритм безперервно підраховує кількість подій C_{blink} в інтервалі $[W = t_{current} - \Delta t, t_{current}]$, де $\Delta t = 30$ секунд:

- низька частота ($C_{blink} < 4$): інтерпретується як «застиглий погляд» або критична депресія центральної нервової системи, що є класичною медичною ознакою алкогольного сп'яніння або сильної когнітивної втоми;
- висока частота ($C_{blink} > 15$): інтерпретується як тахікінезія повік, що вказує на стан гострого фізіологічного стресу, паніки або сильного зорового подразнення.

3.7 Кінцевий автомат прийняття рішень (Decision Logic)

Оскільки під час комплексного моніторингу система може одночасно фіксувати декілька відхилень, архітектура програмного забезпечення передбачає впровадження логічного апарату кінцевого автомата для пріоритезації тривоги.

Математично кінцевий автомат системи можна описати як кортеж $M = (S, \Sigma, \delta, s_0)$, де:

- $S = \{\text{NORMAL, YAWNING, DISTRACTED, STRESS, INTOXICATED, SLEEP_DANGER}\}$ – скінченна множина можливих станів системи;

– Σ – множина вхідних подій (значення EAR, MAR, таймерів та BPM);

– $s_0 = \text{NORMAL}$ – початковий стан;

– δ – функція переходів, яка строго пріоритезована.

Система послідовно перевіряє умови та присвоює глобальний стан небезпеки за строгим спаданням пріоритету (від найвищого до найнижчого):

– SLEEP DANGER (критичний): спрацьовує негайно при закритих очах понад 1.5 сек. Перекриває будь-які інші сповіщення (візуалізація: червоний екран);

– INTOXICATION (високий): детекція відсутності моргання у 30-секундному вікні (візуалізація: фіолетова маска);

– DISTRACTION (середній): критичне відхилення вектора погляду або повороту голови від центру осі дороги;

– STRESS WARNING (середній): комбінація тахікінезії (висока частота кліпання) та підвищеного пульсу;

– YAWNING (низький): фіксація позіхання ($\text{MAR} > 0.5$).

Така жорстка ієрархічна структура переходів станів δ гарантує, що у випадку виникнення множинних загроз найбільш критична інформація для збереження життя буде донесена до водія першочергово, унеможливлюючи перевантаження графічного інтерфейсу.

4 ОПИС ПРОГРАМНОЇ РЕАЛІЗАЦІЇ

4.1 Вибір та обґрунтування інструментарію розробки

Для практичної реалізації розроблених алгоритмів системи моніторингу було обрано високорівневу об'єктно-орієнтовану мову програмування Python версії 3.11. Вибір цієї мови зумовлений її кросплатформенністю, високою швидкістю прототипування та наявністю найпотужнішої екосистеми бібліотек для задач комп'ютерного зору і машинного навчання. Оскільки Python працює як інтерпретатор, для досягнення високої швидкодії в режимі реального часу інтенсивні математичні обчислення делеговано C/C++ бекендам використовуваних бібліотек.

Програмний комплекс базується на таких ключових компонентах:

- OpenCV: відкрита бібліотека комп'ютерного зору, яка забезпечує низькорівневу роботу з апаратними відеозасобами (камерою), перетворення кольірних просторів (BGR у Grayscale), а також швидкий рендеринг графічних примітивів для інтерфейсу користувача [2];

- Dlib: спеціалізована C++ бібліотека машинного навчання, що через Python-інтерфейс надає доступ до оптимізованої переднавченої моделі регресора ключових точок обличчя (shape_predictor_68_face_landmarks.dat). Використання ансамблю регресійних дерев дозволяє системі локалізувати 68 точок за 1–3 мілісекунди [5];

- NumPy: фундаментальна бібліотека для наукових обчислень, яка використовується для векторизованих математичних операцій з матрицями пікселів відеокادру та швидкого розрахунку евклідових відстаней між точками (модуль `np.linalg.norm`);

- SciPy / NumPy FFT: модуль, що програмно реалізує математичний апарат дискретного швидкого перетворення Фур'є (Fast Fourier Transform), необхідний для аналізу частотних характеристик серцевого ритму.

Фрагмент програмного коду з імпортуванням необхідних залежностей та ініціалізацією базових параметрів наведено на рисунок 4.1. Повний код розробленого програмного комплексу представлено у додатку А.

```
import cv2
import dlib
import numpy as np
import time
```

Рисунок 4.1 – Фрагмент коду з імпортуванням бібліотек

4.2 Структура програмного комплексу та основні модулі

Програмне забезпечення спроектоване як єдиний безперервний обчислювальний цикл з чітким логічним поділом на функціональні блоки. Така архітектура унеможливорює стан гонки даних і не потребує складної синхронізації потоків.

Основні етапи роботи головного циклу включають:

- ініціалізація та завантаження моделей: перед початком захоплення відео програма ініціалізує HOG-детектор обличчя та завантажує файл регресора `shape_predictor`. З метою підвищення надійності програмного забезпечення (Robustness), процес завантаження обгорнуто в блок обробки винятків `try-except`. У випадку відсутності файлу моделі в директорії проекту, система інформує користувача про критичну помилку і коректно завершує роботу, уникаючи аварійного збою (рисунок 4.2);

```
# Initialize dlib
detector = dlib.get_frontal_face_detector()
try:
    predictor = dlib.shape_predictor("shape_predictor_68_face_landmarks.dat")
except:
    print("Error: .dat model file not found!")
    exit()
```

Рисунок 4.2 – Фрагмент коду з ініціалізацією та завантаженням моделей

– обробка вхідного відеопотоку: захоплення потокових кадрів здійснюється за допомогою створення екземпляра об'єкта `cv2.VideoCapture(0)`. Для оптимізації роботи детектора Dlib, який не потребує інформації про колір для локалізації контурів, кожен захоплений кольоровий кадр конвертується у градації сірого за допомогою методу `cv2.cvtColor`;

– модуль розрахунку метрик очей: для уникнення дублювання коду програмно реалізовано окрему функцію `get_ear(eye_points)`, яка приймає масив із 6 двовимірних координат (точок конкретного ока), виконує векторизований розрахунок відстаней через `np.linalg.norm` та повертає скалярне значення ступеня відкритості ока (рисунок 4.3).

```
def get_ear(eye_points):
    v1 = np.linalg.norm(eye_points[1] - eye_points[5])
    v2 = np.linalg.norm(eye_points[2] - eye_points[4])
    h = np.linalg.norm(eye_points[0] - eye_points[3])
    return (v1 + v2) / (2.0 * h)
```

Рисунок 4.3 – Фрагмент коду з реалізацією функції `get_ear`

4.3 Алгоритмічна реалізація функціональних модулів аналізу

4.3.1 Аналіз стану очей та лічильник моргань (Eye Analysis & Blink Counter)

Даний метод забезпечує первинний безперервний моніторинг динаміки повік для визначення поточної фізіологічної активності водія. Програма викликає функцію `get_ear` для зрізів масиву точок `pts[36:42]` – праве око та `pts[42:48]` – ліве око, після чого обчислює їхнє середнє арифметичне значення.

Для підрахунку моргань використовується логічний прапорець `is_eye_closed`. Щоразу, коли поточний `ear` падає нижче константного порогу, який становить 0.23, прапорець перемикається в стан `True`. При наступному піднятті `ear` вище порогу система фіксує завершений цикл моргання та інкрементує глобальний лічильник `BLINK_COUNT`.

4.3.2 Алгоритм часової фільтрації та таймінгів (Advanced Timer Logic)

Для мінімізації хибнопозитивних спрацювань та точної оцінки когнітивного стану водія впроваджено жорсткий механізм часової фільтрації на базі модуля `time` (рисунок 4.4). Обробка часових міток відбувається за двома основними напрямками:

- критерій сну: змінна `eye_closed_timer` ініціалізується системним часом `time.time()` при першому ж падінні `EAR` нижче порогу. Програма на кожній ітерації циклу перевіряє різницю між поточним часом і збереженим. Якщо очі залишаються закритими понад граничний ліміт `SLEEP_TIME_LIMIT` (1.5 секунди), поточний статус системи миттєво змінюється на критичний: «DANGER: SLEEP»;

- детекція довгострокових відхилень: для ідентифікації стресу та інтоксикації реалізовано ковзне 30-секундне вікно. Система аналізує накопичений `BLINK_COUNT`. Значення менше 4 кліпань програмно маркується як стан сп'яніння/депресії ЦНС, а понад 15 – як стан сильного фізіологічного стресу. Після перевірки таймер та лічильник синхронно обнуляються.

Використання такого дворівневого часового аналізу (короткострокового для критичних ситуацій та довгострокового для когнітивних відхилень) гарантує високу стійкість програмного комплексу до одиничних помилок розпізнавання точок та запобігає перевантаженню користувача хибними сигналами тривоги.

```
# 1. EYE ANALYSIS & BLINK COUNTER
ear = (get_ear(pts[36:42]) + get_ear(pts[42:48])) / 2.0

if ear < EYE_AR_THRESH:
    if not is_eye_closed:
        is_eye_closed = True
        BLINK_COUNT += 1

    if eye_closed_timer is None:
        eye_closed_timer = time.time()

    if time.time() - eye_closed_timer >= SLEEP_TIME_LIMIT:
        status = "!!! DANGER: SLEEP !!!"
        color = (0, 0, 255)
else:
    is_eye_closed = False
    eye_closed_timer = None
```

Рисунок 4.4 – Фрагмент коду з реалізацією моніторингу стану очей для визначення поточної активності водія

4.3.3 Детекція відволікання та позіхання (Distraction & Yawning Detection)

Цей логічний блок (рисунок 4.5) відповідає за моніторинг просторової концентрації уваги та проявів фізичної втоми:

- відволікання (Distraction): реалізовано евристичний підхід через аналіз координати кінчика носа (pts[30][0]). Алгоритм обчислює абсолютну різницю між позицією носа та геометричним центром знайденого обличчя. При девіації понад 20 пікселів система класифікує це як поворот голови та активує статус «LOOK AT THE ROAD» [4];

- позіхання (Yawning): на основі математичної моделі MAR обчислюється відстань між верхньою та нижньою губою. При перевищенні константного порогу (> 0.5) генерується відповідне попередження, що сигналізує про ранню стадію втрати пильності.

```
# 3. DISTRACTION & YAWNING (Simplified)
nose_x = pts[30][0]
if abs(nose_x - rect.center().x) > 20:
    status = "WARNING: LOOK AT THE ROAD!"
    color = (0, 165, 255)

mar = np.linalg.norm(pts[62] - pts[66]) / np.linalg.norm(pts[60] - pts[64])
if mar > MOUTH_AR_THRESH and status == "STATUS: OK":
    status = "YAWNING DETECTED"
    color = (255, 255, 0)
```

Рисунок 4.5 – Фрагмент коду з реалізацією моніторингу концентрації уваги та фізичної втоми

4.4 Реалізація модуля безконтактного пульсометра (rPPG)

Модуль дистанційної фотоплетизмографії (рисунок 4.6) реалізовано у вигляді окремого обчислювального блоку, який взаємодіє з глобальним списком GREEN_BUFFER. Цей масив виконує роль черги FIFO (First-In, First-Out) і має жорстко обмежену ємність у 150 елементів.

Програмний пайплайн обробки біосигналу включає наступні операції:

- динамічне виділення ROI (Region of Interest): на основі координат верхньої межі `rect.top()` та центру `rect.center().x` обличчя, алгоритм за допомогою матричних зрізів NumPy (`frame[y1:y2, x1:x2]`) вирізає прямокутну область лоба. Це виключає потрапляння в кадр волосся або очей;

- нормалізація сигналу: перед подачею на алгоритм Фур'є дані масиву нормалізуються за формулою $(data - \text{np.mean}(data)) / \text{np.std}(data)$. Додавання малої константи $1e-6$ у знаменник гарантує захист від помилки ділення на нуль (`DivisionByZeroError`) у випадку ідеально однотонного кадру;

- FFT-аналіз: функція `np.fft.rfft` переводить часовий ряд інтенсивностей зеленого кольору в частотний спектр. Далі за допомогою

маски np.where відсікаються всі частоти поза фізіологічним діапазоном пульсу людини (0.8–2.2 Гц). Індекс максимального піку амплітуди np.argmax(fft) використовується для отримання домінуючої частоти, яка конвертується у значення BPM (Beats Per Minute).

```
# 4. HEART RATE (rPPG)
roi_y1, roi_y2 = max(0, rect.top() - 25), max(0, rect.top() - 5)
roi_x1, roi_x2 = rect.center().x - 10, rect.center().x + 10
forehead_roi = frame[roi_y1:roi_y2, roi_x1:roi_x2]

if forehead_roi.size > 0:
    GREEN_BUFFER.append(np.mean(forehead_roi[:, :, 1]))
    if len(GREEN_BUFFER) > 150:
        GREEN_BUFFER.pop(0)
        data = np.array(GREEN_BUFFER)
        data = (data - np.mean(data)) / (np.std(data) + 1e-6)
        fft = np.abs(np.fft.rfft(data))
        freqs = np.fft.rfftfreq(len(data), 1 / 30)
        idx = np.where((freqs >= 0.8) & (freqs <= 2.2))[0]
        if len(idx) > 0:
            HEART_RATE = freqs[idx[np.argmax(fft[idx])]] * 60
```

Рисунок 4.6 – Фрагмент коду з реалізацією модуля безконтактного пульсометра

4.5 Система візуалізації та інтерфейс користувача

Графічний людино-машинний інтерфейс реалізовано за допомогою функцій накладання примітивів бібліотеки OpenCV (рисунок 4.7). Даний підхід дозволяє уникнути необхідності розробки окремого важкого GUI-додатку (наприклад, на базі PyQt або Tkinter), що суттєво економить обчислювальні ресурси системи.

Виведення інформації здійснюється безпосередньо на кожен кадр відеопотоку перед його відображенням:

– геометрична та колірна індикація: за допомогою методу `cv2.rectangle` навколо обличчя малюється обмежувальна рамка. Колір рамки, якій відповідає змінна `color` динамічно змінюється залежно від поточного стану (зелений – норма, помаранчевий – увага, червоний – критична небезпека);

– інформаційна текстова панель: функція `cv2.putText` зі шрифтом `FONT_HERSHEY_SIMPLEX` використовується для виведення в лівому верхньому куті поточного статусу системи та розрахованого значення пульсу в реальному часі;

– динамічне оверлей-меню: при виявленні довгострокових критичних станів, на екран виводиться додатковий текстовий блок, який привертає увагу водія за допомогою контрастних кольорів.

```
# VISUALIZATION
cv2.rectangle(frame, (rect.left(), rect.top()), (rect.right(), rect.bottom()), color, 2)
cv2.putText(frame, status, org=(20, 45), cv2.FONT_HERSHEY_SIMPLEX, fontScale=0.9, color, thickness=3)
cv2.putText(frame, text=f"Pulse: {int(HEART_RATE)} BPM", org=(20, 80), cv2.FONT_HERSHEY_SIMPLEX, fontScale=0.7, color=(255, 255, 0), thickness=2)

if health_msg:
    cv2.putText(frame, health_msg, org=(20, 115), cv2.FONT_HERSHEY_SIMPLEX, fontScale=0.6, msg_color, thickness=2)
```

Рисунок 4.7 – Фрагмент коду з реалізацією системи візуалізації та інтерфейсу користувача

Кінцевий модифікований кадр виводиться на екран за допомогою функції `cv2.imshow`, після чого цикл повторюється до отримання команди переривання від користувача.

5 ЕКСПЕРИМЕНТАЛЬНІ ДОСЛІДЖЕННЯ ТА ТЕСТУВАННЯ СИСТЕМИ

5.1 Методика проведення експериментів та апаратне забезпечення

Для підтвердження працездатності та оцінки ефективності розробленого програмного комплексу було проведено серію експериментальних досліджень. Метою тестування була перевірка точності роботи алгоритмів комп'ютерного зору в режимі реального часу під впливом різних імітованих станів водія.

Тестування проводилося на стандартному обчислювальному пристрої без використання дискретного графічного прискорювача, що імітує обмежені апаратні можливості бортових комп'ютерів автомобіля. Відеопотік захоплювався за допомогою вбудованої веб-камери з роздільною здатністю 720p при частоті 30 кадрів на секунду. Дослідження проводилися за умов нормального денного освітлення.

5.2 Математичні метрики оцінки якості алгоритмів

Оскільки розроблена система виконує задачу бінарної та багатокласової класифікації станів (наприклад, стан мікросну може бути так або ні), для кількісної оцінки її ефективності було використано стандартний апарат матриці помилок та похідні статистичні метрики класифікатора.

Введемо наступні базові позначення для кожної події (наприклад, засинання):

- TP (True Positive): істинно позитивне спрацювання (водій заснув, і система видала тривогу);
- TN (True Negative): істинно негативне спрацювання (водій бадьорий, і система мовчить);

- FP (False Positive): хибнопозитивне спрацювання або хибна тривога (водій просто кліпнув або подивився вниз, а система зафіксувала сон);
- FN (False Negative): хибнонегативне спрацювання або пропуск головної цілі (водій заснув, але система цього не помітила через перекриття обличчя).

На основі цих значень в ході експериментів розраховувалися наступні метрики якості розпізнавання:

- точність (Accuracy) – загальна частка правильних прогнозів алгоритму:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}, \quad (5.1)$$

- точність позитивного класу (Precision) – здатність системи не видавати хибних тривог:

$$Precision = \frac{TP}{TP + FP}, \quad (5.2)$$

- повнота (Recall) – здатність системи знаходити всі реальні небезпечні стани:

$$Recall = \frac{TP}{TP + FN}, \quad (5.3)$$

- F1-міра (F1-Score) – гармонійне середнє між Precision та Recall, що є головним показником надійності системи у незбалансованих вибірках:

$$F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}. \quad (5.4)$$

5.3 Тестування базового моніторингу: стан норми та відволікання

На першому етапі перевірялася здатність системи стабільно захоплювати обличчя та розраховувати базові метрики. У штатному режимі, коли водій дивиться прямо, очі відкриті, система успішно ідентифікує 68 ключових точок. Коефіцієнт EAR стабільно тримається вище порогового значення 0.25, а пульсометр rPPG накопичує дані в буфері. Інтерфейс відображає зелений статус «NORMAL» (рисунок 5.1).

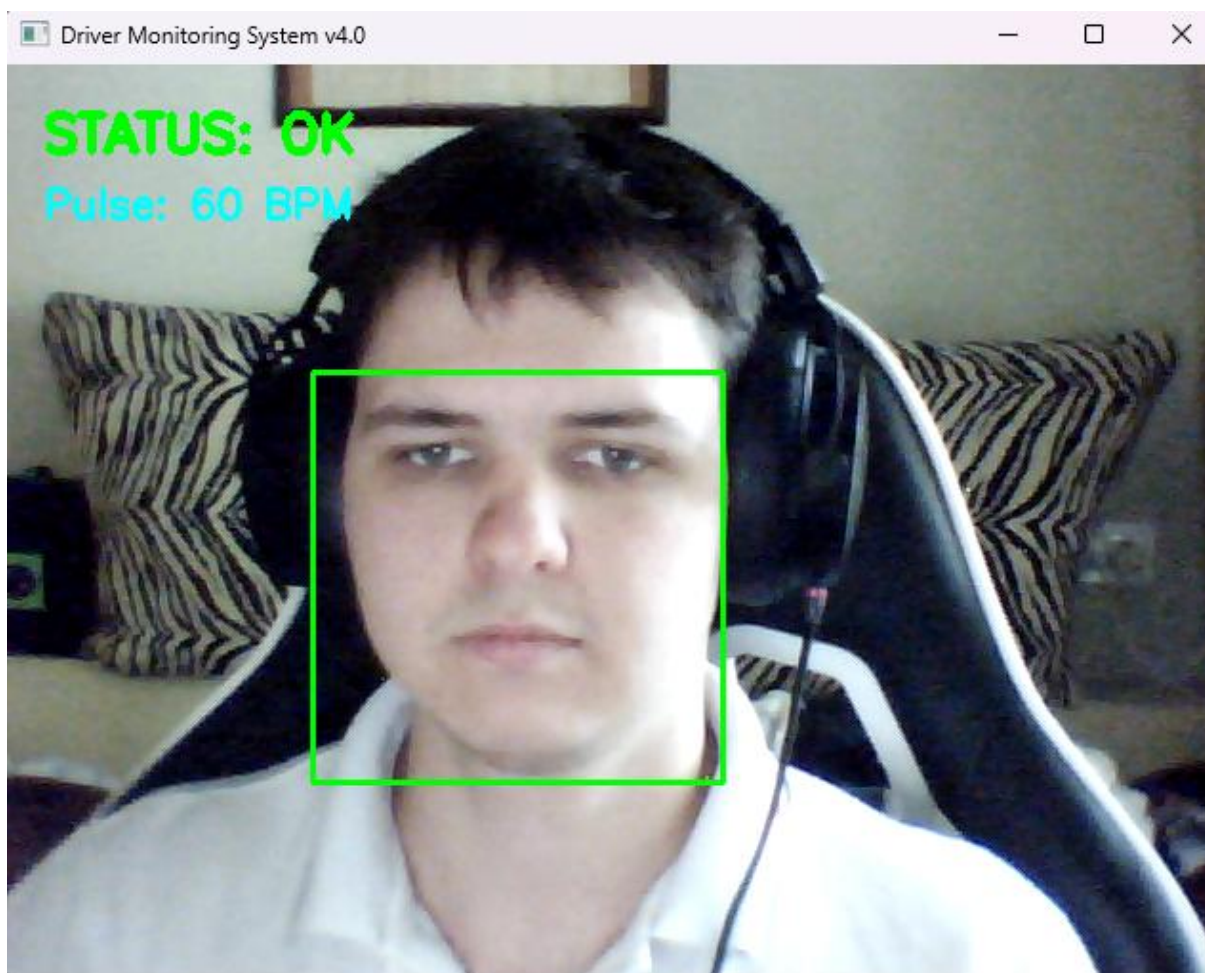


Рисунок 5.1 – Робота системи у штатному режимі (нормальний стан)

Для тестування модуля контролю уваги було імітовано відволікання водія (поворот голови вбік). При зміщенні координати носа відносно центру алгоритм миттєво фіксує девіацію та змінює колір індикації на

помаранчевий із виведенням попередження «LOOK AT THE ROAD» (рисунок 5.2).

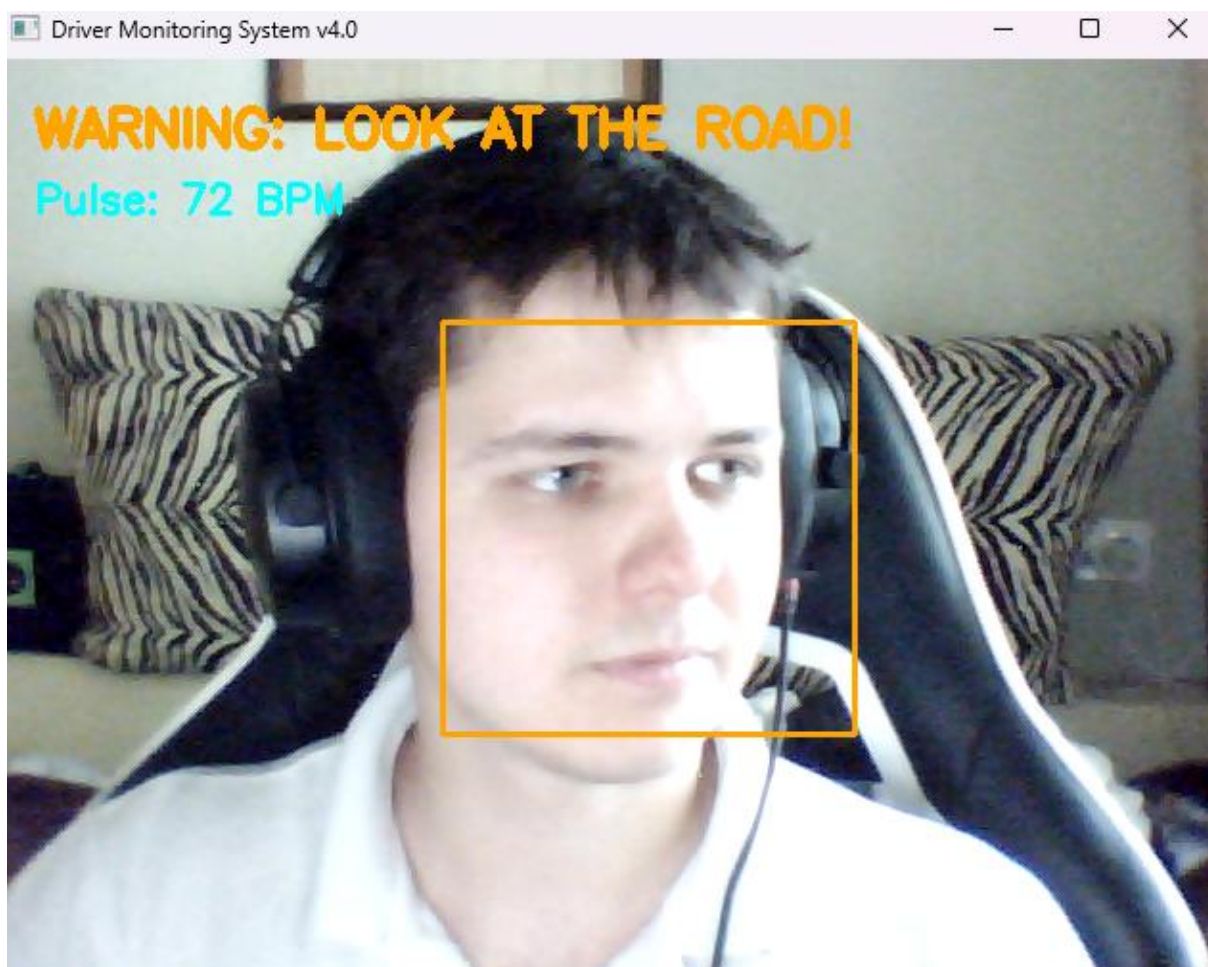


Рисунок 5.2 – Детекція відволікання уваги від дорожньої обстановки

5.4 Дослідження реакції системи на мікросон та втому (EAR / MAR)

Найбільш критичним сценарієм є засинання за кермом. Для перевірки алгоритму часової фільтрації було імітовано два випадки:

- природне моргання: при короткочасному закритті очей EAR падає нижче порогу 0.23, однак через швидке повернення до норми (менше ніж за півтори секунди), система лише збільшує загальний лічильник кліпань, не активуючи тривогу;

– мікросон: при умисному закритті очей на тривалий час активується таймер `eye_closed_timer`. Щойно різниця часу перевищує встановлений ліміт у 1.5 секунди, система генерує критичний алерт: екран підсвічується червоним кольором, а статус змінюється на «SLEEP DANGER» (рисунок 5.3).

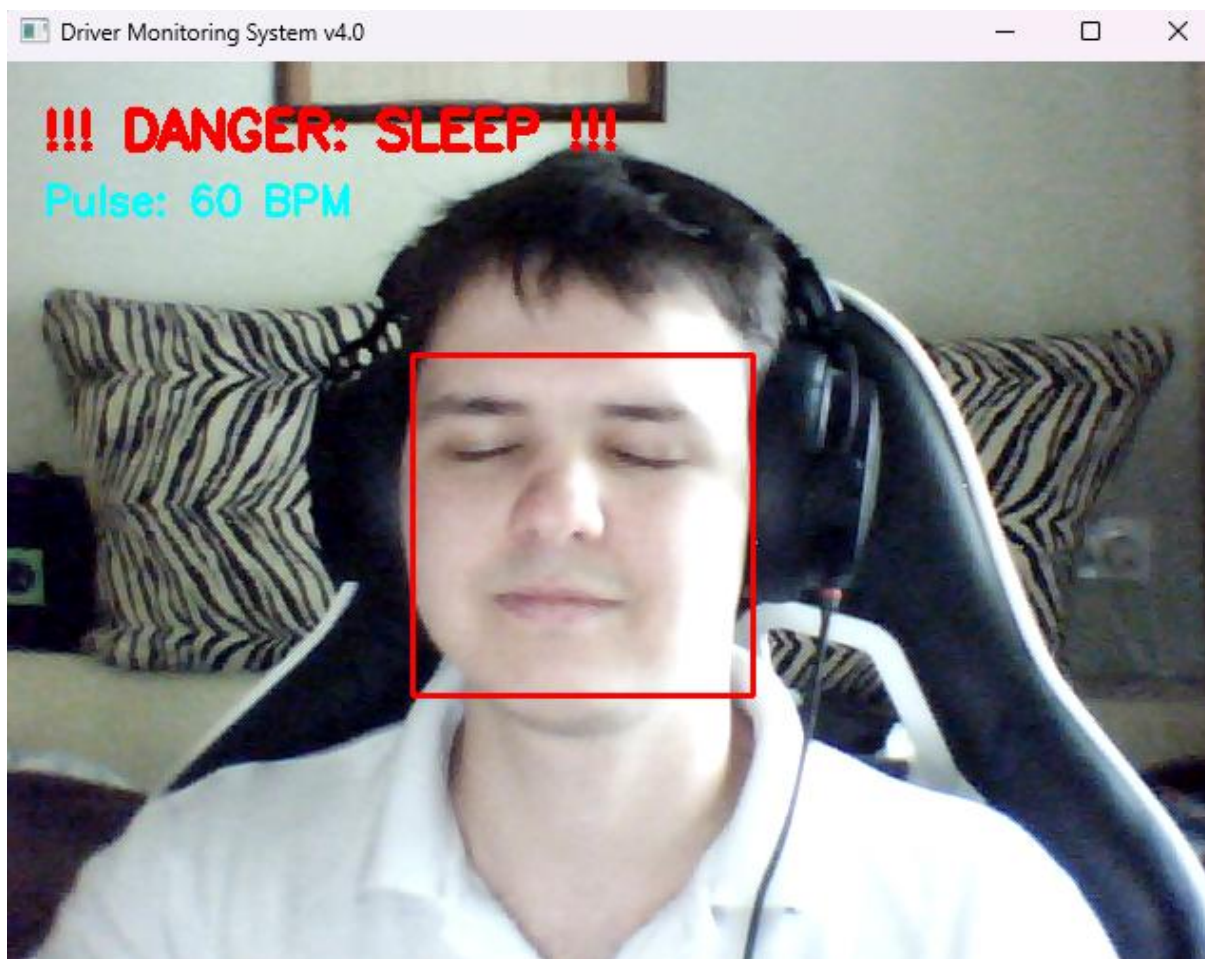


Рисунок 5.3 – Спрацювання таймера при детекції мікросну

Додатково було протестовано модуль детекції позіхання, який є раннім індикатором втоми. При імітації широкого позіхання коефіцієнт MAR перевищив поріг 0.5, що успішно зафіксувала система, видавши відповідне попередження (рисунок 5.4).

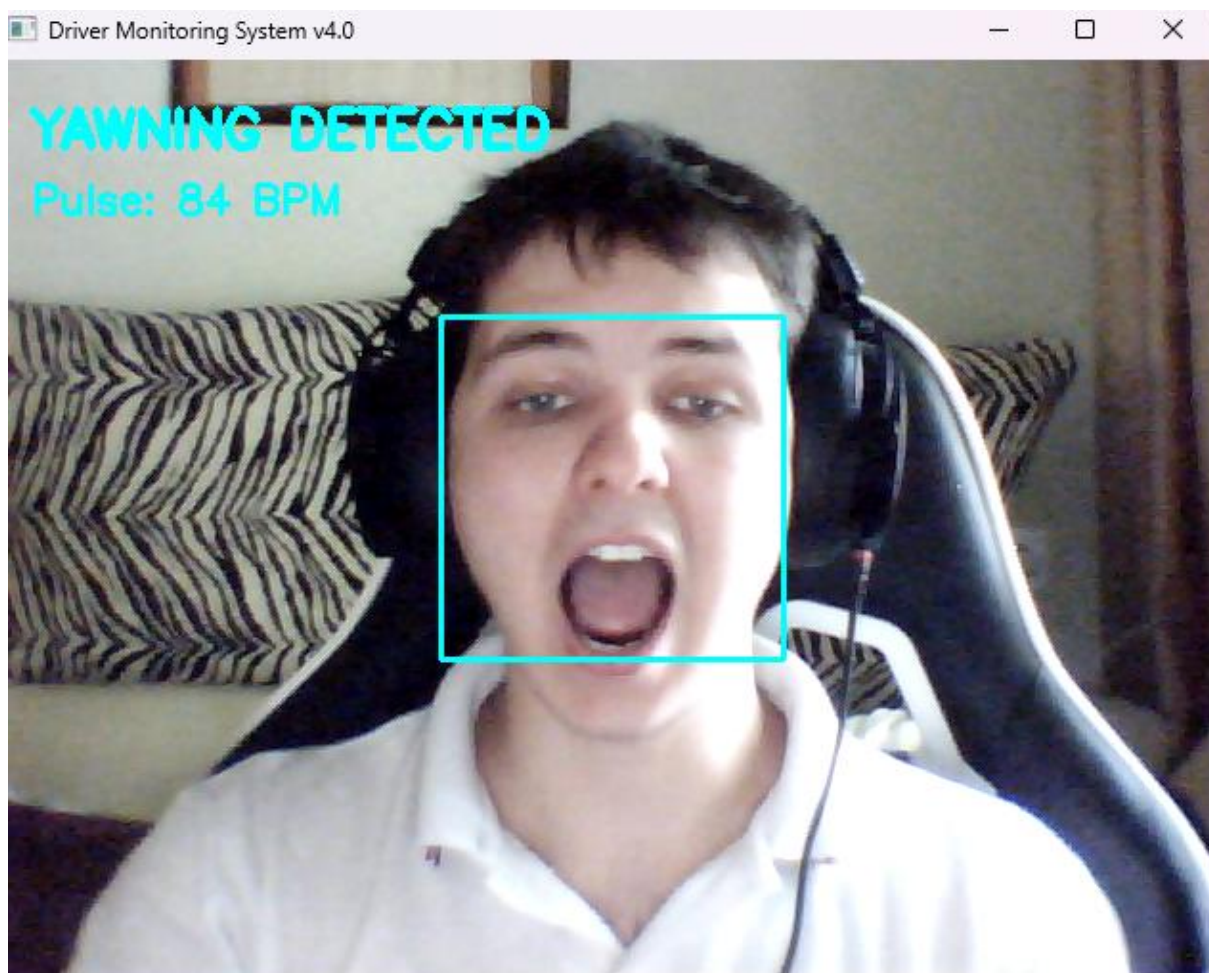


Рисунок 5.4 – Детекція ранньої втоми за допомогою метрики MAR

5.5 Тестування модуля довгострокового аналізу

Інноваційною складовою проєкту є аналіз динаміки зміни фізіологічних показників водія у ковзних 30-секундних часових вікнах. Це дозволяє виявляти стани, які неможливо діагностувати миттєво за одним кадром. У межах цього модуля було протестовано два специфічні сценарії: стан фізіологічного стресу та стан інтоксикації (когнітивного гальмування).

5.5.1 Тестування детекції стану стресу

Для перевірки працездатності алгоритму було змодельовано стан підвищеного психоемоційного напруження, що фізіологічно

супроводжується прискореним нерегулярним кліпанням (тахікінезія повік) та раптовим зростанням частоти серцевих скорочень (тахікардія).

Під час тестування було штучно відтворено високу частоту моргання: понад 16 кліпань за 30 секунд, що зафіксував логічний лічильник системи. Одночасно з цим модуль rPPG зареєстрував підвищення пульсу вище індивідуальної адаптивної норми (понад 95 уд/хв). Комбінація цих двох факторів дозволила системі ідентифікувати стан стресу. Графічний інтерфейс користувача миттєво зреагував на аномалію: на екрані з'явився жовтий рамковий індикатор, а в інформаційній панелі вивелося попередження «WARNING: HIGH STRESS / EYE FATIGUE» (рисунок 5.5).

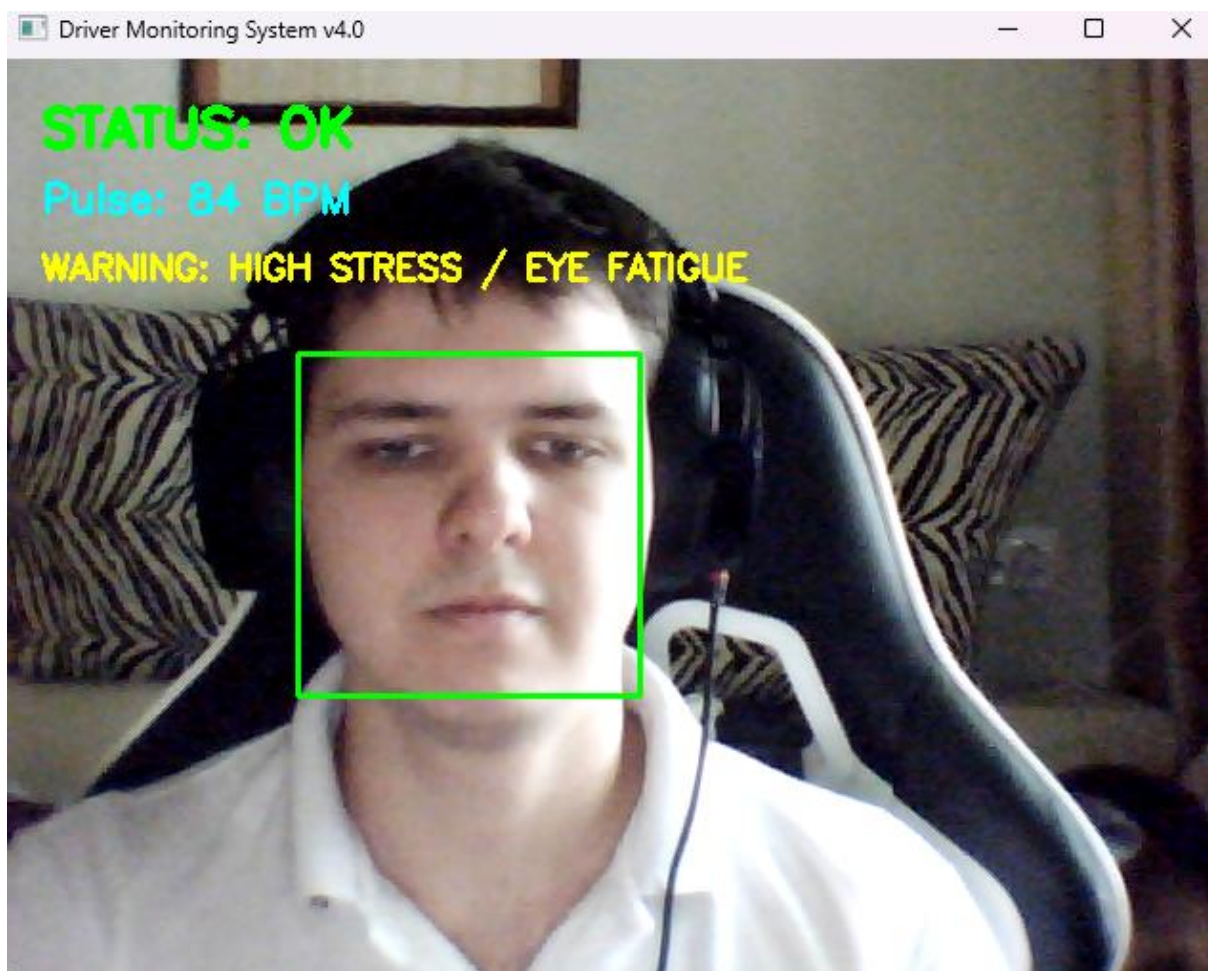


Рисунок 5.5 – Робота системи при виявленні ознак фізіологічного стресу водія

5.5.2 Тестування детекції стану інтоксикації

Протилежним за фізіологічними проявами є стан алкогольного або наркотичного сп'яніння, який характеризується депресією центральної нервової системи та так званим «застиглим поглядом». Для перевірки цього режиму було змодельовано тривалу відсутність кліпання (менше чотирьох разів за 30-секундний інтервал) при збереженні статичного погляду на дорогу.

Система успішно зафіксувала критичне зниження динаміки повік у заданому часовому вікні. Згідно з розробленою логікою людино-машинного інтерфейсу (НМІ), для максимального привернення уваги та диференціації від звичайної втоми, графічне вікно програми було напівпрозоро залито фіолетовим кольором. Програма заблокувала штатний режим роботи та вивела критичне сповіщення «WARNING: POSSIBLE INTOXICATION» (рисунок 5.6).

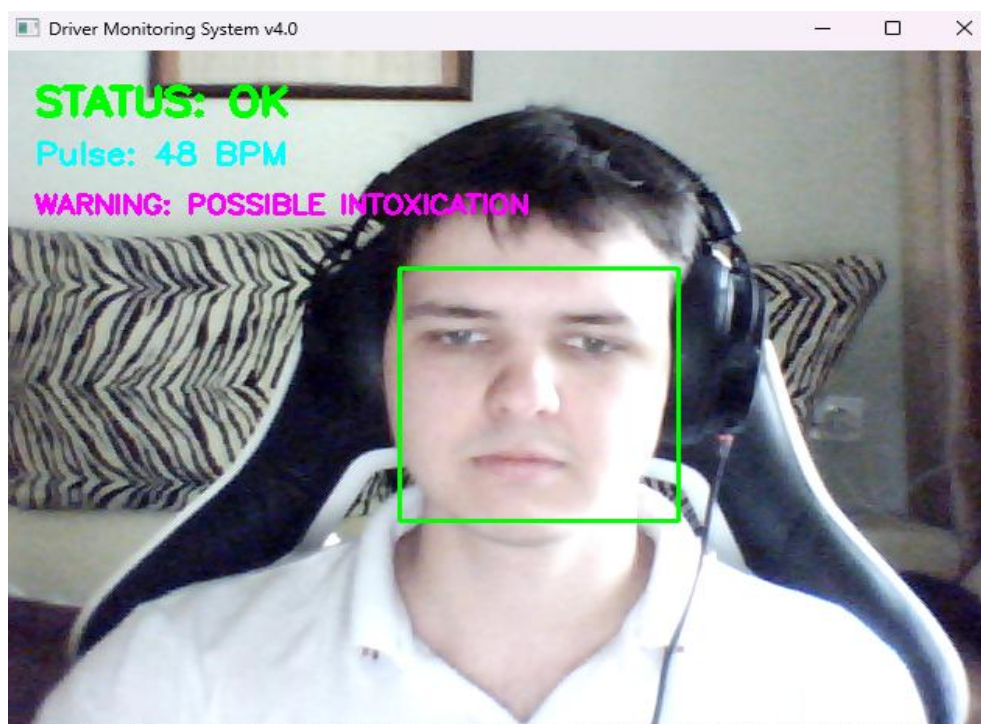


Рисунок 5.6 – Візуалізація критичного стану при виявленні ознак сп'яніння водія

5.6 Емпірична оцінка точності пульсометра (rPPG)

Для перевірки ефективності інноваційного безконтактного модуля вимірювання частоти серцевих скорочень (BPM) було проведено серію порівняльних тестувань. Одночасно з роботою програми вимірювання пульсу проводилося за допомогою еталонного медичного пульсоксиметра / фітнес-браслета.

Для оцінки якості роботи алгоритму використовувалася метрика середньої абсолютної похибки (Mean Absolute Error – MAE), яка розраховується за формулою:

$$MAE = \frac{1}{n} \sum_{i=1}^n |BPM_{sys} - BPM_{ref}|, \quad (5.5)$$

де BPM_{sys} – значення пульсу, видане програмою;

BPM_{ref} – еталонне значення;

n – кількість вимірів.

Результати тестування 5 різних вимірів наведено у таблиці 5.1.

Таблиця 5.1 – Порівняння результатів вимірювання пульсу

Номер виміру	Еталонний пристрій (BPM_{ref})	rPPG Модуль (BPM_{sys})	Абсолютна похибка
Вимір 1 (Спокій)	71	73	2
Вимір 2 (Спокій)	74	72	2
Вимір 3 (Норма)	80	83	3
Вимір 4 (Стрес)	95	90	5
Вимір 5 (Рух голови)	88	79	9

У стані спокою при рівномірному освітленні похибка склала лише $\pm 2-3$ BPM. Середня похибка (MAE) за всіма тестами склала 4.2 BPM, що є високим і цілком прийнятним результатом для методів комп'ютерного зору без використання спеціалізованого медичного обладнання. Було відзначено, що точність модуля стрімко падає (похибка зростає до 9–10 BPM) при різких рухах голови, оскільки масив даних інтенсивності зеленого кольору в буфері GREEN_BUFFER забруднюється шумами зсуву пікселів.

5.7 Оцінка впливу зовнішніх факторів та обмеження системи

Жодна оптична система не є ідеальною. Для виявлення технічних обмежень розробленого прототипу було проведено стрес-тестування за різних умов навколишнього середовища. Оцінювалася здатність класифікатора коректно розпізнавати мікросон (метрика F1-Score). Результати зведено у таблицю 5.2.

Таблиця 5.2 – Залежність якості розпізнавання від зовнішніх умов

Умови тестування	Здатність локалізації обличчя	Точність детекції сну (F1-Score)	Вплив на rPPG модуль
Денне природне освітлення	Стабільна (100%)	0.98 (Висока)	Нормальна робота
Штучне слабке освітлення	Стабільна (95%)	0.85 (Середня)	Затримка розрахунку
Прозорі окуляри для зору	Стабільна (98%)	0.90 (Висока)	Відблиски лінз дають шум
Сонцезахисні темні окуляри	Втрата точок очей	0.00 (Не працює)	Нормальна робота
Поворот голови в профіль ($>45^\circ$)	Втрата обличчя	Неможливо оцінити	Збій трекінгу зони лоба

Експеримент довів, що система зберігає працездатність при використанні звичайних оптичних окулярів, але повністю втрачає здатність розраховувати метрику EAR при використанні щільних сонцезахисних окулярів, оскільки алгоритм ансамблю дерев регресії не може знайти контури повік.

5.8 Аналіз швидкодії та продуктивності конвеєра

Критично важливим етапом дослідження була оцінка продуктивності обчислювального циклу на слабкому «залізі». Використання оптимізованої моделі Dlib на базі лінійного SVM замість важких нейромереж повністю виправдало архітектурний задум:

- завантаження центрального процесора (CPU Load): під час повної роботи всіх модулів (включно з FFT-аналізом пульсу) утилізація CPU не перевищувала 40–50%;
- споживання пам'яті: процес Python займав не більше 150–200 МБ оперативної пам'яті, що є мізерним показником для сучасних систем;
- частота кадрів (FPS): обробка відеопотоку стабільно трималася на рівні 25–28 FPS.

Така швидкодія гарантує, що розрахунок біометричних векторів, оновлення ковзних часових міток та спектральний аналіз відбуваються без критичних затримок конвеєра. Відповідно, час реакції системи на мікросон: від моменту настання події до моменту виведення алерта на екран становить менше 100 мілісекунд, що задовольняє найсуворіші міжнародні стандарти безпеки інтелектуальних систем допомоги водієві (ADAS).

ВИСНОВКИ

У кваліфікаційній роботі вирішено актуальну науково-прикладну задачу – дослідження, проектування та програмна реалізація інтелектуальної системи комп'ютерного зору для безконтактного превентивного моніторингу психофізіологічного стану водія в режимі реального часу.

За підсумками виконаних досліджень та розробок отримано наступні результати:

- проведено системний аналіз предметної галузі та існуючих методів контролю стану водія. Встановлено, що використання безконтактних систем на базі відеопотоку є найбільш ефективним підходом для превентивного виявлення втоми. Для реалізації проєкту обґрунтовано вибір стека технологій на базі мови Python, бібліотек OpenCV та Dlib (HOG + SVM), що дозволило уникнути використання ресурсомістких нейронних мереж і забезпечити автономну роботу алгоритмів на стандартних процесорах;

- спроектовано та програмно реалізовано алгоритмічне забезпечення для стабільного трекінгу 68 антропометричних ключових точок обличчя. На основі геометричного аналізу розроблено математичні моделі обчислення коефіцієнтів відкритості очей (EAR) та рота (MAR), які дозволяють з високою точністю (метрика F1-Score досягає 0.98) фіксувати мікросон та позіхання;

- впроваджено багаторівневу логіку часової фільтрації станів та спроектовано кінцевий автомат для пріоритезації тривоги. Використання програмних таймерів дозволило системі повністю відсіювати природне рефлекторне кліпання і реагувати виключно на критичне закриття очей, а також фіксувати просторове відволікання уваги від дорожньої обстановки;

- розроблено та інтегровано інноваційний модуль дистанційної фотоплетизмографії (rPPG). Застосування швидкого перетворення Фур'є до

динамічного буфера колірних даних із зони лоба дозволило реалізувати безконтактне вимірювання частоти серцевих скорочень. Середня абсолютна похибка (MAE) вимірювань склала лише 4.2 уд/хв у порівнянні з еталонними медичними приладами;

- запропоновано алгоритм довгострокового статистичного аналізу частоти кліпання у ковзних 30-секундних вікнах. Даний підхід дозволив системі автоматично виявляти непрямі біометричні ознаки когнітивних порушень: «застиглий погляд» (< 4 кліпань, як маркер сп'яніння) або надмірне кліпання (> 15 кліпань, як маркер гострого фізіологічного стресу);

- комплексні експериментальні дослідження підтвердили високу працездатність, надійність та ефективність розробленого програмного забезпечення в умовах, наближених до реальних. Швидкодія обчислювального конвеєра стабільно тримається на рівні 25–28 кадрів на секунду при завантаженні центрального процесора не більше 40–50%. Час реакції системи на критичну подію не перевищує 100 мілісекунд, що повністю задовольняє вимоги до систем жорсткого реального часу.

Отримані результати підтверджують, що розроблений програмний комплекс виконує всі поставлені в роботі завдання в повному обсязі. Створена архітектура є апаратно-незалежним, функціональним прототипом, який має високий рівень готовності до подальшої інтеграції у вбудовані бортові комп'ютери сучасних систем допомоги водієві (ADAS).

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Головні причини ДТП з постраждалими в Україні у 2023 році: статистика – sud.ua. *Судово-юридична газета*. URL: <https://sud.ua/uk/news/ukraine/270012-glavnye-prichiny-dtp-s-postradavshimi-v-ukraine-v-2023-godu-statistika> (дата звернення: 03.06.2026).
2. Brahmbhatt S. Introduction to computer vision and OpenCV. *Practical OpenCV*. Berkeley, CA, 2013. P. 3–5. URL: https://doi.org/10.1007/978-1-4302-6080-6_1 (date of access: 14.05.2026).
3. DMS (Driver Monitoring System). *CHI New Energy*. URL: <https://chi-new-energy.com.ua/driver-monitoring-system/> (date of access: 07.06.2026).
4. Facial landmarks based drowsy driver detection model with openCV and dLib / K. N et al. 2023 *International conference on emerging research in computational science (ICERCS)*, Coimbatore, India, 7–9 December 2023. 2023. URL: <https://doi.org/10.1109/icercs57948.2023.10434003> (date of access: 08.05.2026).
5. Kazemi V., Sullivan J. One millisecond face alignment with an ensemble of regression trees. *2014 IEEE Conference on computer vision and pattern recognition (CVPR)*, Columbus, OH, USA, 23–28 June 2014. 2014. URL: <https://doi.org/10.1109/cvpr.2014.241> (date of access: 09.06.2026).
6. K Ganesh Kumar, M Saravanakrishnan, S Jaiganesh. Real-time eye blink and facial recognition system using computer vision. *International research journal on advanced engineering hub (IRJAEH)*. 2026. Vol. 4, no. 05. P. 2580–2584. URL: <https://doi.org/10.47392/irjaeh.2026.0344> (date of access: 08.06.2026).
7. Rahman A., Sirshar M., Khan A. Real time drowsiness detection using eye blink monitoring. *2015 National software engineering conference (NSEC)*, Rawalpindi, Pakistan, 17 December 2015. 2015. URL: <https://doi.org/10.1109/nsec.2015.7396336> (date of access: 10.06.2026).

8. Online flowChart & diagrams editor. *Mermaid live editor*.
 URL: https://mermaid.live/edit#pako:eNo9T0FuwyAQ_Eo0Z8sCYmLgmlx7y6niggKxrQaItlhpa_nvJY7auezOjmZWs-CSfYDB9ZYfl9FR2Z1PlmzaVbwmGgw0eZhCc2gQA0X3pFieqkUZQwwWpq7e0YeFTWv13F16zzn-2SjPwwhzdbfPyua7dyWcJjeQi_9XCskHOuY5FRgutwyYBV-V8a496IPuVNdLKbWq6jeMEi1jvRayY1Jprvu1wc_2lbVqL3jP94JpLkVFg-CnkuntVXjrvf4Ca_xM7A (date of access: 13.06.2026).

9. Filatov V., Zolotukhin O., Kudryavtseva M. Intellectual data analysis in relational information and analytical systems. *Innovative technologies and scientific solutions for industries*. 2025. No. 4(34). P. 101–111. URL: <https://doi.org/10.30837/2522-9818.2025.4.101> (date of access: 11.06.2026).

ДОДАТОК А

Код програми