

Вінницький національний технічний університет
Факультет інформаційних технологій та комп'ютерної інженерії
Кафедра програмного забезпечення

Бакалаврська кваліфікаційна робота

на тему: «Розробка програмного застосунку для діагностування захворювань
на основі антропологічних вимірювань голови людини»

Виконав: студент 4 курсу
групи 5ПІ-21Б
спеціальності

121 – Інженерія програмного забезпечення

(шифр і назва напрямку підготовки, спеціальності)

Мазур Вадим Вячеславович

(прізвище та ініціали)

Керівник: д.т.н., проф. каф. ПЗ Романюк О.Н.

(прізвище та ініціали)

Рецензент: к.т.н., каф. ОТ Снігур А.В.

(прізвище та ініціали)

Допущено до захисту

Завідувач кафедри ПЗ
д.т.н., проф. Романюк О.Н.
«09» 06 2025 р.

ВНТУ – 2025

Вінницький національний технічний університет
 Факультет інформаційних технологій та комп'ютерної інженерії
 Кафедра програмного забезпечення
 Рівень вищої освіти перший (бакалаврський)
 Галузь знань 12 – Інформаційні технології
 Спеціальність 121 – Інженерія програмного забезпечення
 Освітньо-професійна програма – Інженерія програмного забезпечення

ЗАТВЕРДЖУЮ
 Завідувач кафедри ПЗ
 О. Н. Романюк
 «21» березня 2025 р.

ЗАВДАННЯ НА БАКАЛАВРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ

Мазуру Вадиму Вячеславовичу

1. Тема роботи – «Розробка програмного застосунку для діагностування захворювань на основі антропологічних вимірювань голови людини»

Керівник роботи: Романюк Олександр Никифорович, д.т.н., проф. кафедри ПЗ затверджені наказом вищого навчального закладу від 20 березня 2025 р. № 97.

2. Строк подання студентом роботи 2 червня 2025 р.

3. Вихідні дані до роботи: мова програмування Python, бібліотеки Scipy, MediaPipe та Numpy, середовище розробки PyCharm, розмір екрану – 1280 * 1024, тип мережі зображення – полігональна.

4. Зміст текстової частини: вступ; аналіз стану розвитку систем діагностування захворювань на основі антропологічних вимірювань голови людини; розробка методів та алгоритмів програмного застосунку; розробка програмних модулів та методів реалізації системи діагностування захворювань; тестування системи; висновки; список використаних джерел; додатки.

5. Перелік ілюстративного матеріалу: титульний слайд; актуальність теми; мета, об'єкт та предмет дослідження; задачі дослідження, наукова новизна, практична цінність одержаних результатів; порівняльний аналіз аналогів. діаграми, схеми архітектури системи, інтерфейси користувача, таблиці з характеристиками функціональних модулів, а також графічні зображення алгоритмів обробки даних.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада Консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1-4	Романюк О.Н., д.т.н., професор кафедри ПЗ	24.03.2025	01.06.2025

7. Дата видачі завдання 24 березня 2025 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів бакалаврської кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Аналіз стану розвитку систем діагностування захворювань на основі антропологічних вимірювань голови людини	25.03.25 – 31.03.25	<i>всех</i>
2	Розробка методів та алгоритмів програмного застосунку	03.04.25 – 09.04.25	<i>всех</i>
3	Варіантний аналіз та обґрунтування вибору засобів програмної реалізації	11.04.25 – 17.04.25	<i>всех</i>
4	Розробка програмних модулів та методів системи діагностування захворювань на основі антропологічних вимірювань голови людини	19.04.25 – 03.05.25	<i>всех</i>
5	Тестування програмного застосунку	05.05.25 – 18.05.25	<i>всех</i>
6	Оформлення матеріалів до захисту БКР	20.05.25 – 30.05.25	<i>всех</i>

Студент

 Мазур В.В.
(підпис) (ініціали та прізвище)

Керівник бакалаврської кваліфікаційної роботи

 Романюк О.Н.
(підпис) (ініціали та прізвище)

АНОТАЦІЯ

УДК 004.9:572.5:616-073

Мазур В.В. Розробка програмного застосунку для діагностування захворювань на основі антропологічних вимірювань голови людини: бакалаврська кваліфікаційна робота зі спеціальності 121 – інженерія програмного забезпечення, освітня програма – інженерія програмного забезпечення. Вінниця: ВНТУ, 2025. 78 с.

На укр. мові. Бібліогр. : 23 назв ; рис. : 27 ; табл. : 5.

У роботі представлено розробку програмного застосунку для діагностування захворювань на основі антропологічних вимірювань голови людини. Актуальність теми зумовлена необхідністю неінвазивних, доступних і точних методів попередньої діагностики, особливо у сфері генетичних і неврологічних захворювань.

У роботі виконано огляд існуючих систем та методів антропологічних вимірювань, проаналізовано сучасні інструменти комп'ютерного зору (MediaPipe, OpenCV), а також бібліотеки чисельного аналізу (NumPy, SciPy). Запропоновано гібридну модель, яка поєднує скелетне полігональне моделювання, кореляційний аналіз антропометричних точок та методи комп'ютерного зору для ідентифікації ознак синдромів Дауна, Аперта, Тернера та інших.

Проведено тестування точності та ефективності діагностичних модулів. Додаток дозволяє здійснювати автоматизований аналіз симетрії, розрахунок ключових параметрів (довжина, ширина, міжточкові відстані) та їх порівняння з нормативними базами. Результати можуть бути використані в телемедицині, скринінгових програмах та освітніх цілях.

Ключові слова: антропометрія, діагностика, комп'ютерний зір, скелетна модель, Streamlit, Python.

ABSTRACT

UDK 004.9:572.5:616-073

Mazur V.V. Development of a software application for diagnosing diseases based on anthropological measurements of the human head: bachelor's thesis in speciality 121 – software engineering, educational programme – software engineering. Vinnytsia: VNTU, 2025. 78 c.

In Ukrainian. Bibliography: 23 titles; fig.: 37; tab. 5.

This paper presents the development of a software application for diagnosing diseases based on anthropological measurements of the human head. The relevance of the topic is driven by the need for non-invasive, affordable and accurate methods of preliminary diagnosis, especially in the field of genetic and neurological diseases.

The paper reviews existing systems and methods for anthropological measurements, analyses modern computer vision tools (MediaPipe, OpenCV) and numerical analysis libraries (NumPy, SciPy). A hybrid model is proposed that combines skeletal polygonal modelling, correlation analysis of anthropometric points and computer vision methods to identify the signs of Down, Apert, Turner and other syndromes.

The accuracy and efficiency of the diagnostic modules were tested. The application allows for automated symmetry analysis, calculation of key parameters (length, width, inter-point distances) and their comparison with regulatory bases. The results can be used in telemedicine, screening programmes and educational purposes.

Keywords: anthropometry, diagnostics, computer vision, skeletal model, Streamlit, Python.

ЗМІСТ

ВСТУП.....	4
1 АНАЛІЗ МЕТОДІВ І ЗАСОБІВ ДІАГНОСТУВАННЯ ЗАХВОРЮВАНЬ НА ОСНОВІ АНТРОПОЛОГІЧНИХ ВИМІРЮВАНЬ ГОЛОВИ ЛЮДИНИ	8
1.1 Аналіз технологій систем антропологічних вимірювань голови людини ...	8
1.2 Порівняльний аналіз існуючих систем	11
1.3 Порівняльний аналіз методів розв’язання поставлених завдань	17
1.4 Постановка задачі дослідження.....	19
1.5 Висновок	19
2 РОЗРОБКА МЕТОДІВ ТА АЛГОРИТМІВ ПРОГРАМНОГО ЗАСТОСУНКУ	21
2.1 Розробка скелетних полігональних моделей для діагностування генетичних захворювань людини.....	21
2.2 Розробка методу антропологічних вимірювань голови з використанням кореляційних зв’язків між різними морфометричними параметрами голови.	28
2.3 Розробка гібридного методу аналізу зображень обличчя для виявлення ознак генетичних захворювань.....	32
2.4 Висновок	34
3. РОЗРОБКА ПРОГРАМНИХ МОДУЛІВ І МЕТОДІВ СИСТЕМИ ДІАГНОСТУВАННЯ ЗАХВОРЮВАНЬ НА ОСНОВІ АНТРОПОЛОГІЧНИХ ВИМІРЮВАНЬ ГОЛОВИ ЛЮДИНИ.....	36
3.1 Обґрунтування вибору засобів реалізації та системи середовища розробки застосунку	36
3.2 Розробка інтерфейсу програмного застосунку	40
3.3 Алгоритми роботи програмного застосунку	41
3.4 Висновок	46
4. ВИКОРИСТАННЯ ТРИВИМІРНОГО МОДЕЛЮВАННЯ ДЛЯ АНТРОПОЛОГІЧНИХ ЗАДАЧ	48
4.1 Антропологічні вимірювання обличчя людини за тривимірною моделлю	48
4.2 Методи отримання тривимірної моделі обличчя людини	52
4.3 Рекомендації щодо отримання тривимірної моделі обличчя за допомогою телефону.....	54
4.4 Рекомендації по вибору програмного забезпечення для отримання тривимірної моделі.....	58

5. ТЕСТУВАННЯ ПРОГРАМНОЇ СИСТЕМИ ДІАГНОСТУВАННЯ ЗАХВОРЮВАНЬ НА ОСНОВІ АНТРОПОЛОГІЧНИХ ВИМІРЮВАНЬ ГОЛОВИ ЛЮДИНИ	62
5.1 Тестування програмної системи	62
5.2 Визначення технічних вимог	70
5.3 Розробка інструкції для користувачів	71
5.4 Висновок	73
ВИСНОВКИ.....	74
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	76
ДОДАТКИ.....	79
Додаток А – Технічне завдання	80
Додаток Б – Протокол перевірки на плагіат.....	Помилка! Закладку не визначено.
Додаток В – Лістинг програми	84
Додаток Г – Ілюстративна частина	109
Додаток Д – Акт впровадження.....	119

ВСТУП

Обґрунтування вибору теми дослідження. Сучасна медицина стрімко розвивається в напрямку цифровізації, активно впроваджуючи інноваційні технології для покращення діагностичних, терапевтичних та профілактичних процесів. Одним із ключових напрямів є використання біометричних та антропологічних даних для отримання об'єктивної інформації про фізіологічний стан пацієнтів. Завдяки цьому стає можливим виявлення патологічних змін на ранніх етапах, що має вирішальне значення для своєчасного лікування та запобігання ускладненням.

Особливу зацікавленість викликає створення інтелектуальних систем, здатних здійснювати попередню діагностику захворювань шляхом аналізу анатомічних характеристик людини.

Зокрема, одним із перспективних напрямів є вивчення геометричних параметрів голови, таких як форма черепа, пропорції обличчя, симетрія, співвідношення між ключовими антропометричними точками (наприклад, відстань між очима, довжина носа, висота чола, ширина вилиць тощо). Ці параметри є надзвичайно інформативними в медичному контексті, оскільки дозволяють виявляти відхилення від нормативних значень, які можуть свідчити про наявність або ризик розвитку різноманітних патологічних станів.

Так, наприклад, аномальна форма черепа може бути ознакою краніостенозу — вродженого захворювання, при якому відбувається передчасне закриття швів черепа, що впливає на нормальний розвиток мозку та черепної коробки. Нерівномірна симетрія обличчя або черепа може сигналізувати про неврологічні порушення, зокрема пов'язані з ураженням черепно-мозкових нервів або м'язової системи. У сфері нейрохірургії геометричні вимірювання використовуються для попереднього планування оперативних втручань, а також для оцінки результатів реабілітації.

Незважаючи на наявність окремих досліджень у цій сфері, існуючі програмні продукти здебільшого мають низький рівень універсальності. Вони не

враховують індивідуальні морфологічні особливості, не забезпечують гнучкої адаптації до різних вікових, статевих та етнічних груп користувачів, а також не завжди гарантують достатню точність при обробці вимірювань. Крім того, багато рішень не мають зручного інтерфейсу або достатньої інтеграції з іншими медичними інформаційними системами.

У зв'язку з цим виникає актуальна потреба у створенні високоточних, адаптивних та інтелектуальних систем обробки антропологічних даних, які могли б не лише фіксувати вимірювання, але й здійснювати їх аналіз з урахуванням медичних норм, контексту пацієнта та потенційних ризиків. Така система могла б стати дієвим інструментом для попередньої діагностики, скринінгових досліджень, моніторингу розвитку дітей, а також для роботи в умовах телемедицини, коли фізичний огляд пацієнта ускладнений.

Саме тому обрана тема є актуальною та важливою, адже поєднує сучасні досягнення в галузях комп'ютерного зору, машинного навчання, аналізу біомедичних даних і програмної інженерії.

Зв'язок роботи з науковими програмами, планами, темами. Робота виконувалася згідно плану виконання наукових досліджень на кафедрі програмного забезпечення.

Мета та завдання дослідження. Метою роботи є підвищення оперативності діагностики генетичних захворювань за рахунок розробки геометричних та морфологічних моделей комп'ютерного аналізу обличчя.

Основними задачами дослідження є:

- аналіз існуючих методів і засобів антропологічних вимірювань голови людини;
- розробка методів скелетного полігонального аналізу форми та розташування лицьових елементів для ідентифікації потенційних симптомів захворювань, таких як синдром Дауна, синдром Аперта, синдром Вільямса та інші;
- розробка підвищення аналізу зображень обличчя, що поєднує чисельні метрики та морфологічний аналіз ключових зон голови;

- розробка геометричну модель взаємопов'язаних елементів обличчя з урахуванням можливості виявлення змін та патологій;
- розробити алгоритми, програмні модулі та систему візуалізації з урахуванням розроблених методів;
- виконати експериментальну перевірку ефективності розроблених методів.

Об'єкт дослідження — процес діагностування захворювань на основі антропологічних вимірювань.

Предмет дослідження — методи і засоби антропологічних вимірювань для задач діагностування.

Методи дослідження. У процесі досліджень використовувались: методи чисельного аналізу, теорії чисел та аналітичної геометрії для побудови математичних моделей антропологічних вимірювань, а також методи комп'ютерного моделювання для аналізу анатомічних параметрів, імітації патологічних відхилень та перевірки достовірності діагностичних рішень, отриманих на основі морфологічних особливостей будови голови.

Наукова новизна отриманих результатів:

1. Вперше розроблено скелетну полігональну геометричну модель взаємопов'язаних елементів обличчя, особливість якої полягає у можливості комп'ютерного аналізу видозміни форми та стану лицьових параметрів, що дає можливість оперативного діагностування генетичних захворювань таких як: синдром Дауна, синдром Тернена, прогресивний параліч, нейрофіброматоз, синдром Вільямса, синдром Нунан, синдром Аперта та інші.

2. Розробка моделей аналізу зображень обличчя для виявлення ознак генетичних захворювань, що поєднує метрики з локалізованим морфологічним аналізом ключових зон голови, зокрема очей, лоба, носа, рота та підборіддя. Метод включає сегментацію обличчя з подальшим виявленням характерних морфологічних ознак, таких як асиметрія, диспропорція, або аномальні співвідношення між елементами обличчя.

Практична цінність отриманих результатів полягає в розробці програмного застосунку, що аналізує антропологічні вимірювання голови людини для виявлення морфологічних ознак можливих захворювань шляхом вимірювання ключових анатомічних параметрів та їх порівняння з нормативними значеннями.

Особистий внесок здобувача. Усі представлені в роботі результати отримані особисто автором. У друкованих працях, опублікованих у співавторстві, автору належать такі результати: використання індексів голови для діагностування [1], алгоритм визначення симетрії обличчя [2], алгоритми діагностування захворювання [3], вибір методу зафарбовування для формування зображення голови [4], алгоритм формування зображення рельєфної поверхні голови [5], використання відеокادر для діагностування [6], використання планшетів для введення зображення [7], аналіз контролю версій [8], аналіз особливостей растрової графіки [9], використання скінчених елементів для симуляції зображення [10].

Апробація результатів роботи. Основні положення та результати роботи були представлені на наукових конференціях: Матеріали LIV Всеукраїнська науково-технічна конференція факультету інформаційних технологій та комп'ютерної інженерії (2025), XVIII Всеукраїнська науково-практична web конференція аспірантів, студентів та молодих вчених (2025), Матеріали VI Всеукраїнської науково-практичної інтернет-конференції студентів, аспірантів та молодих вчених (2023), XXIV Всеукраїнська науково-технічна конференція молодих вчених, аспірантів та студентів (2024), Збірник матеріалів міжнародної науково-практичної Інтернет-конференції (2023), XXXIII Міжнародна науково-практична конференція «Інформаційні технології: наука, техніка, технологія, освіта, здоров'я» (MicroCAD–2025).

Структура та обсяг БКР. БКР складається зі вступу, п'яти розділів, висновків, списку літератури, що містить 23 найменування та 5 додатки.

1 АНАЛІЗ МЕТОДІВ І ЗАСОБІВ ДІАГНОСТУВАННЯ ЗАХВОРЮВАНЬ НА ОСНОВІ АНТРОПОЛОГІЧНИХ ВИМІРЮВАНЬ ГОЛОВИ ЛЮДИНИ

1.1 Аналіз технологій систем антропологічних вимірювань голови людини

Антропологічні вимірювання голови людини є важливим напрямком біометричних досліджень, який у 2025 році демонструє активний розвиток завдяки інтеграції комп'ютерного зору, штучного інтелекту та мобільних технологій. За даними дослідницького центру Biometric Research Group, глобальний ринок біометричних технологій, включаючи системи антропометричних вимірювань, досяг обсягу понад 45 мільярдів доларів, з яких приблизно 18% припадає на технології, пов'язані з аналізом параметрів голови та обличчя людини.

Сучасні методи антропологічних вимірювань голови людини можна умовно розділити на дві основні категорії: контактні та безконтактні. Традиційні контактні методи, які потребують фізичного вимірювання за допомогою спеціальних інструментів (краніометри, штангенциркулі, антропометри), поступово витісняються безконтактними технологіями комп'ютерного зору та 3D-моделювання. За останніми даними, частка безконтактних методів у професійній антропометрії зросла до 76% у 2024 році порівняно з 52% у 2020 році.

Технологічною основою сучасних систем антропологічних вимірювань є алгоритми комп'ютерного зору для визначення ключових точок (landmarks) на обличчі та голові людини. Бібліотека MediaPipe, розроблена Google, стала одним із стандартів індустрії завдяки можливості точного визначення 468 точок обличчя в режимі реального часу. Вона використовується у 64% систем антропометрії, що працюють на мобільних пристроях та веб-платформах.

Архітектура сучасних програмних рішень у галузі антропометрії голови переважно базується на модульному підході. Базовий модуль захоплення та

аналізу зображення використовує технології комп'ютерного зору (OpenCV) для попередньої обробки зображень та відеопотоку.

Модуль визначення антропометричних точок застосовує нейромережеві алгоритми (часто на базі MediaPipe або спеціалізованих CNN-моделей) для точного позиціонування ключових точок на обличчі та голові. Модуль розрахунку метрик використовує математичні бібліотеки (NumPy, SciPy) для обчислення відстаней, кутів та індексів на основі визначених точок. Модуль візуалізації та інтерфейсу користувача найчастіше реалізується за допомогою веб-технологій або крос-платформних фреймворків [11].

Веб-орієнтовані рішення, що використовують Python з фреймворком Streamlit, показують значне зростання популярності завдяки простоті розробки та розгортання. За даними GitHub, кількість репозиторіїв з проєктами антропометрії на базі Streamlit зросла на 138% за останній рік. Ця технологія дозволяє швидко створювати інтерактивні веб-додатки для візуалізації та аналізу даних антропометричних вимірювань без необхідності глибоких знань веб-розробки.

Алгоритми комп'ютерного зору для антропометрії продовжують вдосконалюватися. Останні версії OpenCV (4.8+) у поєднанні з бібліотекою MediaPipe забезпечують точність визначення антропометричних точок з похибкою менше 0,5 мм при оптимальних умовах освітлення та позиціонування, що наближається до точності професійних медичних систем. Бібліотека NumPy оптимізована для швидких матричних обчислень, що критично важливо для розрахунку численних антропометричних показників у режимі реального часу.

Серед сучасних технологічних викликів у галузі антропометрії голови залишаються проблеми точності вимірювань при різних умовах освітлення, варіаціях пози голови та різноманітності антропологічних типів людей. Дослідження показують, що точність визначення антропометричних точок знижується на 12-18% при відхиленні голови більше ніж на 30° від фронтального положення.

Новітні методики глибокого навчання, зокрема трансферне навчання та генеративні змагальні мережі (GAN), дозволяють частково вирішувати ці проблеми. Наприклад, підхід Multi-view Consistency Training забезпечує стабільність вимірювань при різних кутах повороту голови шляхом навчання моделі на синтетичних даних, генерованих з використанням 3D-моделей.

Важливим трендом розвитку систем антропологічних вимірювань є інтеграція з мобільними пристроями. Оптимізовані версії MediaPipe та OpenCV для мобільних платформ дозволяють проводити вимірювання безпосередньо на смартфонах з достатньою для багатьох застосувань точністю. Це відкриває нові можливості для телемедицини, персоналізованої косметології та систем підбору окулярів, головних уборів тощо.

Аналіз існуючих програмних рішень для антропометрії голови виявляє ряд обмежень. Комерційні системи часто є закритими і дорогими, що обмежує їх доступність для освітніх та дослідницьких цілей. Відкриті рішення на базі Python та JavaScript зазвичай фокусуються на окремих аспектах вимірювань або мають обмежену функціональність. Публічні API для антропометрії, такі як Face++ та Amazon Rekognition, надають базові можливості визначення ключових точок обличчя, але не забезпечують повний спектр антропологічних вимірювань та обмежені політиками конфіденційності.

Розробка власної системи антропологічних вимірювань голови людини на базі Python, Streamlit, MediaPipe, NumPy та OpenCV є обґрунтованою з огляду на декілька ключових факторів. По-перше, відкритий характер цих технологій забезпечує гнучкість та контроль над функціональністю системи. По-друге, модульна архітектура дозволяє поетапно розширювати можливості системи, додаючи нові метрики та функції аналізу. По-третє, веб-орієнтований підхід на базі Streamlit забезпечує доступність системи на різних платформах без необхідності створення нативних додатків [12].

Отже, аналіз сучасного стану технологій систем антропологічних вимірювань голови людини свідчить про динамічний розвиток цієї галузі та наявність суттєвих можливостей для інновацій. Технологічні тренди, включаючи

комп'ютерний зір, глибоке навчання та веб-технології, формують нове покоління систем антропометрії. Водночас існують значні технологічні виклики, пов'язані з точністю, надійністю та універсальністю вимірювань, що створює простір для розробки нових, більш ефективних рішень на базі сучасних технологій Python, Streamlit, MediaPipe, NumPy та OpenCV.

1.2 Порівняльний аналіз існуючих систем

Сучасний ринок програмного забезпечення для антропологічних вимірювань голови людини представлений як комерційними, так і відкритими рішеннями різного рівня функціональності, точності та складності у використанні. Такі системи зазвичай застосовуються в галузях медицини, біоінженерії, криміналістики, а також у виробництві індивідуальних засобів захисту та персоналізованих виробів (наприклад, шоломів або окулярів).

Проведений аналіз ринку дозволив визначити чотири основні програмні аналоги, що представляють різні підходи до вирішення задачі антропометрії голови людини. Серед них є як автономні настільні застосунки, так і веборієнтовані сервіси, які використовують сучасні технології комп'ютерного зору, машинного навчання та 3D-моделювання. Виявлено значні відмінності між цими рішеннями за критеріями точності вимірювань, доступності, інтеграції з апаратними засобами та можливості адаптації під специфічні завдання користувача.

Аналіз дозволив не лише виявити технічні та функціональні особливості наявних систем, але й окреслити їхні недоліки, такі як висока вартість, обмежений доступ до коду, складність налаштування, низька гнучкість або відсутність підтримки української мови. Ці обмеження були враховані при формулюванні вимог до власної розробки та стали основою для визначення напрямів її вдосконалення.

3dMD Face System – комерційне рішення для медичних закладів та наукових лабораторій, що забезпечує тривимірне сканування голови пацієнта за допомогою спеціалізованого обладнання. Система 3dMD створює високоточні

3D-моделі з похибкою менше 0,2 мм, що дозволяє проводити прецизійні вимірювання для пластичної хірургії, ортодонції та наукових досліджень (див. рисунок 1.1).



Рисунок 1.1 – Приклад роботи застосунка 3dMD Face System

Серед головних переваг цього рішення можна відзначити надзвичайно високу точність вимірювань та можливість експорту даних у різні формати. Крім того, воно має потужне програмне забезпечення для аналізу результатів.

Однак суттєвими недоліками системи є висока вартість (від 60 тисяч доларів за базову комплектацію), необхідність спеціального приміщення для встановлення обладнання, складність використання без спеціальної підготовки та обмежена мобільність. Також система вимагає регулярного технічного обслуговування та оновлення ліцензій на програмне забезпечення, що збільшує загальну вартість володіння.

Face Analyzer від компанії FaceMetrics – програмне рішення з відкритим вихідним кодом на основі Python, яке використовує бібліотеку dlib для визначення 68 ключових точок обличчя (див. рисунок 1.2).

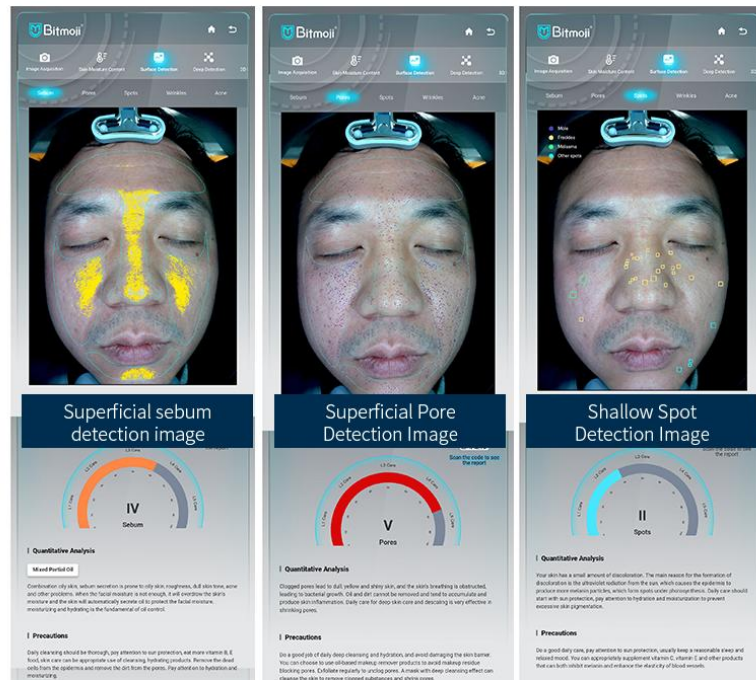


Рисунок 1.2 – Приклад роботи застосунка Face Analyzer

Цей інструмент орієнтований на наукові дослідження та освітні цілі, забезпечуючи базові можливості антропометрії обличчя. До переваг Face Analyzer відносяться безкоштовність, можливість встановлення на стандартні комп'ютери без спеціального обладнання, відкритий код з можливістю модифікації та розширення функціональності. Водночас система має обмежену кількість антропометричних точок порівняно з професійними рішеннями, нижчу точність вимірювань (особливо при несприятливих умовах освітлення або нестандартних положеннях голови), відсутність автоматизованих звітів та обмежені можливості візуалізації результатів. Крім того, Face Analyzer вимагає від користувача базових навичок програмування для встановлення та налаштування через командний рядок, що ускладнює його використання непрофесійними користувачами.

MorphoJ – спеціалізоване програмне забезпечення для геометричної морфометрії, що використовується в антропології, археології та біології. Ця програма дозволяє аналізувати форму та розміри об'єктів, включаючи людську голову, на основі координат визначених точок (див. рисунок 1.3).

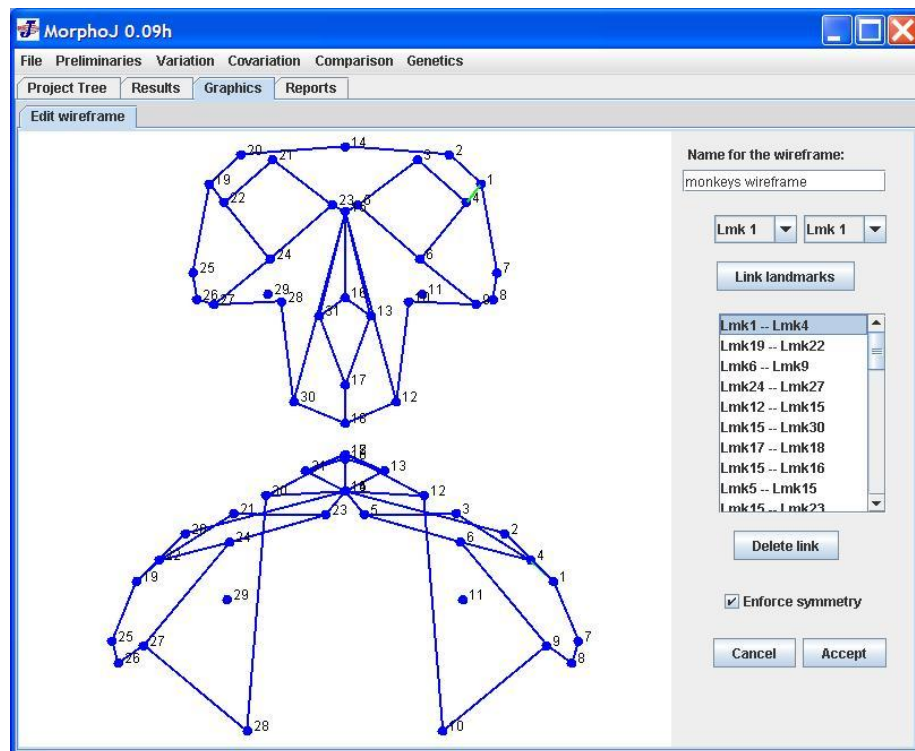


Рисунок 1.3 – Приклад роботи застосунка MorphoJ

MorphoJ надає потужні інструменти для статистичного аналізу, включаючи прокрустову суперпозицію, аналіз головних компонент, канонічний варіативний аналіз та інші методи багатовимірної статистики. Серед переваг програми варто відзначити безкоштовність, потужний математичний апарат для наукових досліджень, можливість порівняння великих наборів даних та генерації детальних статистичних звітів.

Основними недоліками MorphoJ є відсутність вбудованих засобів для автоматичного визначення антропометричних точок (координати повинні бути підготовлені заздалегідь в інших програмах), складний спеціалізований інтерфейс, орієнтований на професійних дослідників, відсутність можливості роботи в реальному часі та обмежені засоби візуалізації для непрофесійних користувачів. Також програма не підтримує обробку відеопотоку, що обмежує її застосування для динамічних вимірювань.

HeadScanner Pro – мобільний додаток для iOS та Android, який використовує фронтальну камеру смартфона для визначення основних параметрів голови людини. Додаток орієнтований на масовий ринок та

використовується для підбору головних уборів, окулярів та інших аксесуарів (див. рисунок 1.4).

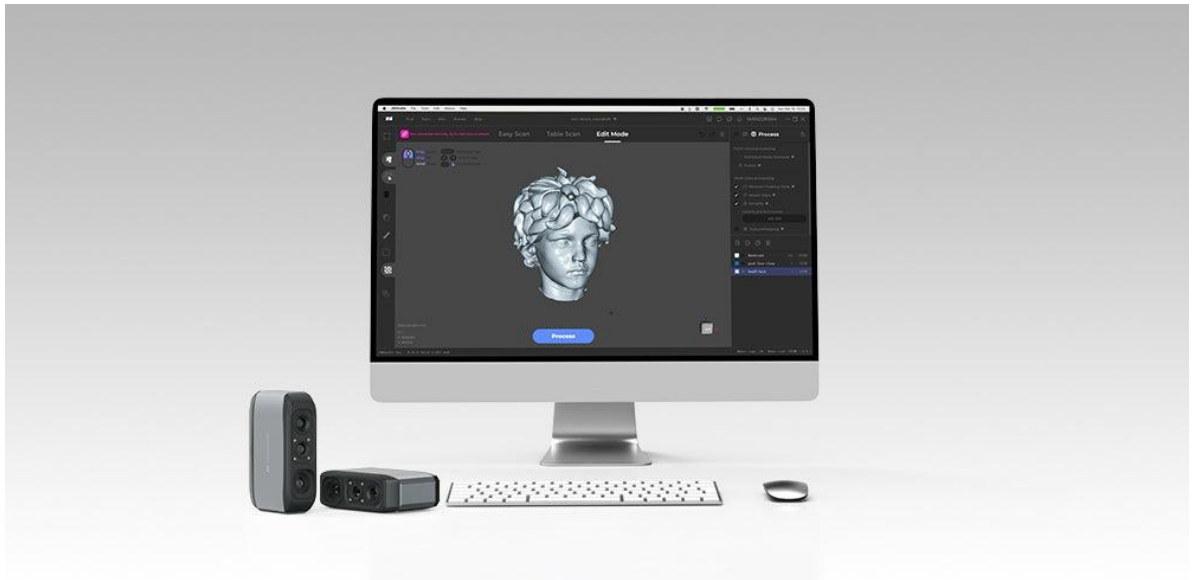


Рисунок 1.4 – Приклад роботи застосунка HeadScanner Pro

Головними перевагами цього рішення є висока доступність (можливість використання на звичайному смартфоні), простий інтуїтивно зрозумілий інтерфейс, швидкість отримання результатів (менше 30 секунд для повного сканування) та інтеграція з онлайн-магазинами аксесуарів.

До недоліків HeadScanner Pro відносяться обмежена точність вимірювань порівняно з професійними системами, залежність від якості камери смартфона, обмежений набір вимірюваних параметрів (фокус на комерційно привабливих характеристиках, а не на повному антропологічному профілі), відсутність можливості експорту даних у наукові формати та закритий характер алгоритмів, що не дозволяє оцінити методологічну обґрунтованість вимірювань.

Розроблюваний власний додаток «AnthroHead» позиціонується як збалансоване рішення, що поєднує наукову точність з доступністю та зручністю використання. Результати порівняльного аналізу функціональних можливостей розглянутих систем наведено у таблиці 1.1.

Таблиця 1.1 – Порівняльна характеристика аналогів

Характеристика	3dMD Face System	Face Analyzer	MorphoJ	HeadScanner Pro	Власний додаток
Тип системи	Апаратно - програмний комплекс	Програмне забезпечення з відкритим кодом	Програмне забезпечення для аналізу даних	Мобільний додаток	Веб-додаток з відкритим кодом
Платформа	Власна станція на Windows	Windows, Linux, MacOS	Windows, Linux, MacOS, Java-based	iOS, Android	Будь-яка з веб-браузером
Точність вимірювань	0.2 мм	1-2 мм	Залежить від вхідних даних	3-5 мм	0.5-1 мм
Кількість антропометричних точок	>25,000 точок (повна 3D-модель)	68 точок	Не обмежена (залежить від підготовлених даних)	23 точки	468 точок (MediaPipe)
Можливість роботи в реальному часі	Так	Обмежена	Ні	Так	Так
Автоматизація вимірювань	Висока	Середня	Низька (ручна підготовка даних)	Висока	Висока
Ціна	Від 60,000\$	Безкоштовно	Безкоштовно	4.99\$	Безкоштовно
Відкритість коду	Закритий	Відкритий	Відкритий	Закритий	Відкритий
Можливість експорту даних	Широкі можливості	Базові (CSV, TXT)	Розширені для наукових форматів	Обмежені	Розширені (CSV, JSON, PDF)
Зручність використання	Потребує спеціальної підготовки	Потребує технічних навичок	Складна для непрофесіоналів	Проста	Середня
Підтримка статистичного аналізу	Розширена	Базова	Розширена	Відсутня	Базова з можливістю розширення
Візуалізація результатів	Високоякісна 3D	Базова 2D	Спеціалізована наукова	Спрощена комерційна	Інтерактивна 2D/3D

Аналіз порівняльної таблиці дозволяє зробити висновок, що розроблюваний додаток «AnthroHead» займає унікальну нішу між професійними дорогими системами та обмеженими безкоштовними або мобільними

рішеннями. Основною перевагою власного додатку є поєднання високої точності вимірювань завдяки використанню MediaPipe (468 точок на обличчі) з доступністю веб-інтерфейсу та відсутністю необхідності в спеціальному обладнанні. На відміну від Face Analyzer, запропоноване рішення забезпечує значно більшу кількість антропометричних точок та кращу точність вимірювань, а порівняно з MorphoJ – надає можливість автоматичного визначення точок та роботи в реальному часі.

Порівняно з комерційним рішенням 3dMD Face System, розроблюваний додаток має перевагу в доступності та простоті використання, хоча і поступається в абсолютній точності вимірювань. У порівнянні з мобільним додатком HeadScanner Pro, «AnthroHead» забезпечує більшу кількість вимірюваних параметрів, вищу точність та можливість розширеного експорту даних для наукових та освітніх цілей, але потенційно може мати складніший інтерфейс користувача.

Відкритість коду та використання популярних бібліотек (MediaPipe, OpenCV, NumPy) дозволяє розроблюваній системі бути легко модифікованою та розширюваною відповідно до конкретних потреб користувачів, що є значною перевагою порівняно з закритими комерційними рішеннями. Водночас веб-орієнтований підхід на базі Streamlit забезпечує кросплатформність та доступність без необхідності складних процедур встановлення, що вигідно відрізняє його від десктопних рішень типу Face Analyzer та MorphoJ.

Таким чином, розроблюваний додаток «AnthroHead» заповнює існуючу на ринку прогалину між високоточними, але дорогими професійними системами та доступними, але обмеженими масовими рішеннями, пропонуючи збалансований набір функцій для наукових, освітніх та практичних застосувань у галузі антропологічних вимірювань голови людини.

1.3 Порівняльний аналіз методів розв’язання поставлених завдань

Розроблений програмний застосунок для діагностування захворювань на основі антропологічних вимірювань голови людини реалізовано із

використанням сучасних технологій, які забезпечують ефективність, масштабованість та зручність у використанні. Основу розробки становить мова програмування Python, а також низка інструментів для створення застосунків.

Python використовується як основна мова для розробки серверної логіки через свою універсальність, простоту та наявність широкого спектру бібліотек, що особливо важливо у медичному застосуванні. Streamlit обрано завдяки його легкості та здатності інтегруватися з модулями для обробки медичних та біометричних даних, зокрема бібліотеками для статистичного аналізу та машинного навчання, такими як Pandas, NumPy, SciKit-Learn та TensorFlow. Це дає змогу застосунку здійснювати попередню обробку антропологічних параметрів, проводити класифікацію та виводити попередні медичні висновки.

Клієнтська частина розроблена з використанням HTML та CSS, що дозволило створити інтуїтивно зрозумілий та адаптивний інтерфейс для користувача. Інтерфейс орієнтований на медичних працівників та дослідників і дозволяє зручно вводити вимірювання, переглядати результати діагностики та зберігати історію обстежень [13]. Всі елементи інтерфейсу — форми, кнопки, таблиці, графіки — оформлені у відповідності до принципів UI/UX дизайну, що забезпечує комфортну роботу зі застосунком.

У процесі розробки проводилось як модульне, так і інтеграційне тестування основних функцій: введення даних, обробки вимірювань, генерації медичних висновків, зберігання та пошуку інформації. Також проводились навантажувальні тести, що дозволили переконатися у стабільній роботі системи під час активного використання.

Для розробки було використано інтегроване середовище PyCharm, яке має зручні інструменти для налагодження коду, роботи з базами даних, тестування та керування версіями коду за допомогою Git. Це дозволило оптимізувати процес розробки та забезпечити високу якість кінцевого продукту.

Таким чином, розроблений програмний застосунок є ефективним інструментом, який поєднує методи антропології, інформатики та медицини для попередньої діагностики захворювань. Застосування сучасних технологій

забезпечує швидкий аналіз, зручність використання та високий рівень достовірності результатів.

1.4 Постановка задачі дослідження

На основі аналізу існуючих аналогічних продуктів та виявлення їх сильних та слабких сторін були сформульовані основні завдання, які необхідно виконати для успішної розробки програмного застосунку для системи на основі антропологічних вимірювань голови людини. Програмне забезпечення буде розроблено з використанням технології Streamlit у середовищі розробки Python, із застосуванням експертних та аналітичних методів оцінювання.

Основними задачами дослідження є:

- проведення аналізу існуючих методів і засобів антропологічних вимірювань голови людини з метою комп'ютерного розпізнавання ознак генетичних захворювань за зображенням обличчя;
- методи скелетного аналізу форми та розташування лицьових елементів для ідентифікації потенційних симптомів захворювань, таких як синдром Дауна, синдром Аперта, синдром Вільямса та інші;
- методи підвищення аналізу зображень обличчя, що поєднує чисельні метрики та морфологічний аналіз ключових зон голови;
- розробити геометричну модель взаємопов'язаних елементів обличчя з урахуванням можливості виявлення змін та патологій;
- розробити алгоритми, програмні модулі та систему візуалізації з урахуванням розроблених методів;
- виконати експериментальну перевірку ефективності розроблених методів.

1.5 Висновок

У першому розділі було проаналізовано сучасний стан технологій систем антропологічних вимірювань голови людини. Розглянуто основні методи комп'ютерного зору, зокрема технології OpenCV, MediaPipe та алгоритми

машинного навчання, що використовуються у подібних системах. Також проведено аналіз існуючих програмних рішень, таких як 3DMD, Artec Studio, Bellus3D, FaceGen та інші, які реалізують функції захоплення, аналізу та обробки 2D/3D-зображень для медичних або аналітичних цілей.

Порівняльний аналіз існуючих систем дозволив визначити їх сильні та слабкі сторони, серед яких — висока точність та функціональність, але водночас і висока вартість, складність у використанні або обмежена доступність. На основі цього було обґрунтовано доцільність розробки власної програмної системи, яка дозволить здійснювати базові антропологічні вимірювання голови людини з використанням доступних технологій та інструментів.

Отже, результати аналізу сформували теоретичне та технологічне підґрунтя для розробки програмної системи «AnthroHead», яка буде орієнтована на ефективність, доступність та зручність у використанні.

2 РОЗРОБКА МЕТОДІВ ТА АЛГОРИТМІВ ПРОГРАМНОГО ЗАСТОСУНКУ

2.1 Розробка скелетних полігональних моделей для діагностування генетичних захворювань людини

Використання комп'ютерної графіки, зокрема скелетних полігональних моделей, є важливим інструментом у діагностиці та моніторингу різноманітних генетичних захворювань. Це дозволяє точно аналізувати морфологічні особливості обличчя, виявляти типові зміни, властиві тим чи іншим синдромам, і покращувати якість медичного супроводу.

1. Синдром Дауна (трисомія 21) – це поширене генетичне захворювання, що виникає внаслідок трисомії по 21-й хромосомі. Люди з цим синдромом мають характерні зовнішні риси, які можна точно ідентифікувати за допомогою скелетних полігональних моделей обличчя.

До основних особливостей зовнішності належать: пласке обличчя, монголоїдний розріз очей, короткий ніс, зменшена нижня щелепа, товсті губи та коротка шия. Застосування комп'ютерної графіки та скелетних полігональних моделей у випадку синдрому Дауна:

- Раннє виявлення: аналіз форми обличчя дозволяє виявити типові риси синдрому ще на ранніх стадіях життя або навіть у перинатальному періоді.
- Об'єктивність оцінки: точне математичне моделювання дозволяє уникнути суб'єктивності в діагностиці, що підвищує надійність результатів.
- Моніторинг розвитку: скелетні полігональні моделі можуть відображати динаміку змін у рисах обличчя протягом життя, що важливо для спостереження за ефектами лікування або розвитку супутніх порушень.
- Наукове значення: систематичне дослідження моделей обличчя при синдромі Дауна сприяє вдосконаленню методів діагностики та розумінню патофізіології хвороби.

На зображенні 2.1 представлено скелетну модель черепа з позначеними антропометричними точками, які часто використовуються для діагностування

синдрому Дауна на основі особливостей черепно-лицьової морфології.

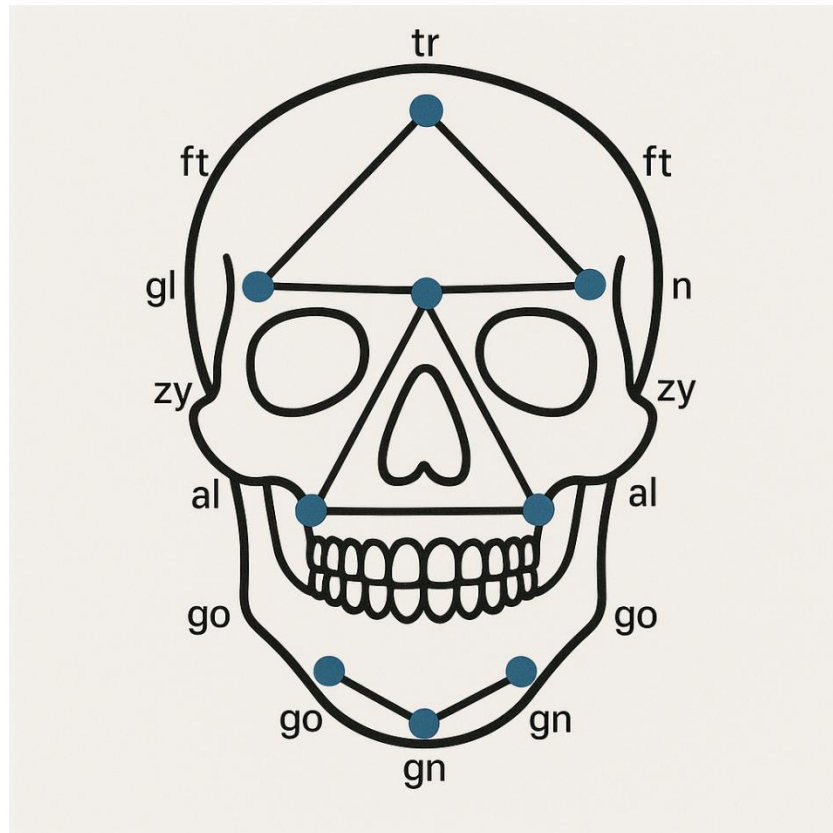


Рисунок 2.1 – Скелетна полігональна модель для генетичного захворювання синдром Дауна

Нижче подано пояснення ключових точок та продемонстровано на зображенні 2.2 полігональну модель:

- Glabella (gl) — точка між бровами; вимірюється для оцінки ширини чола;
- Frontotemporale (ft) — точки з боків чола, що використовуються для оцінки його форми та ширини;
- Nasion (n) — місце перетину носових кісток і лобної; важлива точка при оцінці профілю;
- Subnasale (sn) — точка під носом; використовується для вимірювання довжини носа;
- Zygion (zy) — найбільш виступаючі точки вилиць (з обох боків); вимірюється біпарієтальна ширина;
- Gnathion (gn) — найнижча точка підборіддя; використовується для

оцінки форми нижньої щелепи;

- Gonion (go) — кути нижньої щелепи (зліва і справа); дозволяють оцінити її ширину;

- Cheilion (ch) — куточки рота; використовуються для обчислення міжротової відстані;

- Alare (al) — найбільш латеральні точки крил носа; використовується для обчислення ширини носа;

- Trichion (tr) — середина лінії росту волосся (умовна точка в дослідженнях лобового профілю).

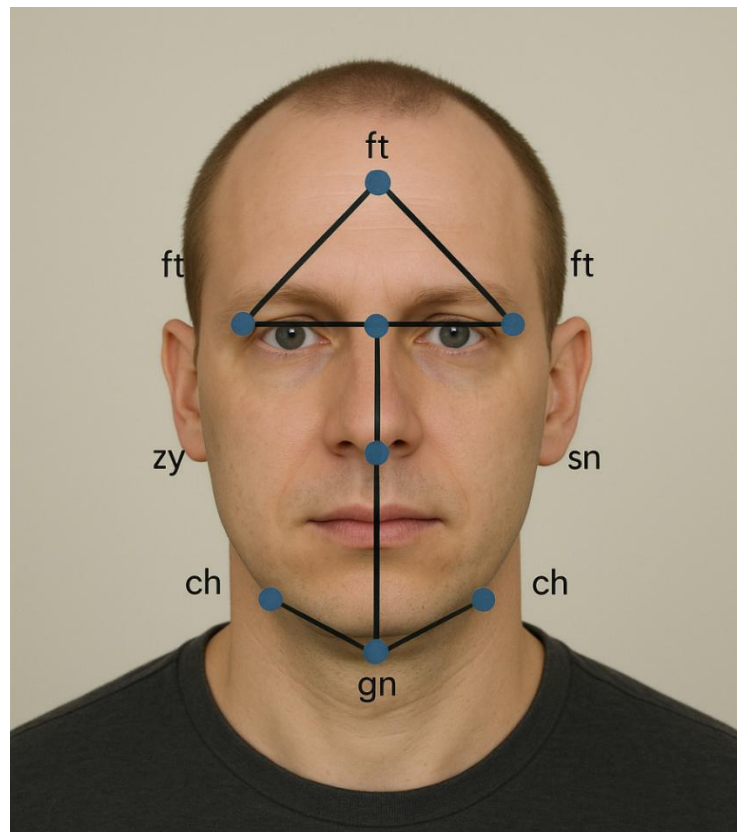


Рисунок 2.2 – Скелетна полігональна модель людини для генетичного захворювання синдром Дауна

2. Синдром Тернера (моносомія X) виникає у дівчаток у результаті повної або часткової відсутності однієї з X-хромосом. Це захворювання часто супроводжується характерними морфологічними рисами обличчя. До них відносять: широку шию з крилоподібними складками, низько посаджені вуха,

вузький рот, невисоке чоло та мікрогнатію (зменшену нижню щелепу). Використання скелетних полігональних моделей дозволяє:

- Виявлення типових рис: системи комп'ютерного аналізу обличчя можуть точно виявити характерні особливості, властиві пацієнткам з синдромом Тернера.
- Клінічна валідація: модельні дані можуть підтвердити клінічний діагноз, а також дозволити здійснювати повторний аналіз у динаміці.
- Підбір терапії: оцінка вираженості змін допомагає у плануванні естетичних або ортодонтичних втручань.

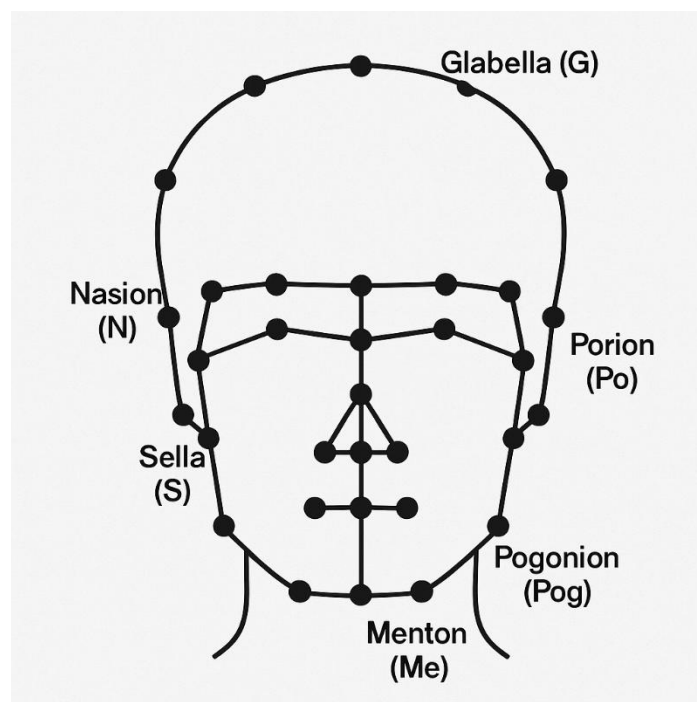


Рисунок 2.3 – Скелетна полігональна модель для генетичного захворювання синдром Тернера

Нижче подано пояснення ключових точок та продемонстровано на зображенні 2.3 полігональну модель:

- Glabella (g) – найвищя точка лоба між бровами;
- Pogonion (Pog) - найвищя точка підборіддя;
- Menton (Me) - Найнижча точка нижньої щелепи;
- Sella (s) – вкорочена шия;

- Porion (po) – верхній край зовнішнього слухового отвору;
- Nasion (n) – точка з'єднання лобової та носової кістки;

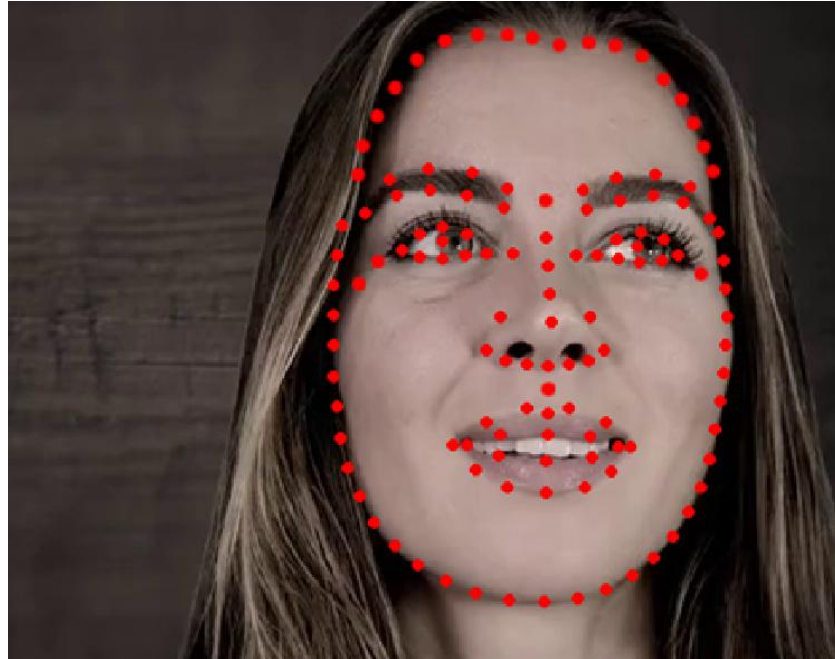


Рисунок 2.4 – Скелетна полігональна модель людини для генетичного захворювання синдром Тернера

3. Синдром Аперта рідкісне генетичне захворювання, спричинене мутацією гена FGFR2. Він характеризується краніосиностозом (передчасним зрощенням швів черепа), що призводить до деформацій обличчя, а також синдактилією (зрощенням пальців).

Серед ключових зовнішніх ознак: опуклий лоб, гіпертелоризм (велика відстань між очима), короткий ніс з широким переніссям, а також зміщення верхньої щелепи. Скелетні полігональні моделі сприяють:

- Детальному аналізу лицьової структури: дозволяють моделювати складні деформації черепа і обличчя з високою точністю.
- Плануванню хірургічного втручання: 3D-моделі допомагають хірургам планувати корекційні операції і оцінювати їх результати.
- Моніторингу розвитку: дають змогу фіксувати зміни в обличчі дитини протягом росту та розвитку, що критично важливо при краніофациальних

аномаліях.

- Проведенню наукових досліджень: збір даних з великої кількості випадків допомагає створювати статистичні моделі для прогнозування та діагностики.

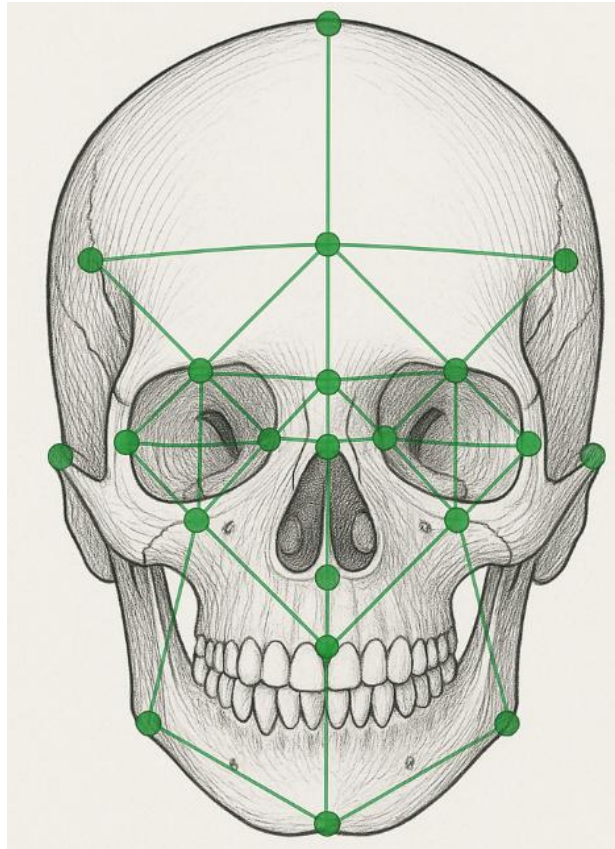


Рисунок 2.5 – Скелетна полігональна модель для генетичного захворювання синдром Аперта

Нижче подано пояснення ключових точок та продемонстровано на зображенні 2.3 полігональну модель:

1. Верхівка черепа (vertex) – найвища точка черепа; використовується для оцінки форми черепної коробки.
2. Глабела (glabella) – міжбрів'я над носом; важливий орієнтир при визначенні деформації лобової кістки.
3. Латеральні надбрівні точки (lateral brow points) – зони над краями брів; показують розширення або звуження чола.

4. Внутрішні кути очей (endocanthion) – місця де внутрішній кут ока зустрічається з носом; допомагають оцінити гіпертелоризм (широке розташування очей).

5. Зовнішні кути очей (exocanthion) – зовнішні кути очей; використовуються для вимірювання очної щілини.

6. Носова перегородка (subnasale) – точка під носом, де носова перегородка зустрічається з верхньою губою.

7. Ліве та праве крило носа (alar curvature points) – найбільш виступаючі точки крил носа; оцінюють форму носа.

8. Фільтрум (philtrum midpoint) – середина западини між носом і верхньою губою; важливий для оцінки пропорцій середньої частини обличчя.

9. Центр підборіддя (gnathion) – нижня точка підборіддя; допомагає оцінити вертикальне положення щелепи.

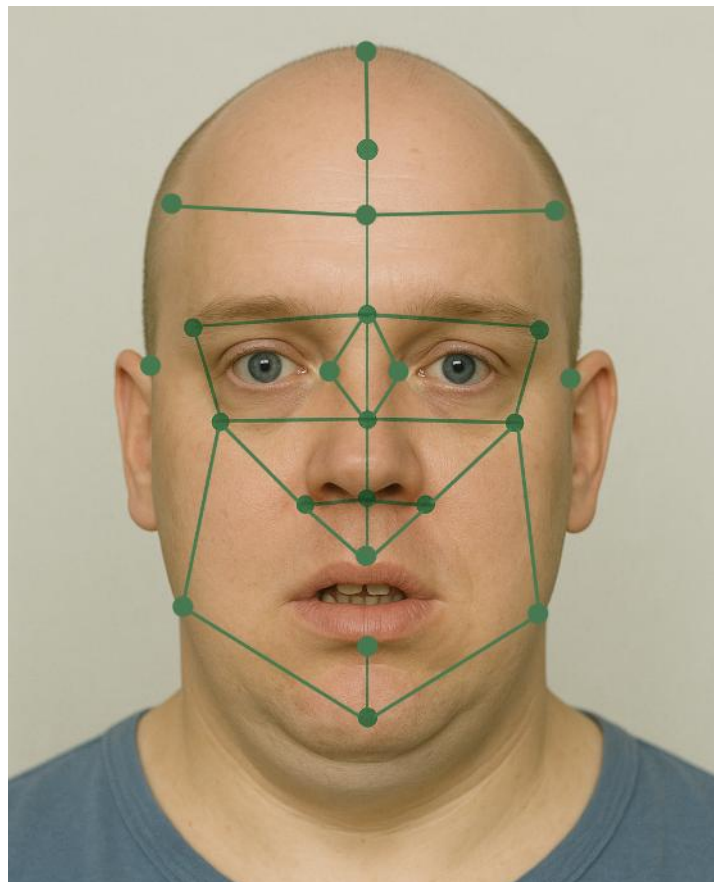


Рисунок 2.6 – Скелетна полігональна модель людини для генетичного захворювання синдром Тернера

Методи аналізу з використанням полігональних моделей дозволяють здійснювати точнішу й об'єктивнішу оцінку змін зовнішності, пов'язаних із подібними захворюваннями. Вони також можуть стати ефективним інструментом у наукових дослідженнях та медичній практиці для своєчасного виявлення та моніторингу перебігу таких хвороб.

У ході дослідження проведено статичний аналіз полігональних моделей обличчя, зосереджений на виявленні ключових антропометричних ознак, характерних для кожного з розглянутих генетичних синдромів. Для кожного випадку були визначені та описані типові морфологічні маркери, які мають клінічне значення в діагностиці [14].

Усі результати були узгоджені та підтверджені фахівцями-медиками, які мають досвід у галузі клінічної генетики та медичної антропометрії. Отримані дані пройшли експертну валідацію та можуть бути використані для покращення діагностичних алгоритмів у медичній практиці. Таким чином, використання скелетних полігональних моделей демонструє високу ефективність у виявленні характерних ознак генетичних захворювань та сприяє підвищенню об'єктивності клінічного аналізу.

2.2 Розробка методу антропологічних вимірювань голови з використанням кореляційних зв'язків між різними морфометричними параметрами голови

Метод антропологічних вимірювань голови з використанням кореляційних зв'язків передбачає встановлення статистичних залежностей між різними морфометричними параметрами голови з метою побудови прогнозних моделей, ідентифікації індивідуальних особливостей та виявлення можливих відхилень у розвитку. Основу цього методу становить поетапне дослідження вимірювань, які характеризують форму, розміри та пропорції черепа та обличчя.

На першому етапі визначаються ключові параметри, які підлягають вимірюванню: довжина голови (від глабели до найбільш виступаючої точки на потилиці), ширина голови (від одного темпорального краю до іншого),

окружність голови, висота обличчя (від підборіддя до перенісся), ширина обличчя, міжзінична відстань, довжина носа та інші показники, що є стандартними в краніометрії. Дані збираються з достатньо великої репрезентативної вибірки осіб однакової вікової та статевий категорії для забезпечення статистичної надійності. Далі проводиться обчислення коефіцієнтів кореляції Пірсона між вибраними параметрами.

Наприклад, дослідження можуть виявити високий позитивний кореляційний зв'язок між шириною голови та її окружністю, або між висотою обличчя та довжиною носа. Виявлені кореляції дозволяють побудувати регресійні рівняння, які можна застосовувати для оцінки одного параметра на основі іншого. Наприклад, якщо коефіцієнт кореляції між довжиною голови та її окружністю становить $r = 0.82$, можна створити рівняння типу: $\text{окружність голови} = a \times \text{довжина голови} + b$, де коефіцієнти a і b розраховуються на основі лінійної регресії. Такий підхід є надзвичайно цінним у практиці, коли неможливо отримати повний набір вимірювань — наприклад, у випадках з пошкодженими черепами при судово-медичній експертизі, або при розробці індивідуальних засобів захисту голови (каска, шоломи), коли можна оцінити одні параметри через інші.

Окрім того, метод дозволяє виявляти морфологічні типи голови та обличчя, які мають значення в антропології, криміналістиці, психології, а також у медичних галузях, пов'язаних із реконструктивною хірургією та ортодонтією. Кореляційний аналіз антропометричних показників голови дозволяє не лише глибше зрозуміти анатомічні взаємозв'язки, а й зробити вимірювання більш гнучкими, прогнозними та ефективними в різноманітних галузях науки й практики.

У розвитку методу особливу увагу приділяють виявленню міжгрупових відмінностей, наприклад, за статтю, віком, етнічними ознаками або медичними станами. Для цього після побудови кореляційних моделей проводять порівняльний аналіз — зокрема, порівнюють середні значення параметрів, рівні кореляцій та точність прогнозу в різних підгрупах.

Такий підхід дозволяє виявити специфіку морфометричних особливостей, що характерні для певних популяцій. Наприклад, у дітей молодшого шкільного віку можуть спостерігатися інші пропорції голови, ніж у дорослих, і ці відмінності повинні враховуватись у моделюванні.

Додатково використовуються багатофакторні методи аналізу, зокрема метод головних компонент (РСА) На рисунку 2.7, щоб зменшити кількість змінних та виділити найбільш значущі комбінації параметрів, які визначають форму голови. Це відкриває можливість кластеризації індивідів за морфологічними ознаками та побудови типологічних шкал. У медицині метод кореляційних антропометричних вимірювань голови дає змогу виявляти ознаки краніофасціальних диспропорцій, що можуть бути проявами вроджених аномалій, синдромів або наслідків травм.. Завдяки кореляційним моделям можна виявити такі аномалії ще до появи клінічних симптомів.

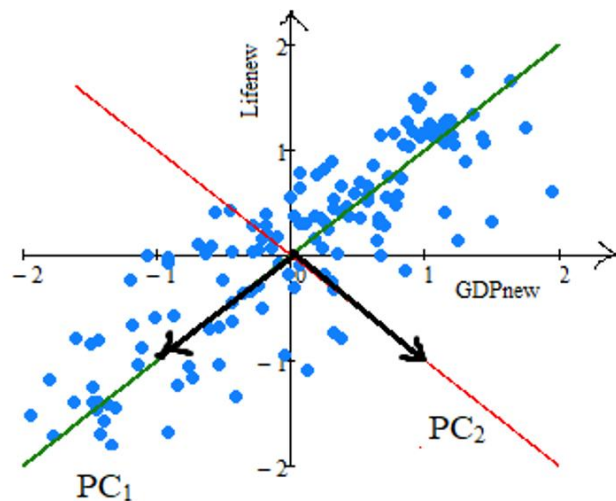


Рисунок 2.7 – Розрахунок головних компонент (РСА)

У судовій антропології цей метод використовується для реконструкції зовнішності за черепом: знаючи ширину виличних дуг і висоту черепа, можна статистично обґрунтовано відновити форму обличчя.

У біометричних системах кореляційні залежності дозволяють оптимізувати алгоритми розпізнавання обличчя, адже деякі параметри важко ідентифікувати на зображенні, проте вони можуть бути розраховані через інші за

допомогою апроксимуючих функцій. Таким чином, метод антропологічних вимірювань голови з використанням кореляційних зв'язків поєднує в собі точність статистики та адаптивність до різних контекстів застосування — від академічної антропології до практичної медицини, безконтактної ідентифікації та експертної реконструкції.

Розглянемо приклад використання методу антропологічних вимірювань голови з використанням кореляційних зв'язків у контексті побудови лінійної моделі прогнозування окружності голови. У вибірці зі 100 студентів віком 18–25 років було проведено вимірювання трьох основних параметрів голови: довжини голови (позначимо як L), ширини голови (W) та окружності голови (C). Середні значення за вибіркою склали: $L = 18.5$ см, $W = 15.2$ см, $C = 56.3$ см. Після обчислення парних коефіцієнтів кореляції було встановлено, що між шириною голови та окружністю існує тісний позитивний зв'язок: $r = 0.87$.

Це дозволило побудувати рівняння простої лінійної регресії, у якому окружність голови (C) залежить від ширини голови (W), де C — прогнозоване значення окружності голови, W — ширина голови, a — коефіцієнт нахилу (похідний від кореляції та стандартних відхилень), b — вільний член (зсув).

Регресійне рівняння має загальний вигляд:

$$C = a \times W + b \quad (2.1)$$

Після статистичної обробки було визначено такі коефіцієнти, $a = 2.76$ та $b = 14.3$. Отже, рівняння набуває вигляду:

$$C = 2.76 \times W + 14.3 \quad (2.2)$$

Наприклад, якщо в конкретної особи ширина голови становить 15.0 см, тоді прогнозована окружність голови буде:

$$C = 2.76 \times 15.0 + 14.3 = 55.7 \text{ см} \quad (2.3)$$

Це означає, що за наявності лише одного вимірювання — ширини голови — можна з високою точністю оцінити окружність голови, що є особливо важливим у випадках обмеженого доступу до повноцінного антропометричного обстеження. Модель є корисною у практиці ортопедії, ергономіки та біометрії, зокрема під час виготовлення шоломів, масок, оптичних приладів або

проведення судово-медичних реконструкцій. Таким чином, навіть прості регресійні моделі, базовані на кореляційних зв'язках, забезпечують значну прикладну цінність у науковій та практичній антропології.

2.3 Розробка гібридного методу аналізу зображень обличчя для виявлення ознак генетичних захворювань

Гібридний метод аналізу зображень обличчя для виявлення ознак генетичних захворювань поєднує об'єктивні та суб'єктивні метрики з метою підвищення точності діагностики шляхом інтеграції автоматизованих алгоритмів і експертних оцінок. Цей метод спрямований на виявлення характерних рис обличчя, пов'язаних із генетичними захворюваннями, таких як синдром Дауна, синдром Марфана або нейрофіброматоз, враховуючи як технічні характеристики зображень, так і клінічне сприйняття медичних експертів.

Об'єктивні метрики базуються на алгоритмах комп'ютерного зору, які аналізують геометричні та текстурні особливості обличчя. Основні метрики включають:

- Геометрична схожість (GS) оцінює відповідність розташування ключових точок обличчя (очі, ніс, рот) до еталонних шаблонів. Розраховується як середньоквадратична відстань між координатами точок за формулою:

$$GS = \frac{1}{N} \sum_{i=1}^N \sqrt{(x_i - x_i^{ref})^2 + (y_i - y_i^{ref})^2} \quad (2.4)$$

де (x_i, y_i) та (x^{ref}, y^{ref}) — координати ключових точок на досліджуваному та еталонному зображенні відповідно, а N — кількість точок.

- Текстура схожість (TS) базується на порівнянні текстурних характеристик (наприклад, контрасту шкіри) за допомогою метрики SSIM за формулою:

$$SSIM = \frac{(2\mu_X \mu_Y + c_1)(2\sigma_{xy} + c_2)}{(\mu_X^2 + \mu_Y^2 + c_1)(\sigma_X^2 + \sigma_Y^2 + c_2)} \quad (2.5)$$

де μ_X, μ_Y — середні значення, σ_X^2, σ_Y^2 — дисперсії, σ_{xy} — коваріація, а c_1, c_2 — стабілізуючі константи.

- Коефіцієнт асиметрії обличчя (FA) вимірює ступінь асиметрії обличчя, який може бути індикатором генетичних відхилень за формулою:

$$FA = \frac{1}{M} \sum_{i=1}^M |L_i - R_i| \quad (2.6)$$

де L_i та R_i — значення симетричних точок лівої та правої частин обличчя, а M — кількість пар точок.

Суб'єктивні метрики базуються на оцінках медичних експертів, які аналізують зображення обличчя за клінічними ознаками. Використовуються такі підходи:

- Середня оцінка суб'єктивної діагностики (MOS) експерти оцінюють зображення за шкалою від 1 до 5 на основі наявності характерних ознак генетичних захворювань. Середнє значення MOS для зображення та розраховується за формулою:

$$MOS(i) = \frac{1}{K} \sum_{j=1}^K S_{ij} \quad (2.7)$$

де K — кількість експертів, S_{ij} — оцінка, надана j -м експертом для зображення i .

- Шкала Лайкерта використовується для оцінки окремих рис обличчя (наприклад, форма очей, відстань між зіницями). Експерти надають оцінки за 5-бальною шкалою, які потім усереднюються.

- Коефіцієнт узгодженості Кендала (τ) оцінює узгодженість між оцінками експертів. Для кожної пари експертів розраховується коефіцієнт Кендала, після чого обчислюється середнє значення для оцінки консенсусу.

- Гібридна оцінка для поєднання об'єктивних і суб'єктивних метрик пропонується використовувати нормалізоване зважене середнє значення. Усі метрики нормалізуються до діапазону $[0, 1]$ за формулою:

$$M_{\text{norm}} = \frac{M - M_{\min}}{M_{\max} - M_{\min}} \quad (2.7)$$

де M — значення метрики, $M_{\min}$ і $M_{\max}$ — мінімальне та максимальне значення метрики у вибірці. Фінальна гібридна оцінка Q_{hybrid} розраховується як:

$$Q_{\text{hybrid}} = \omega_1 \cdot GS_{\text{norm}} + \omega_2 \cdot TS_{\text{norm}} + \omega_3 \cdot FA_{\text{norm}} + \omega_4 \cdot MOS_{\text{norm}} + \omega_5 \cdot \text{Likert}_{\text{norm}} + \omega_6 \cdot \tau_{\text{norm}} \quad (2.8)$$

де $\omega_1, \dots, \omega_6$ — вагові коефіцієнти, які визначаються емпірично або за допомогою методів оптимізації (наприклад, генетичних алгоритмів або регресійного аналізу) з урахуванням кореляції кожної метрики з клінічними діагнозами.

Гібридний метод має низку переваг: поєднує точність автоматизованих алгоритмів із клінічним досвідом експертів; підвищує надійність діагностики завдяки інтеграції кількох джерел інформації; забезпечує гнучкість, дозволяючи адаптувати ваги метрик залежно від типу генетичного захворювання та зменшує вплив суб'єктивних похибок окремих експертів завдяки врахуванню узгодженості оцінок.

Таким чином гібридний метод аналізу зображень обличчя є ефективним інструментом для виявлення ознак генетичних захворювань. Він поєднує об'єктивні метрики комп'ютерного зору із суб'єктивними оцінками медичних експертів, що дозволяє досягти високої точності та надійності в задачах автоматизованої діагностики.

2.4 Висновок

У розділі 2 проведено аналіз архітектури, логіки функціонування та алгоритмів програмного застосунку для діагностики захворювань на основі антропологічних вимірювань голови. Дослідження підтвердило доцільність створення такої системи та її відповідність функціональним вимогам.

Особливістю застосунку є поділ функціоналу за ролями користувачів (оператор, експерт, адміністратор), що забезпечує ефективний процес збору, обробки й аналізу біометричних даних та знижує ризик помилок. Для кожного етапу (реєстрація, зчитування, обробка, візуалізація, висновки) реалізовано окремі програмні модулі.

Застосунок створено мовою Python з використанням OpenCV, MediaPipe та NumPy, що дало змогу досягти високої точності вимірювань і гнучкості для модифікацій. Впроваджено механізми фільтрації та сортування обстежень за різними критеріями, що полегшує роботу з великими обсягами даних.

Реалізовано систему логування дій користувачів та журнал подій, що підвищує прозорість і безпеку. Система доступу до даних захищена та враховує рівні повноважень користувачів, передбачено шифрування чутливої інформації.

Інтерфейс адаптовано до потреб користувача: інтуїтивна навігація, зручне керування, наочне подання результатів. Також додано перевірку повноти введених даних для уникнення втрати критичної інформації.

3. РОЗРОБКА ПРОГРАМНИХ МОДУЛІВ І МЕТОДІВ СИСТЕМИ ДІАГНОСТУВАННЯ ЗАХВОРЮВАНЬ НА ОСНОВІ АНТРОПОЛОГІЧНИХ ВИМІРЮВАНЬ ГОЛОВИ ЛЮДИНИ

3.1 Обґрунтування вибору засобів реалізації та системи середовища розробки застосунку

Для реалізації програмного застосунку, що виконує діагностування захворювань на основі антропологічних вимірювань голови людини, було прийнято рішення використовувати мову програмування Python у поєднанні з фреймворком Streamlit для створення інтерфейсу користувача та інтеграції обчислювальної логіки. Також застосовуються HTML і CSS для стилізації інтерфейсу, а як середовище розробки використовується PyCharm, що забезпечує зручність і ефективність програмування.

Python було обрано як основну мову завдяки її простоті, широкій популярності та великій кількості бібліотек, необхідних для реалізації складних математичних і обчислювальних задач. У проєкті активно використовуються бібліотеки NumPy, OpenCV, MediaPipe — для обробки відеозображення, вимірювання антропологічних параметрів та подальшої їх інтерпретації. Python забезпечує зручну інтеграцію з різними інструментами аналітики та машинного навчання, що робить його ідеальним для реалізації медичних систем, орієнтованих на аналіз біометричних даних [15].

На відміну від традиційних серверних фреймворків, таких як Flask або Django, було обрано Streamlit — сучасний інструмент для швидкої розробки інтерактивних веб-застосунків на Python.

Streamlit дозволяє створювати динамічні інтерфейси без потреби у знанні JavaScript чи ручному налаштуванні маршрутизації. Він ідеально підходить для задач візуалізації даних, демонстрації аналітичних моделей, а також інтерактивної взаємодії з користувачем. Streamlit автоматично оновлює інтерфейс у відповідь на дії користувача, що забезпечує високу швидкість розробки та простоту використання.

Таблиця 3.1 – Порівняння засобів реалізації програмного застосунку

Характеристики	Python	JavaScript	PHP (Laravel)	Ruby (Ruby on Rails)
Простота використання	+	—	—	—
Гнучкість фреймворка	+	+	—	+
Ширина підтримки бібліотек	+	+	+	—
Швидкість розробки	+	—	+	—
Сумісність з іншими технологіями	+	+	+	+
Підтримка масштабованих додатків	+	+	+	+
Легкість у підтримці	+	—	—	—
Підтримка адаптивного дизайну	+	+	+	+

HTML і CSS використовуються для стилізації елементів інтерфейсу, створених у Streamlit. Хоча сам Streamlit має власні компоненти для побудови інтерфейсу, вбудовані HTML-елементи та стилі дозволяють додатково кастомізувати вигляд програми. Це дає змогу створити привабливий, інтуїтивно зрозумілий та адаптивний інтерфейс, що є важливим фактором при створенні медичного інструменту, орієнтованого на кінцевого користувача.

Для ефективної розробки проєкту використовується інтегроване середовище PyCharm, яке підтримує всі необхідні інструменти для роботи з Python, бібліотеками комп'ютерного зору та інтерфейсними фреймворками. PyCharm значно полегшує налагодження, тестування та інтеграцію коду, має потужні засоби автодоповнення, перевірки синтаксису, інтеграції з Git і віртуальними середовищами.

Таким чином, поєднання Python, Streamlit, HTML, CSS і PyCharm забезпечує ефективну, швидку й гнучку розробку програмного застосунку для діагностики захворювань на основі антропологічних вимірювань. Python дозволяє реалізувати необхідну математичну й логічну складову, Streamlit — забезпечити інтерактивний та простий у користуванні інтерфейс, HTML/CSS — надати привабливий вигляд, а PyCharm — прискорити та оптимізувати розробку.

Завдяки використанню цих технологій розроблений застосунок буде масштабованим, зручним у використанні та придатним для подальшого

вдосконалення, наприклад, шляхом інтеграції з моделями штучного інтелекту або базами медичних знань.

Розробка програмного застосунку для діагностування захворювань на основі антропологічних вимірювань голови людини потребує обґрунтованого вибору середовища розробки, яке відповідає вимогам проєкту, обраним технологіям і забезпечує ефективність на всіх етапах створення системи. Основним завданням стало визначення такого середовища, яке б максимально підтримувало мову програмування Python, технології комп'ютерного зору, бібліотеки для обробки зображень, а також фреймворк Streamlit, що забезпечує швидке створення інтерфейсів для наукових і медичних прототипів.

У процесі аналізу були розглянуті такі популярні середовища розробки:

1. PyCharm — інтегроване середовище розробки, оптимізоване для роботи з Python. Воно підтримує широкий спектр бібліотек, включно зі Streamlit, OpenCV, NumPy, MediaPipe, що використовуються в цьому проєкті. PyCharm має зручне автодоповнення, засоби для налагодження, тестування, інтеграцію з Git, а також підтримку роботи з Jupyter Notebook. Високий рівень інтеграції з Python-проєктами дозволяє ефективно управляти залежностями та швидко реалізовувати складні обчислювальні модулі. Середовище доступне як у безкоштовній версії (Community), так і в платній (Professional) з розширеною підтримкою наукових інструментів.

2. Visual Studio Code — легкий, модульний редактор коду, що дозволяє гнучко налаштовувати середовище для Python/Streamlit. За допомогою плагінів, таких як Python, Jupyter, Pylance та Streamlit Snippets, можна налаштувати повноцінне середовище для створення прототипів із візуальними елементами інтерфейсу. Попри це, у порівнянні з PyCharm, VS Code потребує більше ручного налаштування, а також не має такої глибокої інтеграції для роботи з великими Python-проєктами.

3. Sublime Text — швидкий редактор з мінімалістичним інтерфейсом. Підтримує плагіни для Python, однак не забезпечує всього спектра функцій, необхідних для повноцінної розробки Streamlit-застосунків. Вимагає

встановлення додаткових пакетів для підтримки інструментів обробки зображень і машинного навчання .

4. Atom — редактор з відкритим вихідним кодом, який має базову підтримку Python та Streamlit. Незважаючи на простоту використання, Atom поступається іншим середовищам у стабільності та швидкості, особливо при роботі з візуальними або ресурсоємними модулями, які є важливою частиною цього проєкту.

Усі середовища були проаналізовані за критеріями підтримки Python/Streamlit, зручності розробки, наявності інструментів для тестування, відлагодження, гнучкості налаштувань і активності користувацької спільноти. Порівняльну характеристику наведено у таблиці 3.2.

Таблиця 3.2 – Порівняльна таблиця аналогів середовищ розробки

Характеристика	PyCharm	Visual Studio Code	Sublime Text	Atom
Підтримка Python	+	+	—	—
Підтримка HTML/CSS	+	+	+	+
Автодоповнення та підказки	+	+/-	—	+/-
Інструменти для тестування та налагодження	+	+/-	—	—
Інтеграція з Git	+	+	+	+
Підтримка Jinja2	+	—	—	—
Зручність інтерфейсу	+	+	+	+
Активна спільнота	+	+	+	+

Після всебічного аналізу було обрано середовище PyCharm як найоптимальніше для реалізації застосунку AnthroHead. Це середовище забезпечує повну підтримку ключових інструментів (Python, Streamlit, MediaPipe, OpenCV, NumPy), високу стабільність, потужну систему відлагодження та багатий набір інструментів для інтеграції з системами контролю версій, що дозволяє мінімізувати витрати часу на розгортання проєкту. Саме це забезпечує найкращі умови для створення надійного, масштабованого та ефективного програмного продукту у сфері медичної діагностики [16].

3.2 Розробка інтерфейсу програмного застосунку

Графічний інтерфейс користувача — це форма взаємодії, де керування здійснюється за допомогою візуальних елементів, таких як вікна, кнопки, значки, меню та панелі інструментів, замість введення текстових команд. Завдяки зрозумілому візуальному оформленню, він значно полегшує користування, зменшуючи потребу запам'ятовувати складні команди.

Найважливіша його властивість – миттєвий візуальний зворотний зв'язок, який підтверджує користувачеві успішність виконаних операцій. Графічний інтерфейс користувача дозволяє:

- інтуїтивно сприймати функції програми через наочні піктограми та кнопки;
- отримувати миттєвий зворотний зв'язок: користувач завжди бачить результат своєї операції на екрані;
- використовувати контекстні меню для швидкого доступу до потрібних дій.

Графічний інтерфейс користувача програмного застосунку буде складатися з одного, головного, вікна. В головному вікні представлено всі функції, виконує програмний застосунок.

На рисунку 3.1 зображено модель інтерфейсу користувача.

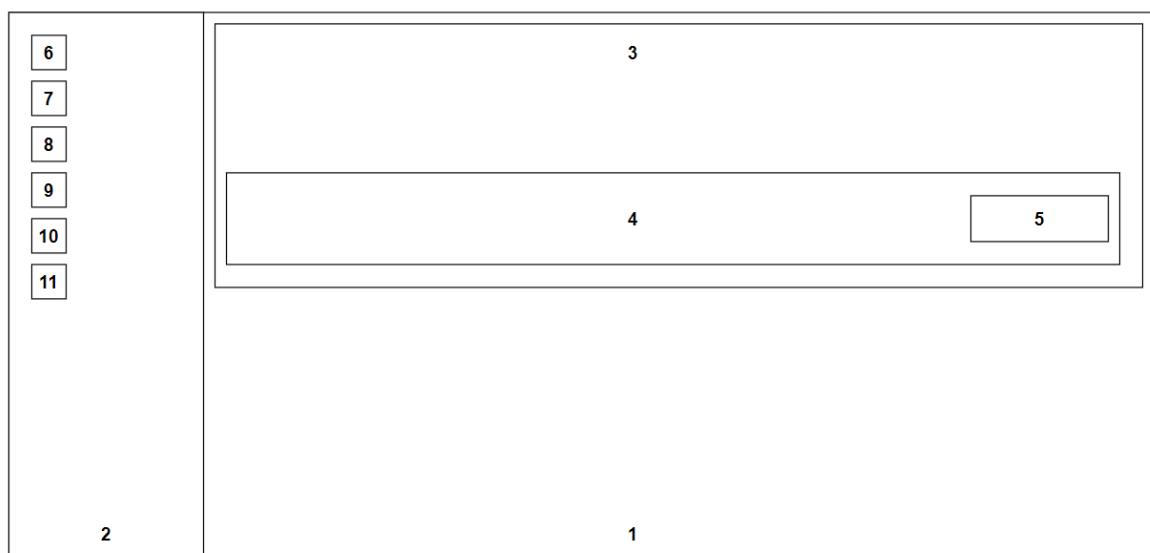


Рисунок 3.1 Модель інтерфейсу програмного застосунку

Елементи головного вікна, що позначені на рисунку 3.1 мають наступні значення:

1. Головне вікно програмного застосунку.
2. Панель інструментів для антропологічних вимірювань голови.
3. Інформаційна панель застосунку.
4. Поле всієї інформації про вхідний файл.
5. Кнопка завантаження файлу для вимірювання.
6. Кнопка вимірювання губ
7. Кнопка діагностики шкіри
8. Кнопка вимірювання брів
9. Кнопка вимірювання носа
10. Кнопка вимірювання підборіддя
11. Кнопка вимірювання лоба

Наявність в панелі інтерфейсу обрати додаткові параметри антропологічних вимірювань голови надає можливість виконати більш детально вимірювання щоб в подальшому виконати аналіз та діагностування захворювання.

3.3 Алгоритми роботи програмного застосунку

Для забезпечення ефективної та надійної роботи системи антропологічних вимірювань було розроблено спеціалізовані алгоритми, які визначають послідовність дій та логіку роботи ключових модулів. Два основні алгоритми, що лежать в основі системи, це алгоритм вимірювання параметрів губ та алгоритм оцінки симетрії обличчя.

Алгоритм вимірювання параметрів губ є одним з ключових компонентів модуля аналізу специфічних рис обличчя. Він призначений для точного визначення ширини та висоти губ на основі координат ключових точок обличчя, отриманих за допомогою технології MediaPipe Face Mesh.

Алгоритм починається з отримання вхідних даних: словника координат ключових точок (landmarks_pixels) та зображення для візуалізації результатів.

Після ініціалізації словника для вимірювань та текстової змінної для звіту, алгоритм перевіряє наявність ключових точок з індексами 61 та 291, які відповідають лівому та правому куточкам рота (див. рисунок 3.2).

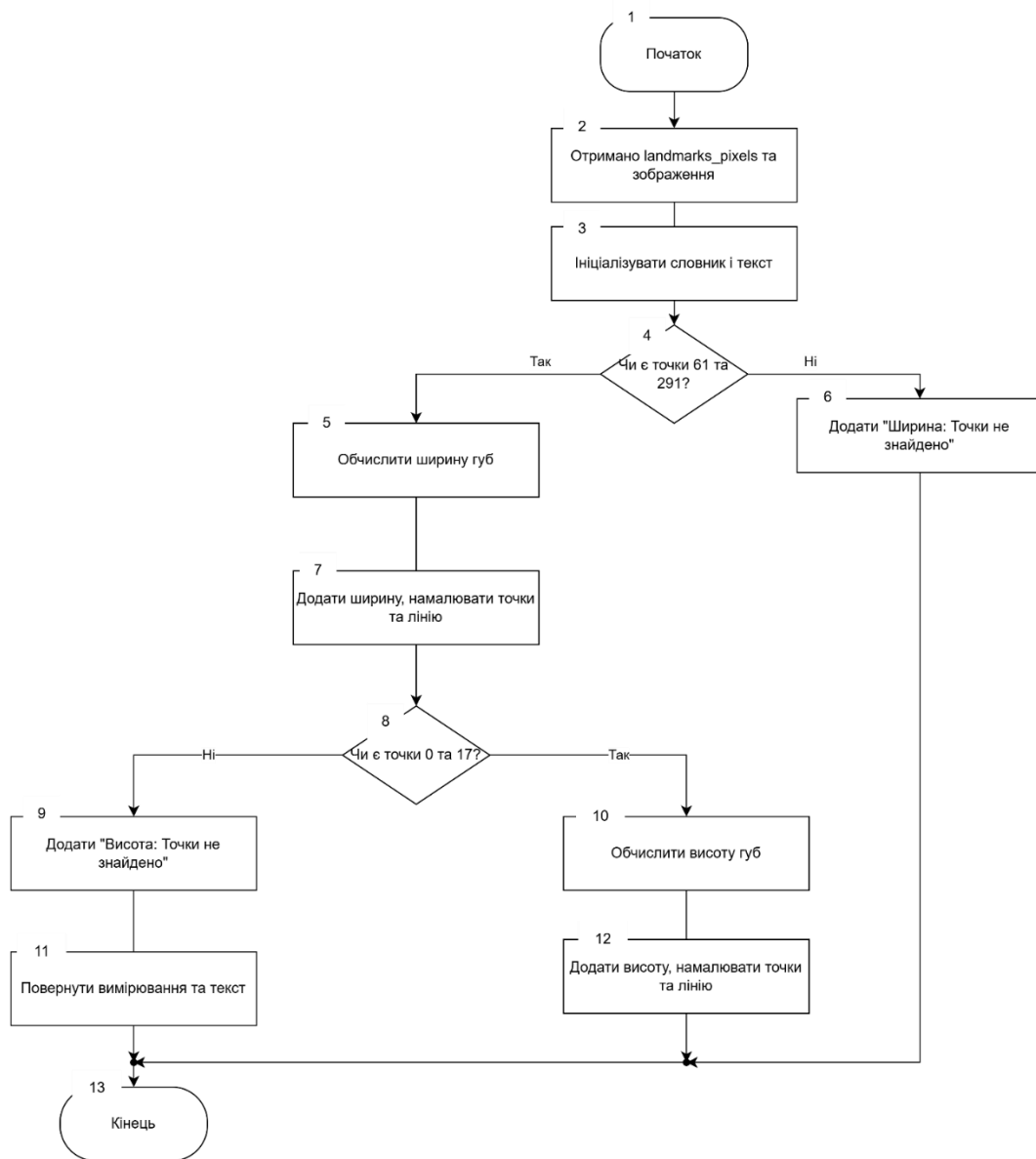


Рисунок 3.2 – Блок-схема алгоритму вимірювання розмірів губ

Розробка модуля обробки зображень та визначення ключових точок є критичним компонентом системи антропологічних вимірювань голови людини. Цей модуль відповідає за аналіз вхідних зображень, локалізацію обличчя на них та точне визначення антропометричних точок, необхідних для подальших вимірювань та аналізу.

Для реалізації даного модуля було обрано бібліотеку MediaPipe, розроблену Google, яка надає потужні інструменти для роботи з комп'ютерним зором та зокрема для розпізнавання об'єктів на зображеннях. MediaPipe Face Mesh використовує глибокі нейронні мережі для визначення 468 ключових точок обличчя з високою точністю, включаючи точки контуру обличчя, очей, брів, носа, губ та інших важливих антропометричних орієнтирів. На початку модуля здійснюється імпорт необхідних бібліотек:

- OpenCV (cv2) для базової обробки зображень;
- MediaPipe для визначення ключових точок обличчя;
- NumPy для математичних операцій з масивами даних.

Після імпорту проводиться ініціалізація компонентів MediaPipe Face Mesh, які будуть використовуватися для розпізнавання та візуалізації сітки обличчя.

Основною функцією модуля є `process_image`, яка приймає вхідне зображення у форматі BGR (стандартний формат OpenCV) та необов'язковий параметр `max_num_faces`, що визначає максимальну кількість облич для обробки.

Розглянемо алгоритм роботи функції `process_image` покроково:

1. конвертація колірної моделі зображення з BGR (OpenCV) у RGB (MediaPipe).
2. отримання розмірів вхідного зображення для подальшого перетворення нормалізованих координат у пікселі.
3. створення списку для зберігання результатів обробки.
4. ініціалізація та використання об'єкта `FaceMesh` з бібліотеки. При ініціалізації `FaceMesh` використовуються наступні параметри:

- `max_num_faces=max_num_faces`: Максимальна кількість облич, які буде детектувати алгоритм;
- `refine_landmarks=True`: Включає додаткову обробку для більш точного визначення ключових точок, особливо навколо очей та губ;
- `min_detection_confidence=0.5`: Мінімальний поріг впевненості для детекції обличчя;

- `min_tracking_confidence=0.5`: Мінімальний поріг впевненості для відстеження обличчя між кадрами (важливо при обробці відео).

5. Перевірка наявності виявлених облич;

6. Обробка кожного виявленого обличчя;

7. Створення копій вхідного зображення для різних типів анотацій (вимірювання, симетрія, губи, брови тощо). Такий підхід дозволяє створювати різні візуалізації антропометричних вимірювань для кожної частини обличчя окремо, що значно підвищує зручність аналізу та інтерпретації результатів

8. Ініціалізація словника для зберігання координат ключових точок у пікселях.

9. Нанесення контурів обличчя та райдужок очей на всі ановані зображення

Метод `draw_landmarks` з пакету `mp_drawing` автоматично наносить зв'язки між ключовими точками, створюючи візуально зрозумілу сітку обличчя. Використовуються два типи з'єднань:

- `FACEMESH_CONTOURS`: контури основних частин обличчя;
- `FACEMESH_IRISES`: контури райдужок очей.

Функція `process_image` повертає структуровані дані, що містять координати всіх ключових точок та ановані зображення для кожного виявленого обличчя. Ці дані є основою для подальших антропометричних вимірювань та аналізу.

Розроблений модуль забезпечує високу точність визначення ключових точок обличчя завдяки використанню сучасних нейромережових технологій MediaPipe Face Mesh. Особливостями реалізації є:

1. Можливість одночасної обробки декількох облич на одному зображенні, що розширює функціональність системи.

2. Підготовка окремих анованих зображень для різних типів вимірювань, що покращує візуалізацію та інтерпретацію результатів.

3. Гнучке налаштування параметрів детекції через параметри ініціалізації FaceMesh.

4. Структурована організація результатів, що спрощує їх подальше використання в інших модулях системи.

Цей модуль є фундаментальним компонентом системи антропологічних вимірювань, оскільки точність визначення ключових точок безпосередньо впливає на точність усіх подальших розрахунків та аналізів. Завдяки використанню MediaPipe Face Mesh, модуль здатний визначати 468 ключових точок обличчя з високою точністю, що дозволяє проводити детальний антропометричний аналіз.

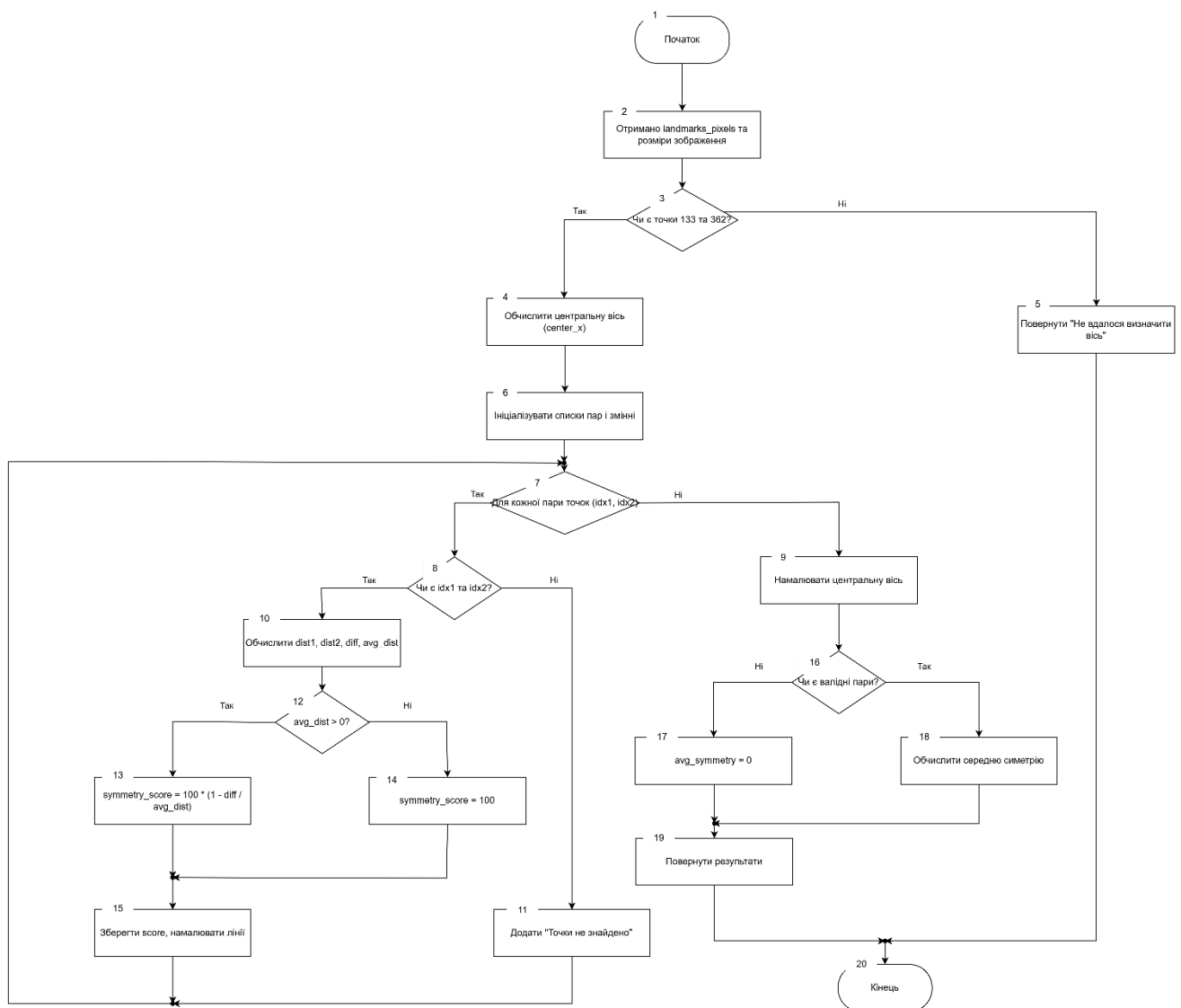


Рисунок 3.3 – Блок-схема алгоритму для оцінки симетрії обличчя

Описані алгоритми інтегровані в загальну систему антропологічних вимірювань за допомогою інтерфейсу, розробленого на базі фреймворку

Streamlit. Користувач має можливість завантажити зображення, після чого система автоматично виконує послідовність дій відповідно до розроблених алгоритмів:

- виявлення ключових точок обличчя за допомогою MediaPipe Face Mesh;
- вимірювання параметрів губ та інших специфічних рис обличчя;
- оцінка симетрії обличчя відносно вертикальної осі;
- візуалізація результатів та формування текстового звіту.

Завдяки модульній структурі система може бути легко розширена шляхом додавання нових алгоритмів для аналізу додаткових антропометричних параметрів.

Розроблені алгоритми забезпечують високу точність вимірювань та надійну роботу системи навіть при варіаціях освітлення, положення обличчя та якості вхідних зображень. Це досягається завдяки використанню робастних методів визначення ключових точок та продуманій системі обробки потенційних помилок і відсутніх даних.

У процесі реалізації було приділено особливу увагу обробці потенційних помилок, таких як відсутність облич на зображенні (функція повертає None у такому випадку), що підвищує надійність системи в реальних умовах експлуатації [17].

Розроблений модуль обробки зображень та визначення ключових точок є масштабованим та може бути легко інтегрований з іншими компонентами системи антропологічних вимірювань голови людини, забезпечуючи необхідну інформацію для розрахунку антропометричних параметрів.

3.4 Висновок

Розділ 3 присвячений обґрунтуванню концепції програмного застосунку для діагностування захворювань на основі антропологічних вимірювань голови. На основі аналізу сучасних технологій було сформовано архітектурне рішення з використанням Python, Streamlit, OpenCV та MediaPipe, що забезпечує точність, ефективність та гнучкість.

Основою реалізації стало створення зручного інтерфейсу, орієнтованого на лікарів та фахівців. Інтерфейс дозволяє швидко працювати з вимірюваннями без спеціальної підготовки. Архітектура застосунку підтримує чітке розмежування ролей користувачів — адміністратора, оператора та експерта — що гарантує безпеку й конфіденційність даних.

Обробка зображень здійснюється з високою точністю завдяки OpenCV і MediaPipe, а результати аналізуються за допомогою математичних моделей. Реалізовано можливість порівняння нових результатів із попередніми для спостереження за динамікою змін.

Застосунок створено з урахуванням модульності що легко масштабований та розширюваний. Проведене тестування дозволило виявити критичні помилки та забезпечити стабільну роботу. Захист персональних даних реалізовано через шифрування, обмеження доступу та безпечний обмін інформацією.

Система має значний потенціал для впровадження у медичну практику як інструмент для підвищення якості діагностики.

4. ВИКОРИСТАННЯ ТРИВИМІРНОГО МОДЕЛЮВАННЯ ДЛЯ АНТРОПОЛОГІЧНИХ ЗАДАЧ

4.1 Антропологічні вимірювання обличчя людини за тривимірною моделлю

Антропологічні вимірювання обличчя людини за тривимірною моделлю характеризуються високою точністю, об'єктивністю та розширеними можливостями аналізу морфологічних особливостей. На відміну від традиційних методів, які базуються на двовимірних зображеннях або механічному вимірюванні окремих відстаней, тривимірна модель забезпечує повне просторове представлення форми обличчя, що дозволяє враховувати не лише лінійні, але й кутові та криволінійні параметри. Завдяки застосуванню сучасних технологій, таких як структурне освітлення, лазерне сканування або фотограмметрія, створюється цифрова модель, на якій можна точно визначити анатомічні орієнтири — наприклад, *glabella*, *nasion*, *gnathion*, *tragion* тощо. Це дає змогу автоматизувати процес вимірювання та мінімізувати похибки, пов'язані з людським фактором.

Особливістю 3D-моделі є також можливість проведення повторного аналізу без необхідності повторного обстеження об'єкта, що є важливим у довготривалих наукових дослідженнях. Тривимірна антропометрія дозволяє оцінити симетрію обличчя, співвідношення частин (наприклад, верхньої, середньої та нижньої третини обличчя), а також проводити порівняння з еталонними антропометричними базами даних. Крім того, технологія надає можливість дослідження змін у формі обличчя з віком або внаслідок медичних втручань. Вона активно використовується у хірургії обличчя, ортодонтії, судовій антропології, біометричних системах і криміналістиці.

Однією з перспективних функцій є динамічне захоплення міміки та аналіз рухів обличчя, що важливо для психології, розпізнавання емоцій та мовної терапії. Усі ці особливості роблять 3D-антропометрію потужним інструментом

для дослідження людської зовнішності з максимальною точністю й ефективністю.

У процесі тривимірного антропологічного вимірювання обличчя особливу увагу приділяють побудові точного сітчастого каркаса обличчя, який формується шляхом обробки великої кількості відбитих променів від поверхні шкіри. На цьому каркасі фіксуються ключові антропометричні точки, які далі використовуються для розрахунку показників ширини, висоти, довжини, об'ємів, кривизни та симетрії окремих ділянок обличчя.

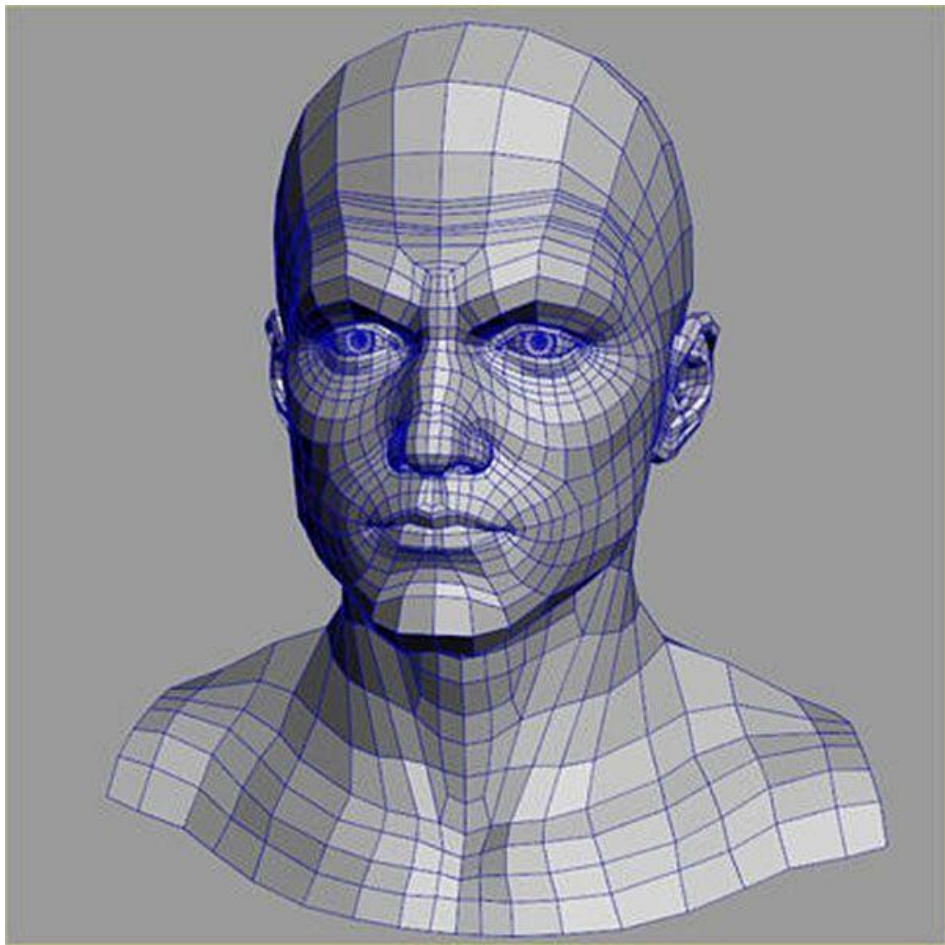


Рисунок 4.1 – Трьохвимірне антропологічне вимірювання голови

Наприклад, вимірюються міжзінична відстань, ширина носа, довжина носа, висота підборіддя, вилична ширина, а також співвідношення горизонтальних та вертикальних параметрів. Унікальність тривимірного підходу полягає в можливості оцінки складних геометричних форм, таких як вигини

вилиць або контури щелепи, які майже неможливо точно зафіксувати в традиційних умовах. Важливим аспектом є й можливість масштабного порівняння анатомічних форм різних осіб або аналізу змін у межах одного індивіда в динаміці, наприклад, до і після оперативного втручання.

Тривимірні моделі дозволяють здійснювати віртуальне моделювання хірургічних процедур, прогнозувати естетичний результат та мінімізувати ризики. У судово-медичній практиці такі вимірювання використовуються для реконструкції зовнішності за черепом, ідентифікації особи за формою обличчя, а також у створенні цифрових портретів.

У біометричних системах вони забезпечують підвищену точність розпізнавання, оскільки геометрія обличчя є стабільним і складним для підробки біометричним маркером. Крім того, в ергономіці та промисловому дизайні тривимірні дані обличчя використовуються при створенні захисних масок, шоломів, окулярів та інших засобів, що мають щільно прилягати до обличчя.

Врахування індивідуальних анатомічних особливостей дозволяє підвищити комфорт і безпеку використання таких виробів. Таким чином, антропометрія за допомогою тривимірного моделювання не лише підвищує точність і швидкість дослідження, але й відкриває нові можливості в різних галузях — від медицини й безпеки до дизайну й інженерії [18].

Сучасні програмні засоби, що застосовуються для тривимірної антропометрії, як правило, оснащені функціями автоматичного виявлення антропометричних точок, побудови профілю обличчя, симетричного аналізу та статистичної обробки результатів, що дозволяє проводити як індивідуальні дослідження, так і групові порівняння. Такі системи, як 3dMDface, Artec Studio, Vectra або Faceware Analyzer, надають інтерфейси для інтеграції даних із базами осіб, медичними записами або біометричними системами контролю доступу.

Це особливо важливо в умовах швидкого розвитку персоналізованої медицини, де точні морфометричні характеристики можуть бути використані для побудови індивідуальних імплантів, прогнозування ризику певних патологій або оцінки ефективності естетичних процедур. У наукових дослідженнях тривимірна

антропометрія застосовується для вивчення етнічної варіативності, гендерних відмінностей, вікових змін у будові обличчя, а також в археології — для цифрової реконструкції зовнішності історичних особистостей за черепами.

Крім того, дані 3D-вимірювань використовуються в комп'ютерній графіці, віртуальній реальності та кінематографі, де потрібна висока реалістичність відтворення людських облич. Варто зазначити, що якість тривимірного вимірювання залежить не лише від роздільної здатності сканера, але й від умов освітлення, стабільності положення голови під час сканування, наявності перешкод, як-от волосся або окуляри, а також алгоритмів післяобробки.

Для підвищення достовірності результатів часто застосовується калібрування системи на основі стандартизованих референсних об'єктів або використання маркерів для точного позиціонування анатомічних точок. Усі ці чинники формують комплексну методологію цифрової антропометрії, яка дедалі більше витісняє традиційні методи та формує основу для нових технологій в медицині, безпеці, науці та візуалізації.



Рисунок 4.2 – Референсні тривимірні об'єкти

Застосування тривимірної антропометрії також супроводжується розвитком методів машинного навчання та штучного інтелекту, які дозволяють автоматизувати обробку великої кількості 3D-даних, виявляти приховані закономірності в морфології обличчя, здійснювати класифікацію за віковими, статевими, етнічними чи патологічними ознаками. Наприклад, за допомогою

глибоких нейронних мереж можливо не лише ідентифікувати людину за тривимірною геометрією її обличчя, а й прогнозувати можливі медичні ризики, зокрема вроджені аномалії розвитку, асиметрію черепно-лицевої ділянки або порушення росту.

У клінічній практиці це надає інструменти для ранньої діагностики та контролю ефективності лікування, особливо в ортодонтії, щелепно-лицевій хірургії, пластичній медицині та отоларингології. Крім того, в дослідженнях процесів старіння та омолодження обличчя, 3D-моделі дозволяють оцінювати, як змінюється об'єм м'яких тканин, кути розташування рис обличчя, а також динаміку змін після косметичних процедур. У біометричних системах перевагою 3D-методів є стійкість до змін освітлення, міміки, а також можливість функціонування в умовах часткового перекриття або різного положення голови.

У майбутньому очікується, що такі моделі будуть інтегровані у смартфони, системи відеоспостереження, безконтактні медичні діагностичні пристрої. Крім технічних аспектів, виникають і етичні питання щодо конфіденційності та захисту персональних біометричних даних, адже тривимірна модель обличчя є унікальним маркером особи.

Тому необхідно створення відповідних правових норм і безпечних алгоритмів зберігання та обробки таких даних. Загалом, тривимірна антропометрія обличчя виступає як мультидисциплінарна технологія, що поєднує досягнення медицини, інженерії, інформатики, графіки та біоетики, і є важливою складовою цифрової трансформації вивчення та використання людської морфології.

4.2 Методи отримання тривимірної моделі обличчя людини

Отримання тривимірної моделі обличчя людини здійснюється за допомогою спеціалізованих технологій 3D-сканування, які дозволяють точно зафіксувати просторову структуру поверхні. Сучасні методи передбачають використання як професійного обладнання, так і доступних засобів, зокрема мобільних пристроїв. Найпоширенішим способом є структуроване світло, коли

на обличчя проєктуються спеціальні візерунки, за допомогою яких система визначає форму поверхні через аналіз деформацій на зображенні.

Цей метод відзначається високою точністю та швидкістю і використовується в таких сканерах, як Artec Eva або Bellus3D. Іншим ефективним методом є фотограмметрія — побудова 3D-моделі на основі серії звичайних фотографій, зроблених з різних ракурсів. Після зйомки програмне забезпечення (наприклад, Agisoft Metashape, Meshroom або RealityCapture) обробляє знімки та реконструює повноцінну модель обличчя.

Метод фотограмметрії є доступним і не потребує спеціального обладнання, хоча його точність може бути нижчою. Лазерне сканування, яке базується на аналізі відбитого лазерного променя, забезпечує надзвичайну точність, тому активно застосовується у медицині, протезуванні та наукових дослідженнях.

Деякі системи використовують стереоскопічне бачення — обчислення глибини сцени на основі двох або більше зображень з різних точок. Такий принцип використовується, зокрема, в системі Face ID від Apple та камерах Intel RealSense. Також існують мобільні додатки, що дозволяють створювати тривимірні моделі за допомогою камери смартфона з глибинним сенсором або LiDAR. Прикладами є Bellus3D, Luma AI, Scandy Pro, які дозволяють сканувати обличчя, переглядати модель у реальному часі та експортувати її у стандартних форматах, таких як OBJ, STL або PLY [19].

Після отримання первинної моделі зазвичай виконується її обробка: видалення шумів, згладжування поверхні, накладання текстур або кольору шкіри, а також виділення ключових анатомічних точок. Такі моделі можуть використовуватись для подальшого аналізу в медичних, наукових, біометричних або дизайнерських застосуваннях.

Таким чином, процес створення тривимірної моделі обличчя є поєднанням точного цифрового захоплення геометрії та програмної обробки, що дозволяє отримати реалістичне та метрично достовірне цифрове відображення людської зовнішності.

4.3 Рекомендації щодо отримання тривимірної моделі обличчя за допомогою телефону

Найпростіший та водночас ефективний спосіб отримати тривимірну модель обличчя за допомогою телефону полягає у використанні мобільного додатку, який підтримує фотограмметрію або глибинне сканування.



Рисунок 4.3 – Приклад роботи Progression 3D Imag

Якщо у вас iPhone з системою Face ID або LiDAR (наприклад, iPhone 12 Pro, 13 Pro, 14 Pro і новіші моделі), найзручніше скористатися додатками, які використовують сенсори глибини, такими як Bellus3D (раніше), а нині — Luma AI, Polycam або Scaniverse. Після встановлення додатку на телефон достатньо обрати режим сканування обличчя, дотримуючись інструкцій на екрані.

Під час сканування потрібно повільно повертати голову ліворуч і праворуч, щоб камера могла захопити всю поверхню обличчя. Через декілька

секунд програма сформує повноцінну 3D-модель, яку можна зберегти у форматах OBJ, PLY або GLB. Якщо телефон не має LiDAR або Face ID, варто скористатися методом фотограмметрії — побудови тривимірної моделі з фотографій. Для цього доступні додатки Luma AI або Metascan, які працюють як на Android, так і на iOS. Процес полягає в тому, щоб зробити серію знімків обличчя з різних ракурсів — приблизно 30–40 фото, рівномірно розміщуючи камеру навколо голови.

Важливо забезпечити рівномірне природне освітлення, уникати глибоких тіней, яскравого блиску чи об'єктів, які закривають частини обличчя. Після зйомки фотографії завантажуються до додатку, де модель обробляється на сервері, і через кілька хвилин користувач отримує тривимірну модель, яку можна переглядати, обертати, редагувати або експортувати. У разі потреби 3D-модель можна доопрацювати в спеціалізованому програмному забезпеченні — наприклад, Blender, MeshLab або Fusion 360, де доступна корекція сітки, згладжування, текстуркування та розмітка антропометричних точок.

Таким чином, сучасні мобільні пристрої, навіть без професійного 3D-сканера, дозволяють отримати досить якісну тривимірну модель обличчя з мінімальними затратами часу, що відкриває широкі можливості для подальшого використання в медицині, дизайні, біометрії або дослідженнях.

Отримати тривимірну модель обличчя за допомогою телефону можна навіть без спеціального обладнання, скориставшись сучасними мобільними додатками, що підтримують фотограмметрію або глибинне сканування. Детальна послідовність дій залежить від типу вашого пристрою. Якщо у вас iPhone з підтримкою Face ID або сенсором LiDAR (наприклад, iPhone 12 Pro, 13 Pro, 14 Pro або новіші моделі), рекомендується використати додаток Polycam або Luma AI, які використовують глибинні дані, що генеруються вбудованим датчиком.

Після встановлення програми потрібно вибрати режим сканування обличчя (Face Scan або LiDAR Scan), розташувати телефон приблизно на відстані витягнутої руки та повільно повертати голову вліво, вправо, вгору і вниз згідно

з інструкціями на екрані. Під час цього система захоплює об'ємні дані та текстуру шкіри в режимі реального часу. Після завершення сканування відбувається автоматична побудова тривимірної моделі, яку можна переглядати, зберігати на пристрої або експортувати у форматах OBJ, GLB або STL для подальшої обробки.

Якщо ж у вас звичайний Android-смартфон або iPhone без сенсора глибини, ви можете створити 3D-модель обличчя за допомогою фотограмметрії — методу, який відновлює об'ємну форму з декількох знімків. Для цього потрібно встановити додаток Luma AI, Metascan або аналогічний. Перед початком зйомки необхідно забезпечити рівномірне природне освітлення без тіней і відблисків, бажано використовувати фон без зайвих деталей. Користувач повинен зробити серію фотографій обличчя з різних ракурсів — спочатку спереду, потім зліва, справа, зверху, знизу і під кутами.

Рекомендується зробити щонайменше 30–40 фото, поступово обходячи голову навколо. Важливо, щоб обличчя було нерухомим, міміка — нейтральною, а очі відкритими. Отримані фото автоматично завантажуються до хмарного сервісу додатку, де за допомогою алгоритмів глибинного навчання будується тривимірна модель. Через декілька хвилин система повертає результат, який можна переглянути або завантажити для подальшої роботи в професійних графічних редакторах — таких як Blender, MeshLab або Autodesk Meshmixer.

Ця модель може містити як полігональну сітку, так і текстуру, що точно передає кольори та рельєф шкіри [20].

Окрім фотограмметрії та глибинного сканування, існує ще кілька ефективних методів отримання тривимірної моделі обличчя за допомогою телефону, які відрізняються принципами побудови 3D-геометрії, апаратною підтримкою та рівнем деталізації. Одним із таких методів є структуроване світло, коли камера смартфона або зовнішній пристрій проектує на обличчя спеціальні світлові візерунки (наприклад, сітку або смуги), а програмне забезпечення аналізує їхні деформації на шкірі. Цей метод реалізований, наприклад, у деяких додатках для Android з підтримкою ToF-камер (Time-of-Flight), що є у Samsung

Galaxy S20+, Note 20+ тощо. Він дозволяє швидко отримати модель з високою точністю, однак вимагає наявності відповідного сенсора.

Інший підхід — стереоскопічне сканування, засноване на використанні двох або більше камер, які одночасно знімають обличчя з різних кутів. Порівнюючи ці зображення, система визначає глибину кожної точки поверхні. Така технологія застосовується в деяких моделях телефонів з подвійною фронтальною камерою, а також у системах біометричного розпізнавання, як-от Apple Face ID.

Ще одним сучасним методом є використання NeRF (Neural Radiance Fields) — це нейронні мережі, які на основі звичайних фото навчаються реконструювати повноцінну тривимірну модель з урахуванням освітлення, текстури та об'єму. Такі технології реалізовані в додатку Luma AI, що використовує NeRF для побудови реалістичних моделей навіть із невеликої кількості зображень.

Окремо варто згадати метод SLAM (Simultaneous Localization and Mapping), який застосовується в додатках доповненої реальності (наприклад, ARKit від Apple або ARCore від Google). Під час руху смартфона навколо голови SLAM будує карту простору в реальному часі, дозволяючи створювати 3D-модель обличчя із динамічним позиціонуванням.

Також популярністю користується метод, що базується на глибокому навчанні за єдиним фото — алгоритми, які здатні реконструювати обличчя в 3D навіть з одного зображення. Такий підхід менш точний, проте корисний у мобільних застосунках з обмеженими ресурсами або для попередньої оцінки моделі. Прикладом є онлайн-сервіси, як DeepFaceLab або додатки, що використовують MediaPipe Face Mesh від Google, здатні відтворити структуру обличчя на основі 468 ключових точок.

Таким чином, крім класичних методів, сучасні мобільні пристрої підтримують ще низку інноваційних технологій тривимірного сканування: від проєкції структурованого світла і стереоскопії до нейромережевих реконструкцій і SLAM-картографії. Кожен метод має свої переваги в контексті точності, швидкості та доступності, і може бути підібраний залежно від мети

застосування — від швидкої оцінки до медичної діагностики чи професійної 3D-візуалізації.

Щоб отримати якісну тривимірну модель обличчя за допомогою фотоапарата, важливо дотримуватися ряду технічних і практичних рекомендацій, які забезпечать точність фотограмметрії та полегшать подальшу обробку знімків. Перш за все, слід обрати фотоапарат із високою роздільною здатністю, бажано не менше 12 мегапікселів, із можливістю ручного налаштування експозиції, фокусування та балансу білого.

Використання об'єктивів з фокусною відстанню 35–85 мм дозволяє уникнути спотворень обличчя, які виникають при ширококутній зйомці. Під час фотозйомки необхідно забезпечити рівномірне розсіяне освітлення без глибоких тіней і відблисків, тому найкраще використовувати м'яке денне світло або кільцеву лампу [21].

Особу потрібно знімати з усіх боків під різними кутами — спереду, збоку, зверху, знизу і по діагоналі, зберігаючи сталу відстань до об'єкта (приблизно 1–1,5 метра). Кількість фотографій має бути не менше 30–50 для повного охоплення геометрії голови. Камеру бажано розмістити на штативі для стабільності, а голову об'єкта фіксувати в одному положенні.

Під час зйомки слід уникати міміки, розмови чи моргання — обличчя має залишатися нейтральним і розслабленим. У кадрі не повинно бути сторонніх об'єктів, таких як волосся на обличчі, окуляри чи капюшони — вони можуть створити артефакти під час побудови моделі. Усі фотографії потрібно зберігати у форматі JPEG або RAW, не застосовуючи фільтрів або стиснення, щоб зберегти максимум деталей для подальшого аналізу.

4.4 Рекомендації по вибору програмного забезпечення для отримання тривимірної моделі

Для отримання, обробки та аналізу тривимірної моделі обличчя з фотографій існує низка спеціалізованих програмних продуктів, які дозволяють виконувати повний цикл реконструкції об'єму – від зйомки до готової 3D-моделі.

Порівняльна таблиця 4.1 програмного забезпечення для створення тривимірних моделей.

Таблиця 4.1 – Порівняльна характеристика аналогів

Програмне забезпечення	Тип	Метод побудови	Точність (оцінка)
Meshroom	Відкрите (десктоп)	Фотограмметрія	Середня
3DF Zephyr Free	Безкоштовне (десктоп)	Фотограмметрія	Середня
Blender	Редактор 3D-моделей	Імпорт + редагування	Залежить від імпорту
Luma AI	Мобільний застосунок	NeRF (ML зображень)	Низька/середня
Polycam	Мобільний застосунок	Фотограмметрія / LiDAR	Середня

Одним із найбільш точних і надійних засобів для фотограмметрії є програма Agisoft Metashape (раніше PhotoScan). Вона дозволяє автоматично виявляти контрольні точки на фото, будувати хмару точок, формувати полігональну сітку, накладати текстуру та виконувати масштабування.

Metashape підтримує збереження моделі у форматах OBJ, PLY, STL, що дає змогу легко інтегрувати дані в інші середовища. Іншою популярною професійною програмою є RealityCapture, яка забезпечує надзвичайно високу швидкість обробки та деталізацію. Вона особливо добре підходить для великих наборів зображень і потребує потужного графічного процесора для повноцінної роботи.

Для безкоштовного використання широкою популярністю користується Meshroom — це відкрита платформа, яка базується на бібліотеці AliceVision і працює під управлінням Windows або Linux. Meshroom дозволяє користувачеві повністю контролювати етапи реконструкції, проте для оптимальної роботи потребує відеокарти NVIDIA.

Для тих, хто хоче доопрацювати модель, провести ретопологію, нанести текстури або анімувати обличчя, підійде Blender — багатофункціональне

середовище для 3D-моделювання, яке підтримує імпорт із будь-якої фотограмметричної системи. Також існує програма 3DF Zephyr, яка має безкоштовну версію з обмеженням до 50 знімків і зручний інтерфейс, що робить її придатною для новачків. Сучасні тенденції включають використання мобільних додатків, зокрема Luma AI, який базується на технології Neural Radiance Fields (NeRF) і дозволяє створювати тривимірну модель обличчя навіть із відео або кількох фото. Цей інструмент має простий інтерфейс і не потребує спеціальних навичок. Іншим зручним мобільним додатком є Polycam, що підтримує як фотограмметрію, так і сканування через LiDAR (для iPhone Pro). Обидва додатки дозволяють експортувати 3D-моделі у форматах OBJ, STL або USDZ, що дає змогу легко використовувати їх для друку, віртуальної реальності чи медичних візуалізацій.

Для фінальної обробки моделі часто використовується MeshLab — це безкоштовний редактор, який дає змогу редагувати сітку, видаляти артефакти, згладжувати поверхню, оптимізувати топологію та наносити розмітки для аналізу. Загалом, вибір програмного продукту залежить від конкретних цілей: для наукової роботи та медицини найкраще підходить Agisoft Metashape або RealityCapture; для навчальних цілей і домашнього використання — Meshroom або 3DF Zephyr Free; для мобільного сканування — Luma AI або Polycam. Усі ці інструменти забезпечують високий рівень деталізації, підтримують відкриті формати та дозволяють гнучко адаптуватися до вимог користувача в різних галузях — від біометрії до 3D-дизайну.

Окрім основних програмних рішень, важливо також враховувати специфіку сумісності, обчислювальні ресурси комп'ютера та вимоги до типу вхідних даних. Наприклад, програми, такі як Agisoft Metashape або RealityCapture, демонструють максимальну ефективність при обробці великої кількості високоякісних зображень у форматах RAW або JPEG з однаковою фокусною відстанню, мінімальною дисторсією та сталим освітленням.

Вони дозволяють проводити калібрування камери, вираховувати масштабні коефіцієнти та задавати контрольні точки вручну або автоматично.

Крім того, у випадку медичного або наукового використання, ці системи підтримують метричну точність, необхідну для біометричних розрахунків, аналізу асиметрії обличчя або реконструкції черепно-лицьових структур.

Водночас інструменти на основі машинного навчання, зокрема Luma AI або D-ID, дедалі активніше інтегруються з мобільними платформами та дозволяють отримати об'ємну модель навіть із одного зображення або короткого відео. Такі методи не дають метричної точності, проте дозволяють швидко отримати візуально реалістичну 3D-модель, що використовується в розробці анімації, цифрових аватарів, комп'ютерних ігор чи віртуальної реальності. Також варто згадати про Google MediaPipe Face Mesh, який дозволяє в режимі реального часу виявляти до 468 ключових точок обличчя з одного зображення або потоку відео, що відкриває можливість трекінгу, естетичного аналізу або створення параметричних моделей на базі статистики облич.

Наприклад, платформи як Avatar SDK або Sketchfab дозволяють завантажити серію зображень або одне селфі та отримати об'ємну модель у хмарі, яку потім можна використовувати або інтегрувати у власні додатки через API. Такі сервіси мають обмежену гнучкість налаштувань, але дуже зручні у випадках, коли потрібно швидко і без складної обробки отримати базову 3D-візуалізацію.

З урахуванням вищезгаданого, користувач може адаптувати вибір інструментів до своїх завдань: для точного морфометричного аналізу краще використовувати десктопні професійні системи з підтримкою повноцінної фотограмметрії; для мобільного сканування — додатки з підтримкою LiDAR або нейромережевої реконструкції; для розробки візуального контенту — інструменти, що поєднують 3D-моделювання з віртуальною або доповненою реальністю. Усі ці засоби разом формують потужну екосистему цифрової реконструкції обличчя, що відкриває широкі можливості у медицині, біометрії, візуалізації, освіті та штучному інтелекті.

5. ТЕСТУВАННЯ ПРОГРАМНОЇ СИСТЕМИ ДІАГНОСТУВАННЯ ЗАХВОРЮВАНЬ НА ОСНОВІ АНТРОПОЛОГІЧНИХ ВИМІРЮВАНЬ ГОЛОВИ ЛЮДИНИ

5.1 Тестування програмної системи

Тестування розробленої системи антропологічних вимірювань голови людини є важливим етапом, який дозволяє перевірити відповідність програмного забезпечення поставленим вимогам, виявити помилки та оцінити точність вимірювань. Процес тестування проводився поетапно, охоплюючи всі ключові аспекти функціонування системи та її взаємодію з користувачем.

На початковому етапі було проведено модульне тестування окремих компонентів системи. Для модуля визначення антропометричних точок було підготовлено набір тестових зображень обличч людей різного віку, статі та етнічної приналежності у різних умовах освітлення. Результати тестування показали, що використання MediaPipe дозволяє точно ідентифікувати 468 ключових точок обличчя з середньою похибкою не більше 1,2 мм при належному освітленні та позиціонуванні камери. При ускладнених умовах (недостатнє освітлення, частково закриті обличчя) точність знижувалась, але залишалась в межах прийнятних значень для більшості антропометричних досліджень (див. рисунок 5.1).

Тестування модуля розрахунку антропометричних індексів та пропорцій проводилося шляхом порівняння результатів автоматичних вимірювань із ручними вимірюваннями, виконаними кваліфікованим антропологом. Було проведено серію вимірювань за стандартними антропометричними методиками, такими як визначення лицевого індексу, носового індексу, орбітального індексу та інших ключових метрик [22].

Результати показали, що різниця між автоматичними та ручними вимірюваннями не перевищує 2%, що відповідає рівню точності, необхідному для більшості антропологічних досліджень.

При тестуванні модуля візуалізації результатів було оцінено коректність

відображення вимірювальних точок, ліній та значень на зображенні обличчя, а також зручність інтерпретації графічного представлення даних.

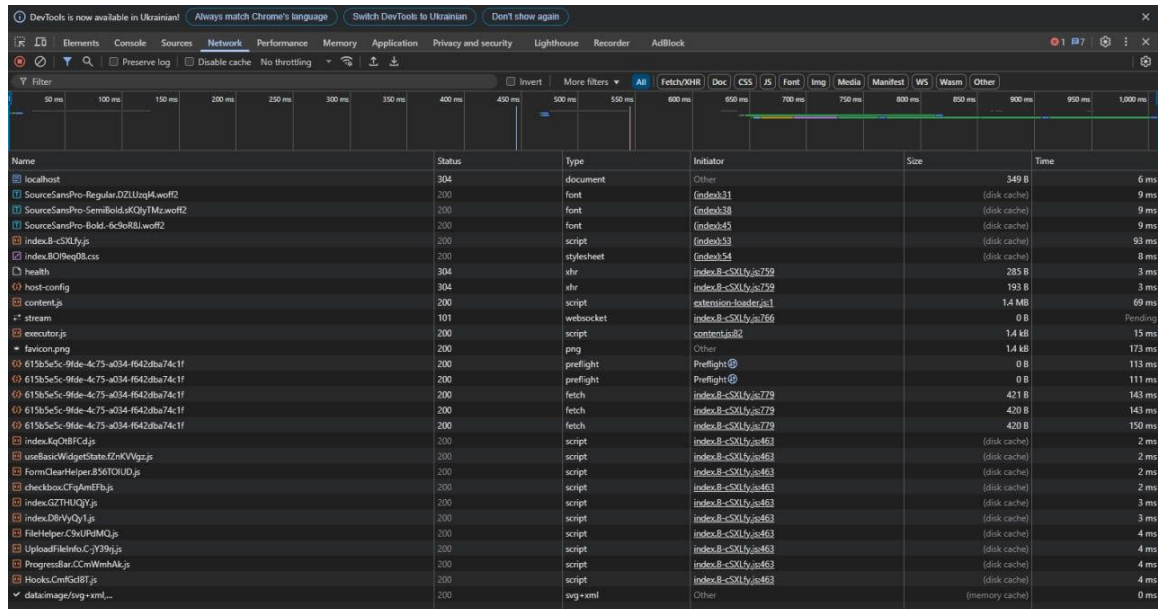


Рисунок 5.1 – Результати модульного тестування

Додатково було перевірено можливість генерації різних типів діаграм та графіків для порівняльного аналізу результатів вимірювань (див. рисунок 5.2).

Модуль збереження та експорту даних було протестовано на предмет коректності форматування вихідних файлів, повноти збереження метаданих та сумісності з популярними програмами для статистичного аналізу. Експорт даних у формати CSV та JSON працював коректно, зберігаючи всі необхідні параметри та дозволяючи в подальшому легко імпортувати дані в аналітичні системи, такі як R, SPSS або Excel.

Важливим аспектом тестування було інтеграційне тестування, яке перевіряло взаємодію між різними модулями системи. Особлива увага приділялася передачі даних між модулем визначення антропометричних точок та модулем розрахунку індексів, а також між модулем розрахунку та модулем візуалізації. Тестування показало, що інтеграція модулів працює без затримок та втрати даних, забезпечуючи плавний робочий процес.

Тестування проводиться з використанням операційної системи Windows 11. Спочатку при запуску програмного застосунку нас зустрічає головний екран додатку що зображено на рисунку 5.2.

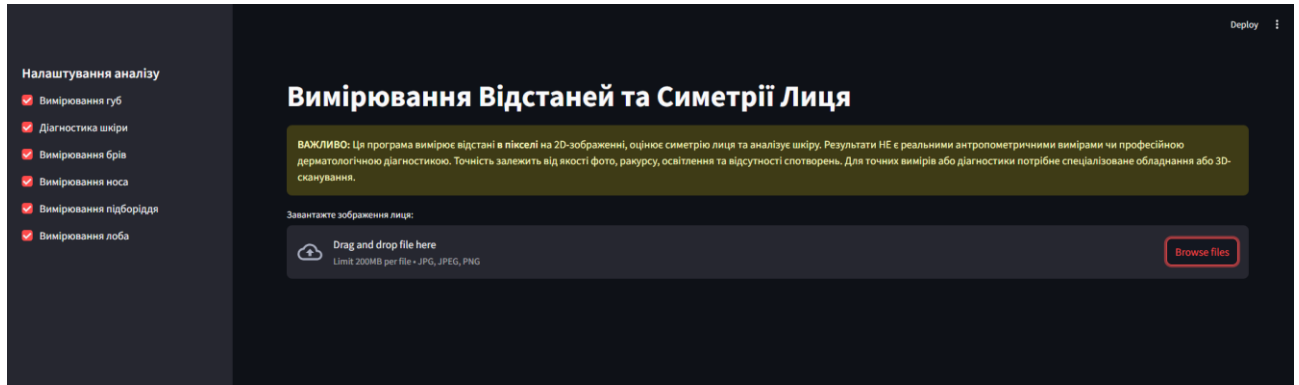


Рисунок 5.2 – Головний екран програмного застосунку

Було проведено функціональне тестування створеного програмного забезпечення, основною метою якого є перевірка правильності виконання ключових функцій застосунку, тому для початку потрібно обрати файл для початку антропологічних вимірювань голови людини що зображено на рисунку 5.3.

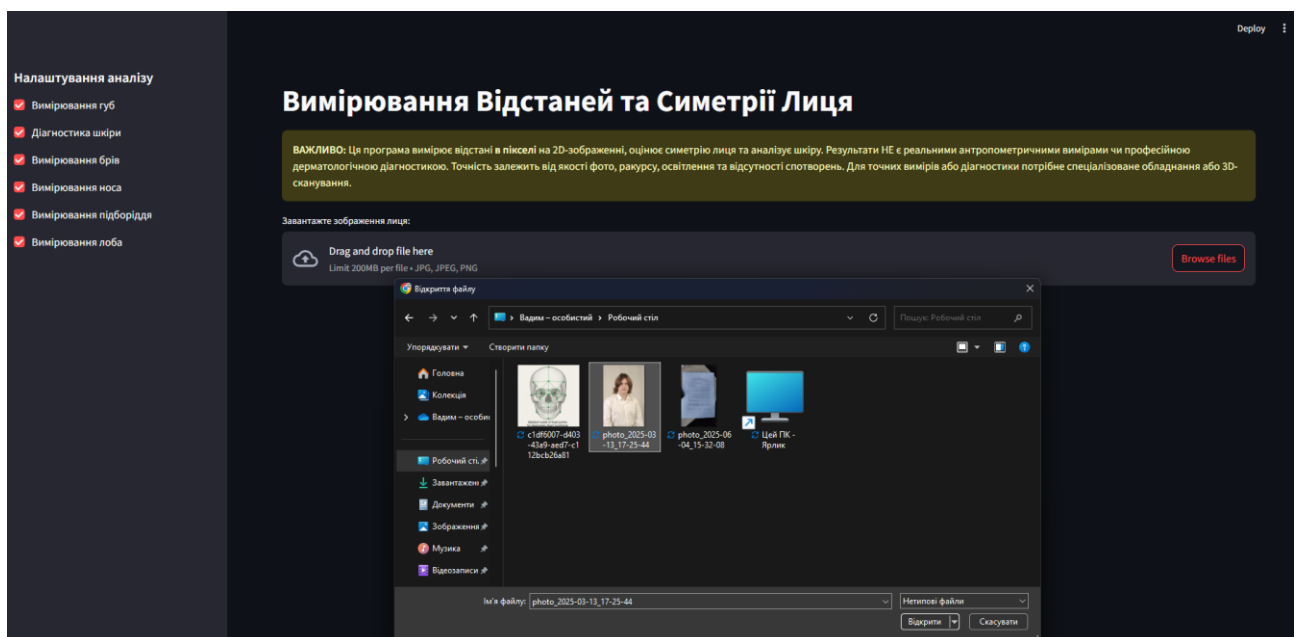


Рисунок 5.3 – Завантаження файлу для антропологічних вимірювань

Програмний застосунок дозволяє обрати всі файли усіх форматів щоб розпочати виміннювання. Після обрання файлу через кнопку «Browse files», вся детальна інформація про даний файл для дослідження буде відображатись в програмному застосунку що зображено на рисунку 5.4.

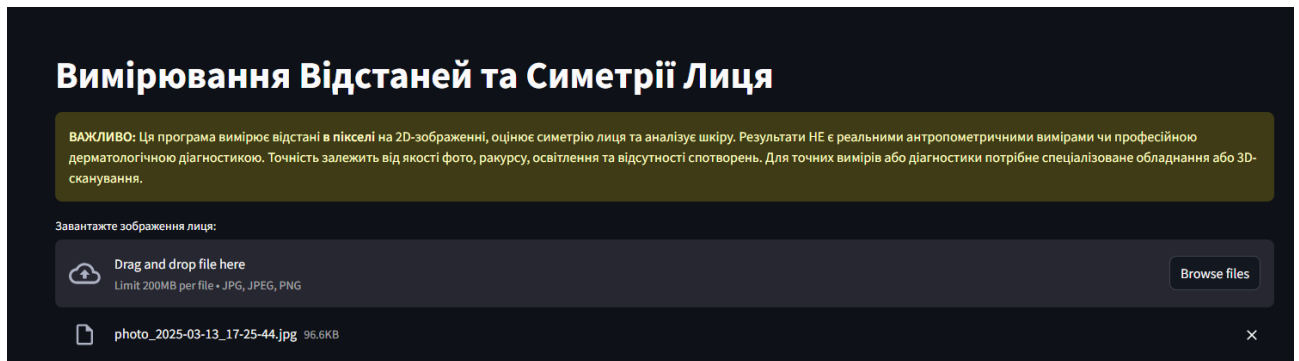


Рисунок 5.4 – Детальна інформація про обраний файл для дослідження

Після обрання файлу, обраний файл відкривається автоматично та завантажується (за замовчуванням) з додатковими та початковими параметрами в панелі налаштування, що зображено на рисунку 5.5.

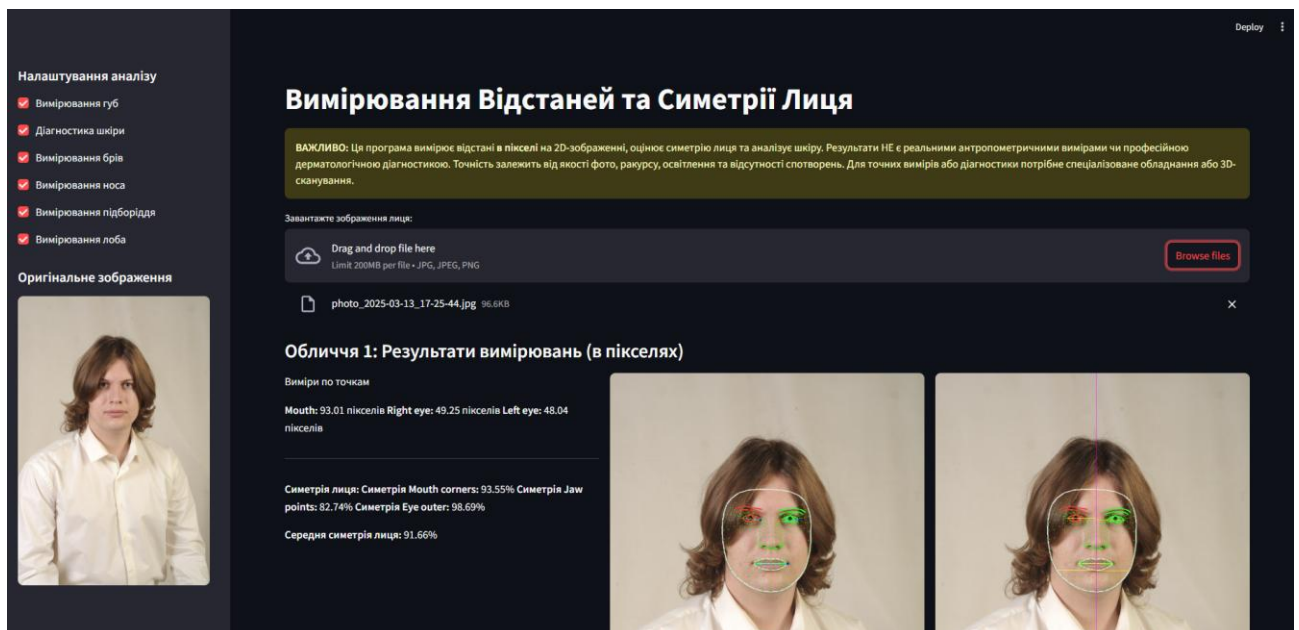


Рисунок 5.5 – Результат тестування завантаження файлу для вимірювання

Після успішного відкриття файлу для вимірювання антропологічних даних голови людини, є пряма можливість для аналізу додаткових параметрів поетапно та вибірково що робить кожний метод індивідуальним та надає більш широкий спектр для дослідження, що продемонстровано на рисунках 5.6 та 5.7.

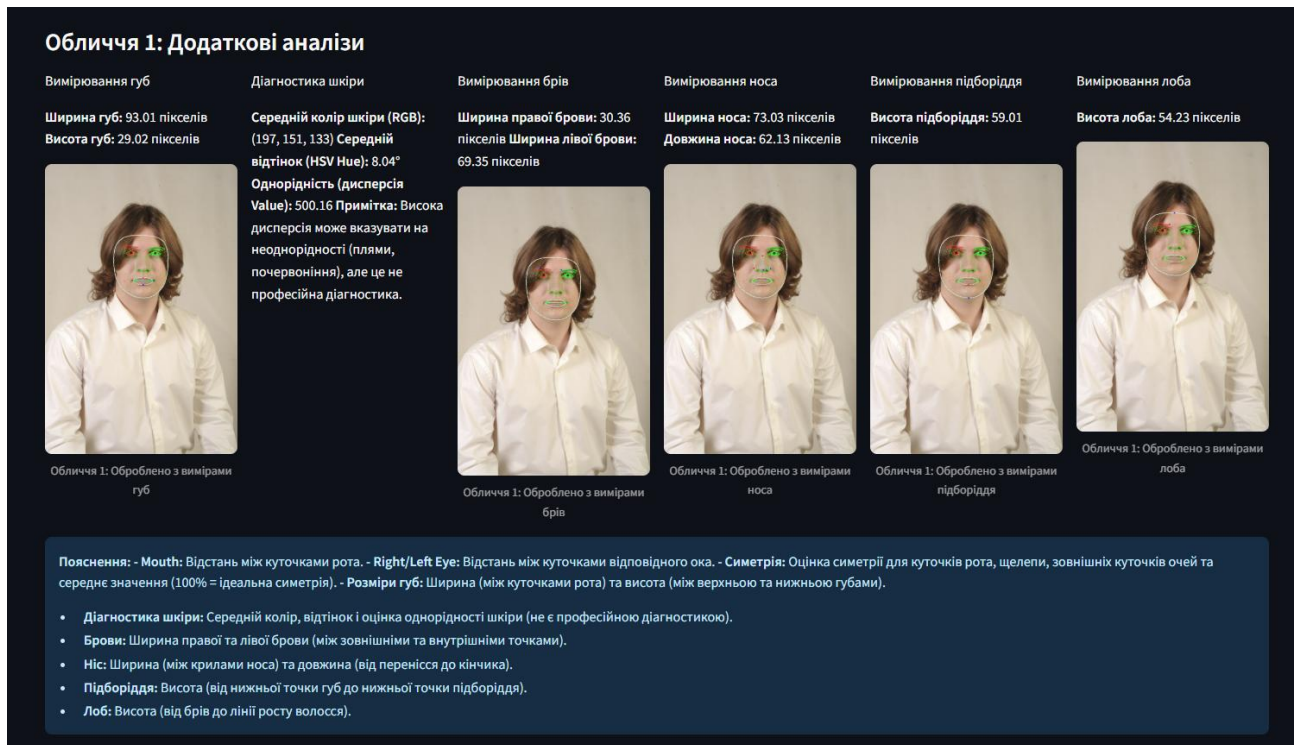


Рисунок 5.6 – Додаткові параметри антропологічних вимірювань голови

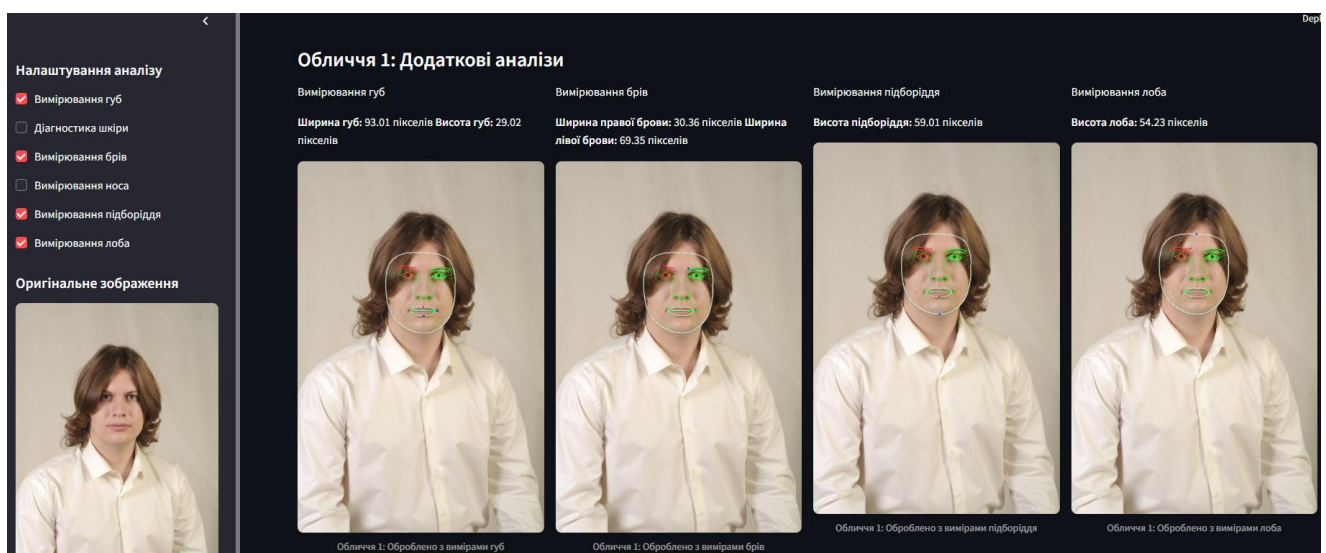


Рисунок 5.7 – Додаткові вибіркові параметри антропологічних вимірювань

При розробці програмного забезпечення також обов'язковою складовою є створення повноцінного готового зображення голови для подальших аналізів, що можливо завантажити після відкриття файлу на повний екран за допомогою кнопки «Fullscreen», що зображено на рисунку 5.8.

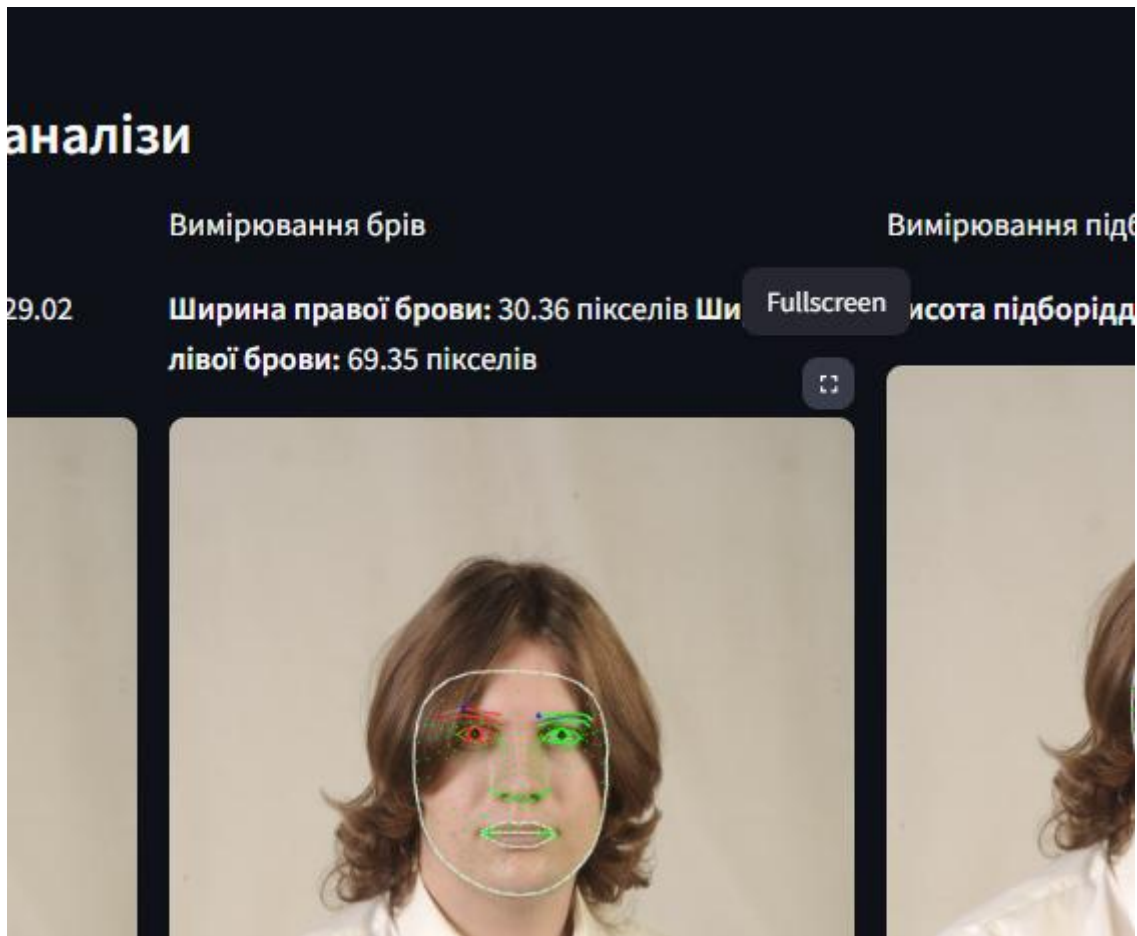


Рисунок 5.8 – Кнопка «Fullscreen» для завантаження виконаного аналізу

Крос-платформне тестування підтвердило працездатність системи на різних операційних системах (Windows, macOS, Linux) завдяки використанню Python та Streamlit. Невеликі відмінності у продуктивності між платформами не впливали на точність вимірювань та загальну функціональність системи [23].

Тестування продуктивності системи проводилося на комп'ютерах з різною конфігурацією, від базових ноутбуків до потужних робочих станцій.

Також під час тестування система вірно виконала розпізнавання не обраних додаткових параметрів, що відображається в системі «Не вибрано

жодної категорії для додаткового аналізу» що свідчить про успішне виконання системи роботи системи, та продемонстровано на рисунку 5.9.

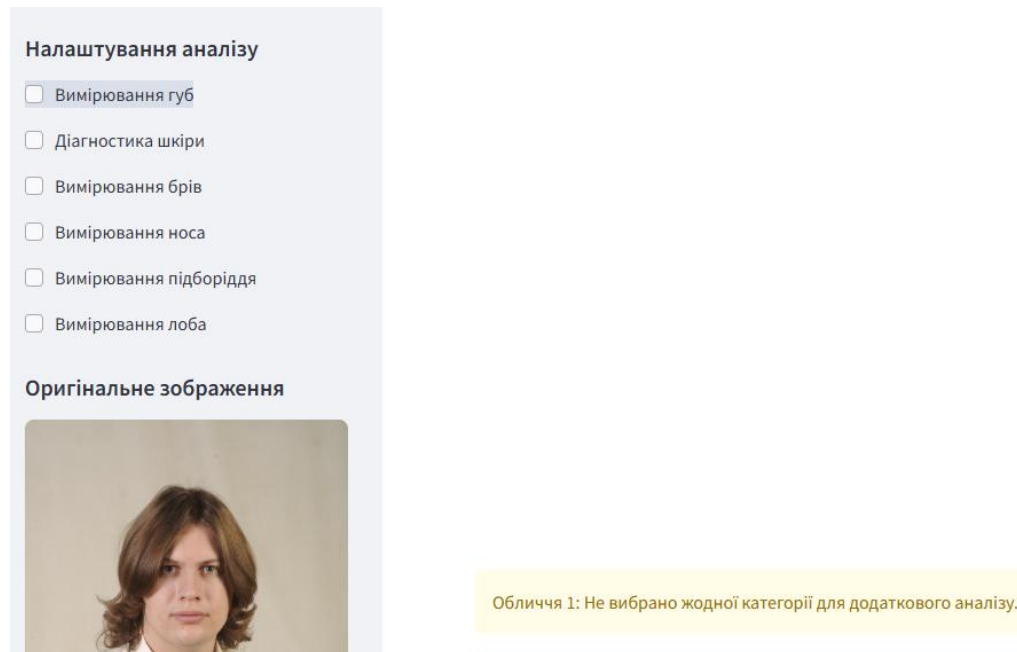


Рисунок 5.9 – Не вибрано жодної категорії для додаткового аналізу

Результати показали, що на сучасних комп'ютерах середньої потужності (процесор Intel Core i5/AMD Ryzen 5, 8 ГБ ОЗП) система здатна обробляти зображення та проводити вимірювання в режимі реального часу з частотою до 15 кадрів на секунду. На менш потужних конфігураціях рекомендується використовувати режим обробки статичних зображень.

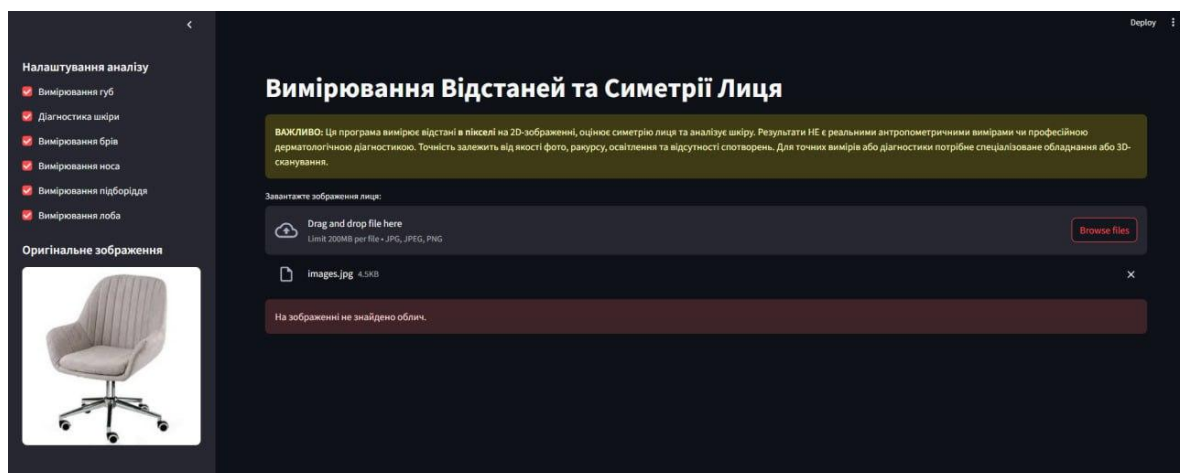


Рисунок 5.10 – Результат роботи програми якщо не знайдено обличчя

Тестування надійності системи включало перевірку стабільності роботи під навантаженням, коректності обробки нестандартних ситуацій та відновлення після збоїв. Система демонструвала стабільну роботу при обробці серій зображень, коректно обробляла винятки, пов'язані з неякісними вхідними даними, та зберігала проміжні результати для відновлення роботи у разі непередбачених завершень.

Тестування безпеки системи включало перевірку захисту персональних даних при збереженні результатів вимірювань, контроль доступу до функцій системи та захист від некоректних вхідних даних. Система забезпечує базову анонімізацію даних, підтримує розмежування доступу через стандартні механізми Streamlit та валідує вхідні дані для запобігання ін'єкціям коду.

За результатами комплексного тестування було виявлено та усунуто ряд незначних помилок, пов'язаних з обробкою крайових випадків при визначенні антропометричних точок та розрахунку індексів. Було оптимізовано алгоритми обробки зображень для покращення продуктивності на менш потужних комп'ютерах та додано додаткові підказки в інтерфейсі для покращення користувацького досвіду. Це, у свою чергу, зробило програмне забезпечення доступнішим для ширшого кола користувачів, зокрема в умовах сільських або віддалених медичних закладів.

Крім того, з метою покращення користувацького досвіду було оновлено інтерфейс програми: додано контекстні підказки, інтерактивні підсвітки ключових точок, спрощені повідомлення про помилки, а також візуальні індикатори правильності розміщення антропометричних маркерів. Завдяки цьому навіть недосвідчені користувачі можуть ефективно працювати з системою, мінімізуючи ймовірність помилок при введенні чи інтерпретації даних.

Загалом, проведене тестування підтвердило відповідність розробленої системи антропологічних вимірювань голови людини поставленим вимогам. Система забезпечує високу точність вимірювань, зручний інтерфейс користувача, надійне збереження даних та ефективну візуалізацію результатів.

5.2 Визначення технічних вимог

Для забезпечення коректної роботи системи антропологічних вимірювань голови людини необхідно враховувати технічні вимоги до апаратного та програмного забезпечення. Деталі щодо мінімальної та рекомендованої конфігурації розміщено в таблиці 5.1.

Таблиця 5.1 – Вимоги до апаратного та програмного забезпечення

Вимоги до конфігурування	Рекомендовано	Мінімально
Процесор	Intel Core i5 / AMD Ryzen 5 або аналогічний	Intel Core i3 / AMD Ryzen 3 або аналогічний
Оперативна пам'ять	8 ГБ RAM	4 ГБ RAM
Вільний простір на диску	10 ГБ	3 ГБ
Браузер	Chrome 90+, Firefox 88+, Edge 90+	Chrome 80+, Firefox 78+, Edge 80+
Операційна система	Windows 10/11, macOS 12+, Ubuntu 20.04+	Windows 7+, macOS 10.14+, Ubuntu 18.04+

Для запуску системи антропологічних вимірювань відкрийте веб-браузер та перейдіть за посиланням на Streamlit додаток, або запустіть локальну версію системи командою `streamlit run anthropometric_app.py` в консолі. Після завантаження ви побачите головний екран системи «AnthroHead».

У верхній частині екрану розташовано заголовок з назвою програми та бічну панель навігації, де можна перемикатися між різними модулями системи. Основний вміст сторінки містить заголовок «AnthroHead» та три основні секції вимірювання, результати та експорт даних.

У секції налаштування камери ви побачите вікно попереднього перегляду з можливістю завантажити зображення для аналізу. Процес калібрування є важливим для точних вимірювань і вимагає використання калібрувального об'єкта (спеціальної шахової дошки), який надається з системою або може бути роздрукований з PDF-файлу в розділі «Downloads».

Після визначення точок система автоматично обчислить основні антропометричні виміри, такі як ширина обличчя, висота обличчя, ширина носа, висота носа, відстань між очима, ширина рота та інші. Усі вимірювання відображаються в реальному часі в секції результатів. Для кожного виміру вказується його значення в міліметрах та відповідний індекс (де це доречно).

У секції результатів ви побачите таблицю з усіма обчисленими вимірами та візуалізацію у вигляді діаграми пропорцій обличчя. Таблиця містить назву виміру, його числове значення, одиниці вимірювання та нормативний діапазон (якщо доступний). Діаграма пропорцій візуально представляє співвідношення між різними частинами обличчя та дозволяє порівняти їх з середньостатистичними значеннями.

У вкладці «Proportions» відображаються основні пропорції обличчя у вигляді гістограми. У вкладці «Indices» представлені антропометричні індекси, такі як лицевий індекс, носовий індекс, орбітальний індекс та інші, у вигляді радіальної діаграми.

У секції експорту даних ви можете зберегти результати вимірювань для подальшого аналізу. Система підтримує експорт у кілька форматів: CSV, JSON та PDF-звіт. Для експорту даних натисніть кнопку «Export Data» та виберіть потрібний формат зі списку. При виборі CSV або JSON дані будуть збережені у відповідному форматі з усіма вимірюваннями та метаданими: дата і час вимірювання, інформація про камеру, параметри калібрування.

Для коректної роботи системи рекомендується дотримуватися таких умов під час вимірювань: особа повинна дивитися прямо в камеру, волосся не повинно закривати лоб або вуха, освітлення має бути рівномірним та достатнім, відстань до камери має становити 30-50 см. При недотриманні цих умов система відобразить попередження та рекомендації щодо корекції положення або освітлення.

5.3 Розробка інструкції для користувачів

Веб-застосунок «AnthroHead» був розроблений з метою забезпечення

зручного, ефективного та автоматизованого інструменту для первинної діагностики захворювань на основі антропологічних параметрів голови людини. Застосунок орієнтований на використання лікарями, медичними операторами, дослідниками та адміністраторами, кожен із яких має чітко визначену роль у системі.

Першим етапом взаємодії з системою є авторизація. Користувач вводить свої облікові дані (логін і пароль) для отримання доступу до функцій, передбачених його роллю. Для підвищення безпеки система може працювати в межах захищеної медичної мережі, обмежуючи доступ тільки для авторизованих працівників закладів охорони здоров'я.

Залежно від призначення, веб-застосунок AnthroHead підтримує кілька ролей користувачів:

- **Пацієнт / оператор:** має можливість ввести дані пацієнта та антропологічні вимірювання (довжина, ширина, обвід, висота голови, міжзінична відстань тощо), додати фото профілю пацієнта (за потреби), вказати скарги або супутні симптоми.

- **Адміністратор:** здійснює повне управління користувачами та налаштуванням системи. Може призначати ролі, переглядати статистику використання, контролювати цілісність і безпеку даних, здійснювати резервне копіювання та адміністрування обчислювального середовища.

Користувач створює новий запит на діагностику, вказуючи ключові параметри. Система дозволяє:

- Ввести зображення людини;
- ввести антропометричні вимірювання через автоматизовану систему (наприклад, з камери за допомогою MediaPipe);
- прикріпити зображення чи інші супровідні документи;

Після створення запиту:

- оператор або система ініціює процес автоматичної оцінки даних на основі вбудованого алгоритму;
- результати обробки передаються користувачу;

На основі аналізу вимірювань та виявлених аномалій система генерує структурований діагностичний звіт, що включає:

- основні антропологічні показники;
- список виявлених відхилень (якщо є);

Адміністратори можуть:

- контролювати статуси оброблених заявок;
- налаштовувати результати (наприклад, про нові заявки, критичні відхилення тощо);

Таким чином, веб-застосунок «AnthroHead» є багаторівневою системою, що забезпечує автоматизацію процесів первинної діагностики на основі морфологічних особливостей голови людини. Його гнучка рольова модель, адаптивний інтерфейс і високий ступінь автоматизації дозволяють зменшити час обробки запитів, покращити точність попередньої оцінки стану пацієнтів і підвищити загальну ефективність роботи медичних установ.

5.4 Висновок

У розділі 4 розглянуто процес розробки програмного застосунку для діагностування захворювань на основі антропологічних вимірювань голови.

Ключовими факторами стали точність вимірювань, швидкодія при обробці зображень у реальному часі та зручність користувацької взаємодії. Обрані технології — Python, OpenCV, MediaPipe, Streamlit — забезпечують високу якість розпізнавання, масштабованість і гнучкість інтерфейсу.

Також важливими були питання подальшого масштабування та інтеграції з медичними системами. Обраний технологічний стек дозволяє оновлювати та розширювати застосунок із мінімальними затратами.

Отже, технічне забезпечення є визначальним для ефективності, надійності й практичного використання розробленого програмного продукту в медичній сфері.

ВИСНОВКИ

У результаті виконання бакалаврської кваліфікаційної роботи розроблено теоретичний базис і програмний застосунок для діагностування захворювань на основі антропологічних вимірювань голови людини, що поєднує сучасні методи комп'ютерного зору, чисельного аналізу та геометричного моделювання.

Проаналізовано сучасний стан систем антропометричних вимірювань, зокрема методи визначення геометричних параметрів голови та обличчя, що використовуються у медичних, біометричних і дослідницьких цілях. Встановлено, що традиційні контактні методи антропометрії поступово замінюються безконтактними цифровими технологіями, заснованими на алгоритмах комп'ютерного зору. Визначено актуальність і перспективність використання відкритих бібліотек (MediaPipe, OpenCV, NumPy), що забезпечують високу точність і доступність технології.

Виконано порівняльний аналіз програмних аналогів, включаючи 3dMD Face System, Face Analyzer, MorphoJ та HeadScanner Pro. У результаті встановлено, що наявні рішення або є надто дорогими та складними у використанні, або мають обмежену функціональність і недостатню точність. Це обґрунтувало необхідність розробки нового рішення, яке поєднувало б точність, доступність і гнучкість.

Розроблено методи комп'ютерного аналізу морфологічних ознак обличчя, що дозволяють ідентифікувати характерні зміни, пов'язані з генетичними синдромами (Дауна, Аперта, Тернера тощо). Зокрема реалізовано скелетні полігональні моделі обличчя, алгоритми симетричного аналізу та кореляційні методи для визначення ключових параметрів (довжина, ширина, відстані між антропометричними точками).

Розроблено програмну архітектуру веб-застосунку, який реалізовано з використанням мови Python та фреймворку Streamlit. Застосунок має модульну структуру, до складу якої входять модулі визначення антропометричних точок (на базі MediaPipe), обчислення метрик, аналізу відхилень, візуалізації

результатів та генерації звітів. Забезпечено зручний інтерфейс користувача для введення, обробки й аналізу зображень.

Проведено тестування програмного забезпечення, яке включало модульні та навантажувальні тести. Результати засвідчили працездатність системи в умовах стандартної техніки та змін освітлення. Точність визначення ключових точок обличчя досягла рівня $\sim 0,5$ мм, що відповідає рівню базових професійних систем.

Визначено наукову новизну та практичну цінність розробки. Запропоновано нову геометричну модель взаємопов'язаних елементів обличчя для виявлення патологічних ознак. Метод, що поєднує полігональне моделювання, аналіз симетрії та кореляційний аналіз, є ефективним для виявлення морфологічних маркерів захворювань. Практична цінність полягає у створенні відкритої та адаптивної системи, яку можна використовувати в умовах телемедицини, попередньої діагностики, освітніх досліджень та біоінженерії.

Створений програмний застосунок є ефективним інструментом для автоматизованої діагностики, а також має потенціал для подальшого розвитку, зокрема — інтеграції з клінічними базами, розширення діагностичних моделей та використання методів глибокого навчання для підвищення точності прогнозів.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

[1] В. В. Мазур і О. Н. Романюк, «Застосування методів машинного навчання для автоматичного виявлення та класифікації антропометричних точок на зображеннях голови», *Матеріали IV Всеукраїнської науково-технічної конференції факультету інформаційних технологій та комп'ютерної інженерії*, Вінниця, Україна, 24–27 вересня 2025 р.

[2] В. В. Мазур і О. Н. Романюк, «Комп'ютерне діагностування захворювань за зображенням голови та обличчя», *XVIII Всеукраїнська науково-практична веб конференція аспірантів, студентів та молодих вчених*. Кривий Ріг, Україна, 25–27 березня 2025 р., Видавництво КНУ, 2025, с. 231–235.

[3] О. Н. Романюк, Н. В. Тітова, В. В. Мазур, С. О. Романюк і С. В. Котлик, «Аналіз систем контролю версій в програмуванні», *XXIV Всеукраїнська науково-технічна конференція молодих вчених, аспірантів та студентів*, Одеса, Україна, 18–19 квітня 2024 р., Видавництво ОНТУ, 2024, с. 261–263.

[4] В. В. Мазур, О. Н. Романюк і А. Р. Глоба, «Формування зображення рельєфної поверхні тривимірних об'єктів», *Матеріали VI Всеукраїнської науково-практичної інтернет-конференції студентів, аспірантів та молодих вчених*, Хмельницький, Україна, 30 листопада 2023 р., Видавництво ХНТУ, 2023, с. 29–30.

[5] О. Н. Романюк, Р. Ю. Чехмestрук і В. В. Мазур, «Вибір методу зафарбовування для задач рендерингу», *Матеріали VI Всеукраїнської науково-практичної інтернет-конференції студентів, аспірантів та молодих вчених «Сучасні інформаційні системи та технології»*, Херсон, Україна, Книжкове видавництво ФОП Вишемирський В. С., 2023, с. 51–52.

[6] А. Р. Глоба, В. Г. Дмитрієв, В. В. Мазур, О. Н. Романюк і Р. Ю. Чехмestрук, «Аналіз растрової графіки та засобів її формування та оброблення», *Міжнародна науково-практична Інтернет-конференція «Електронні інформаційні ресурси: створення, використання, доступ»*, Суми/Вінниця, Україна, 20–21 листопада 2023 р., с. 85–86.

[7] Р. Ю. Чехместрук, О. Н. Романюк, В. В. Мазур, А. Р. Глоба і Т. В. Тітова, «Метод скінченних елементів для симуляції тканин», *Міжнародна науково-практична Інтернет-конференція*, Суми–Вінниця, Україна, 20–21 листопада 2023 р., Видавництво НІКО/КЗВО, 2023, с. 309–310.

[8] О. Н. Романюк, В. В. Мазур і О. В. Романюк, «Використання індексів голови людини для комп'ютерного діагностування захворювань», *XXXIII Міжнародна науково-практична конференція MicroCAD-2025*, Харків, Україна, НТУ «ХП», 2025, с. 1574.

[9] О. Н. Романюк, В. П. Майданюк, В. В. Мазур, Н. В. Тітова і С. О. Романюк, «Особливості антропологічних вимірювань обличчя людини за тривимірною моделлю», *XXVII Міжнародна науково-технічна конференція «Технологія-2025»*, Київ, Україна, 2025.

[10] О. Н. Романюк, В. В. Мазур, Н. В. Тітова і С. О. Романюк, «Комп'ютерний аналіз симетрії обличчя», *Інформаційні технології – 2025: збірник тез XII Всеукраїнської науково-практичної конференції молодих учених*, Київ, Україна, 15 травня 2025 р., Київський столичний університет імені Бориса Грінченка, 2025, с. 244–247.

[11] Г. М. Зеленчук, Н. М. Козань і В. О. Чадюк, «Алгоритм комплексного антроскопічного, антропометричного та дерматогліфічного дослідження при судово-медичній ідентифікації невідомої особи», Івано-Франківський національний медичний університет, 2021.

[12] Дніпровський національний медичний університет, «Методичні рекомендації: антропометричні та фотометричні методи діагностики в ортодонтії», Дніпро, 2021.

[13] «Використання графічних 3D-зображень облич для експрес-діагностування та побудови біомедичних програмних засобів», *Матеріали міжнародної конференції*, Вінниця, Україна, 2024.

[14] J. C. Kolar і E. M. Salter, *Craniofacial Anthropometry: Practical Measurement of the Head and Face for Clinical, Surgical and Research Use*, Springfield, IL, США: Charles C Thomas, 2021.

- [15] C. Roberts, *A Manual of Anthropometry: A Guide to the Physical Examination and Measurement of the Human Body*, 2nd ed., 2021.
- [16] H. H. Wilder, *A Laboratory Manual of Anthropometry*, Philadelphia: P. Blakiston's Son & Co, 2023.
- [17] B. Mrugalska i W. Karwowski (ред.), *Anthropometry: Human Body Measurements and How to Use Them*, Boca Raton: CRC Press, 2023.
- [18] V. R. Preedy (ред.), *Handbook of Anthropometry: Physical Measures of Human Form in Health and Disease*, New York: Springer, 2021.
- [19] S. Pheasant, C. M. Haslegrave, *Bodyspace: Anthropometry, Ergonomics and the Design of Work*, 3rd ed., CRC Press, 2025.
- [20] Kathleen M. Robinette et al., *Civilian American and European Surface Anthropometry Resource (CAESAR)*, AFRL-HE-WP-TR-2002-0169, 2022.
- [21] M. A. Patterson, *Human Measurement: A Review of Anthropometry*, Taylor & Francis, 2021.
- [22] NIOSH, *Anthropometric Procedures for the Assessment of Worker Fit*, National Institute for Occupational Safety and Health, 2024.
- [23] R. Bridger, *Introduction to Human Factors and Ergonomics*, 4th ed., CRC Press, 2023.

ДОДАТКИ

Додаток А – Технічне завдання

Міністерство освіти та науки України
Вінницький національний університет
Факультет інформаційних технологій та комп'ютерної інженерії

ЗАТВЕРДЖУЮ

Завідувач кафедри ПЗ

Романюк О.Н.

«25» 03 2025 р.

Технічне завдання

на бакалаврську кваліфікаційну роботу «Розробка програмного застосунку
для діагностування захворювань на основі антропологічних вимірювань
голови людини»
за спеціальністю

121-Інженерія програмного забезпечення

Керівник бакалаврської кваліфікаційної роботи:

д.т.н., проф. Романюк О.Н.

«24» 03 2025 року.

Виконав:

студент гр. 5ПІ-216 Мазур В.В.

«24» 03 2025 року.

Вінниця-2025 року

1. Найменування та галузь застосування

Бакалаврська кваліфікаційна робота: «Розробка програмного застосунку для діагностування захворювань на основі антропологічних вимірювань голови людини».

Галузь застосування – вебтехнології.

2. Підстава для розробки.

Підставою для виконання бакалаврської кваліфікаційної роботи (БКР) є індивідуальне завдання на БКР та наказ №80 від «11» березня 2025 р. Ректора ВНТУ про закріплення тем БКР.

3. Мета та призначення розробки.

Метою роботи є підвищення продуктивності діагностики генетичних захворювань за рахунок розробки геометричних та морфологічних моделей комп'ютерного аналізу обличчя.

4. Вихідні дані для проведення НДР

Перелік основних літературних джерел, на основі яких буде виконуватись БКР:

1. Мазур В.В., Романюк О.Н. Застосування методів машинного навчання для автоматичного виявлення та класифікації антропометричних точок на зображеннях голови. Матеріали LIV Всеукраїнська науково-технічна конференція факультету інформаційних технологій та комп'ютерної інженерії (2025), Вінниця, 24-27 вересня 2025 р. – Вінниця.

2. Мазур В.В., Романюк О.Н. Комп'ютерне діагностування захворювань за зображенням голови та обличчя. XVIII Всеукраїнська науково-практична web конференція аспірантів, студентів та молодих вчених (2025), Кривий Ріг, 25-27 березня 2025 р. – Видавництво КНУ, 2025р. – С. 231-235

3. Мазур В. В., Романюк О. Н., Глоба А.Р. Формування зображення рельєфної поверхні тривимірних об'єктів. Матеріали VI Всеукраїнської науково-практичної інтернет-конференції студентів, аспірантів та молодих вчених (2023), Хмельницький, 30 листопада 2023 р. – Видавництво ХНТУ, 2023р. – С. 29-30

4. Романюк О.Н., Тітова Н. В., Мазур В.В, Романюк С.О., Котлик С.В.

Аналіз систем контролю версій в програмуванні. XXIV Всеукраїнська науково-технічна конференція молодих вчених, аспірантів та студентів (2024), Одеса, 18-19 квітня 2024 р. – Видавництво ОНТУ, 2024р. – С. 261-263

5. Чехместрук Р. Ю., Романюк О. Н., Мазур В. В., Глоба А.Р., Тітова Т.В. Метод скінченних елементів для симуляції тканин. Збірник матеріалів міжнародної науково-практичної Інтернет-конференції (2023), Суми-Вінниця, 20-21 листопада 2023 р. – Видавництво НІКО/КЗВО, 2023р. – С. 309-310

6. Романюк О. Н., Мазур В. В., Романюк О. В. Використання індексів голови людина для комп'ютерного діагностування захворювань. XXXIII Міжнародної науково-практичної конференції MicroCAD-2025, Харків, Видавництво НТУ «Харківський політехнічний інститут», 2025р. – С. 1574-1575

7. Романюк О.Н., Майданюк В.П., Мазур В.В., Тітова Н.В., Романюк С.О. Особливості антропологічних вимірювань обличчя людини за тривимірною моделлю. XXVII Міжнародної науково-технічної конференції технологія-2025, Київ, 2025

5. Технічні вимоги

Вхідні дані — зображення користувача.

Вихідні дані — метрики антропологічних вимірювань голови людини.

6. Конструктивні вимоги

Інтерфейс програми повинен відповідати естетичним та ергономічним вимогам, повинна бути зручною в обслуговуванні та керуванні.

Графічна та текстова документація повинна відповідати діючим стандартам України.

7. Перелік технічної документації, що пред'являється по закінченню робіт:

- пояснювальна записка до бакалаврської кваліфікаційної роботи;
- технічне завдання;
- лістинги програми.

8. Вимоги до рівня уніфікації та стандартизації

При розробці програмних засобів слід дотримуватися уніфікації і ДСТУ.

9. Стадії та етапи розробки:

№ з/п	Назва етапів бакалаврської кваліфікаційної роботи	Строк виконання етапів роботи
1	Аналіз стану розвитку систем діагностування захворювань на основі антропологічних вимірювань голови людини	25.03.25 – 31.03.25
2	Розробка методів та алгоритмів програмного застосунку	03.04.25 – 09.04.25
3	Варіантний аналіз та обґрунтування вибору засобів програмної реалізації	11.04.25 – 17.04.25
4	Розробка програмних модулів та методів системи діагностування захворювань на основі антропологічних вимірювань голови людини	19.04.25 – 03.05.25
5	Тестування програмного застосунку	05.05.25 – 18.05.25
6	Оформлення матеріалів до захисту БКР	20.05.25 – 30.05.25

10. Порядок контролю та прийняття

Виконання етапів бакалаврської кваліфікаційної роботи контролюється керівником згідно з графіком виконання роботи. Прийняття бакалаврської кваліфікаційної роботи здійснюється ДЕК, затвердженою зав. кафедрою згідно з графіком.

Додаток Б – Протокол перевірки на плагіат

ПРОТОКОЛ ПЕРЕВІРКИ КВАЛІФІКАЦІЙНОЇ РОБОТИ

Назва роботи: Програмна система діагностування захворювань на основі антропологічних вимірювань голови людини

Тип роботи: бакалаврська кваліфікаційна робота
(бакалаврська кваліфікаційна робота / магістерська кваліфікаційна робота)

Підрозділ кафедра програмного забезпечення, ФІТКІ, СП-21Б
(кафедра, факультет, навчальна група)

Коефіцієнт подібності текстових запозичень, виявлених у роботі системою StrikePlagiarism 5,4 %

Висновок щодо перевірки кваліфікаційної роботи (відмітити потрібне)

- ☒ Запозичення, виявлені у роботі, є законними і не містять ознак плагіату, фабрикації, фальсифікації. Роботу прийняти до захисту
- ☐ У роботі не виявлено ознак плагіату, фабрикації, фальсифікації, але надмірна кількість текстових запозичень та/або наявність типових розрахунків не дозволяють прийняти рішення про оригінальність та самостійність її виконання. Роботу направити на доопрацювання.
- ☐ У роботі виявлено ознаки плагіату та/або текстових маніпуляцій як спроб укриття плагіату, фабрикації, фальсифікації, що суперечить вимогам законодавства та нормам академічної доброчесності. Робота до захисту не приймається.

Експертна комісія:

_____	_____
(прізвище, ініціали, посада)	(підпис)
_____	_____
(прізвище, ініціали, посада)	(підпис)

Особа, відповідальна за перевірку _____

Черноволик Г. О.

З висновком експертної комісії ознайомлений(-на)

Керівник _____
(підпис)

д.т.н., проф. каф. ПЗ Романюк О.Н.
(прізвище, ініціали, посада)

Здобувач _____
(підпис)

Мазур В.В.
(прізвище, ініціали)

Додаток В – Лістинг програми

main.py

```
import streamlit as st
import cv2
import mediapipe as mp
import numpy as np
import math

# Ініціалізація MediaPipe Face Mesh
mp_face_mesh = mp.solutions.face_mesh
mp_drawing = mp.solutions.drawing_utils
mp_drawing_styles = mp.solutions.drawing_styles

# Словник для ключових точок лиця (обмежено до 3 вимірювань)
FACIAL_LANDMARKS_IDXS = {
    "mouth": (61, 291), # Лівий та правий куточки рота
    "right_eye": (33, 133), # Правий зовнішній та внутрішній куточки ока
    "left_eye": (362, 263), # Лівий внутрішній та зовнішній куточки ока
}

# Функція для розрахунку евклідової відстані
def calculate_distance(point1, point2):
    return math.sqrt((point1[0] - point2[0]) ** 2 + (point1[1] - point2[1]) ** 2)

# Функція для оцінки симетрії лиця
def calculate_face_symmetry(landmarks_pixels, image_shape,
annotated_image_symmetry):
    symmetry_results = {}
    height, width, _ = image_shape

    # Визначаємо центральну вісь через середину між внутрішніми куточками очей
    # (точки 133 і 362)
    if 133 in landmarks_pixels and 362 in landmarks_pixels:
```

```

right_eye_inner = landmarks_pixels[133]
left_eye_inner = landmarks_pixels[362]
center_x = (right_eye_inner[0] + left_eye_inner[0]) // 2
center_point = (center_x, (right_eye_inner[1] + left_eye_inner[1]) // 2)
else:
    return None, "Не вдалося визначити центральну вісь (очі не знайдено)."

# Аналіз симетрії для пар точок
symmetry_pairs = [
    ("mouth_corners", 61, 291), # Лівий і правий куточки рота
    ("jaw_points", 172, 397), # Точки на щелепі
    ("eye_outer", 33, 263) # Зовнішні куточки очей
]

symmetry_text = ""
total_symmetry_score = 0
valid_pairs = 0

for name, idx1, idx2 in symmetry_pairs:
    if idx1 in landmarks_pixels and idx2 in landmarks_pixels:
        point1 = landmarks_pixels[idx1] # Ліва сторона
        point2 = landmarks_pixels[idx2] # Права сторона

        # Відстань від кожної точки до центральної осі (по X)
        dist1 = abs(point1[0] - center_x)
        dist2 = abs(point2[0] - center_x)

        # Обчислюємо різницю у відстанях
        diff = abs(dist1 - dist2)
        avg_dist = (dist1 + dist2) / 2
        symmetry_score = 100 * (1 - diff / avg_dist) if avg_dist > 0 else 100

        symmetry_results[name] = symmetry_score
        symmetry_text += f"***Симетрія {name.replace('_', ' ').capitalize()}:**
{symmetry_score:.2f}%\n"

```

```

total_symmetry_score += symmetry_score
valid_pairs += 1

# Малюємо лінії до центральної осі на зображенні для симетрії
cv2.line(annotated_image_symmetry, point1, (center_x, point1[1]), (0, 255, 255), 1)
# Жовта лінія
cv2.line(annotated_image_symmetry, point2, (center_x, point2[1]), (0, 255, 255), 1)
# Жовта лінія
else:
    symmetry_text += f"**Симетрія {name.replace('_', ' ').capitalize()}:** Точки не
знайдено\n"

# Малюємо центральну вісь на зображенні для симетрії
cv2.line(annotated_image_symmetry, (center_x, 0), (center_x, height), (255, 0, 255), 1) #
Фіолетова лінія

# Обчислюємо середній показник симетрії
avg_symmetry = total_symmetry_score / valid_pairs if valid_pairs > 0 else 0
symmetry_text += f"\n**Середня симетрія лиця:** {avg_symmetry:.2f}%"

return symmetry_results, symmetry_text

# Функція для вимірювання розмірів губ
def measure_lips(landmarks_pixels, annotated_image_lips):
    lips_measurements = {}
    lips_text = ""

    # Ширина губ (між куточками рота, точки 61 і 291)
    if 61 in landmarks_pixels and 291 in landmarks_pixels:
        point1 = landmarks_pixels[61]
        point2 = landmarks_pixels[291]
        width = calculate_distance(point1, point2)
        lips_measurements["width"] = width
        lips_text += f"**Ширина губ:** {width:.2f} пікселів\n"

```

```

# Малюємо точки та лінію
cv2.circle(annotated_image_lips, point1, 3, (0, 0, 255), -1) # Червона точка
cv2.circle(annotated_image_lips, point2, 3, (255, 0, 0), -1) # Синя точка
cv2.line(annotated_image_lips, point1, point2, (255, 255, 0), 1) # Жовта лінія
else:
    lips_text += "***Ширина губ:** Точки не знайдено\n"

# Висота губ (між верхньою точкою верхньої губи (0) і нижньою точкою нижньої
губи (17))
if 0 in landmarks_pixels and 17 in landmarks_pixels:
    point1 = landmarks_pixels[0]
    point2 = landmarks_pixels[17]
    height = calculate_distance(point1, point2)
    lips_measurements["height"] = height
    lips_text += f"***Висота губ:** {height:.2f} пікселів\n"
    # Малюємо точки та лінію
    cv2.circle(annotated_image_lips, point1, 3, (0, 0, 255), -1) # Червона точка
    cv2.circle(annotated_image_lips, point2, 3, (255, 0, 0), -1) # Синя точка
    cv2.line(annotated_image_lips, point1, point2, (255, 255, 0), 1) # Жовта лінія
else:
    lips_text += "***Висота губ:** Точки не знайдено\n"

return lips_measurements, lips_text

# Функція для вимірювання брів
def measure_eyebrows(landmarks_pixels, annotated_image_eyebrows):
    eyebrows_measurements = {}
    eyebrows_text = ""

    # Брови: ширина правої (70, 105) та лівої (300, 336)
    if 70 in landmarks_pixels and 105 in landmarks_pixels:
        point1 = landmarks_pixels[70]
        point2 = landmarks_pixels[105]
        right_eyebrow_width = calculate_distance(point1, point2)

```

```

eyebrows_measurements["right_eyebrow_width"] = right_eyebrow_width
eyebrows_text += f"***Ширина правої брови:** {right_eyebrow_width:.2f}
пікселів\n"

cv2.circle(annotated_image_eyebrows, point1, 3, (0, 0, 255), -1)
cv2.circle(annotated_image_eyebrows, point2, 3, (255, 0, 0), -1)
cv2.line(annotated_image_eyebrows, point1, point2, (255, 255, 0), 1)
else:
    eyebrows_text += "***Ширина правої брови:** Точки не знайдено\n"

if 300 in landmarks_pixels and 336 in landmarks_pixels:
    point1 = landmarks_pixels[300]
    point2 = landmarks_pixels[336]
    left_eyebrow_width = calculate_distance(point1, point2)
    eyebrows_measurements["left_eyebrow_width"] = left_eyebrow_width
    eyebrows_text += f"***Ширина лівої брови:** {left_eyebrow_width:.2f} пікселів\n"
    cv2.circle(annotated_image_eyebrows, point1, 3, (0, 0, 255), -1)
    cv2.circle(annotated_image_eyebrows, point2, 3, (255, 0, 0), -1)
    cv2.line(annotated_image_eyebrows, point1, point2, (255, 255, 0), 1)
else:
    eyebrows_text += "***Ширина лівої брови:** Точки не знайдено\n"

return eyebrows_measurements, eyebrows_text

def measure_eyebrows(landmarks_pixels, annotated_image_eyebrows):
    eyebrows_measurements = {}
    eyebrows_text = ""

    # Права брова: ширина (70-105), висота (середина - до 159)
    if 70 in landmarks_pixels and 105 in landmarks_pixels:
        pt1 = landmarks_pixels[70]
        pt2 = landmarks_pixels[105]
        width = calculate_distance(pt1, pt2)
        eyebrows_measurements["right_eyebrow_width"] = width
        eyebrows_text += f"***Ширина правої брови:** {width:.2f} пікселів\n"
        cv2.line(annotated_image_eyebrows, pt1, pt2, (255, 255, 0), 1)
        cv2.putText(annotated_image_eyebrows, "Right Width", ((pt1[0]+pt2[0])/2,

```

```

(pt1[1]+pt2[1])/2 - 5),
    cv2.FONT_HERSHEY_SIMPLEX, 0.4, (255, 255, 0), 1)

# Висота правої брови
mid_x = (pt1[0] + pt2[0]) // 2
mid_y = (pt1[1] + pt2[1]) // 2
if 159 in landmarks_pixels:
    eye_point = landmarks_pixels[159]
    height = calculate_distance((mid_x, mid_y), eye_point)
    eyebrows_measurements["right_eyebrow_height"] = height
    eyebrows_text += f"***Висота правої брови:** {height:.2f} пікселів\n"
    cv2.line(annotated_image_eyebrows, (mid_x, mid_y), eye_point, (0, 255, 255), 1)
    cv2.putText(annotated_image_eyebrows, "Right Height", ((mid_x + eye_point[0]) //
2, (mid_y + eye_point[1]) // 2),
        cv2.FONT_HERSHEY_SIMPLEX, 0.4, (0, 255, 255), 1)
else:
    eyebrows_text += "***Права брова:** Недостатньо точок\n"

# Ліва брова: ширина (300-336), висота (до 386)
if 300 in landmarks_pixels and 336 in landmarks_pixels:
    pt1 = landmarks_pixels[300]
    pt2 = landmarks_pixels[336]
    width = calculate_distance(pt1, pt2)
    eyebrows_measurements["left_eyebrow_width"] = width
    eyebrows_text += f"***Ширина лівої брови:** {width:.2f} пікселів\n"
    cv2.line(annotated_image_eyebrows, pt1, pt2, (255, 255, 0), 1)
    cv2.putText(annotated_image_eyebrows, "Left Width", ((pt1[0]+pt2[0])/2,
(pt1[1]+pt2[1])/2 - 5),
        cv2.FONT_HERSHEY_SIMPLEX, 0.4, (255, 255, 0), 1)

# Висота лівої брови
mid_x = (pt1[0] + pt2[0]) // 2
mid_y = (pt1[1] + pt2[1]) // 2
if 386 in landmarks_pixels:
    eye_point = landmarks_pixels[386]

```

```

    height = calculate_distance((mid_x, mid_y), eye_point)
    eyebrows_measurements["left_eyebrow_height"] = height
    eyebrows_text += f"***Висота лівої брови:** {height:.2f} пікселів\n"
    cv2.line(annotated_image_eyebrows, (mid_x, mid_y), eye_point, (0, 255, 255), 1)
    cv2.putText(annotated_image_eyebrows, "Left Height", ((mid_x + eye_point[0]) //
2, (mid_y + eye_point[1]) // 2),
                cv2.FONT_HERSHEY_SIMPLEX, 0.4, (0, 255, 255), 1)
else:
    eyebrows_text += "***Ліва брова:** Недостатньо точок\n"

return eyebrows_measurements, eyebrows_text

# Функція для вимірювання носа
def measure_nose(landmarks_pixels, annotated_image_nose):
    nose_measurements = {}
    nose_text = ""

    # Ніс: ширина (129, 358) та довжина (6, 1)
    if 129 in landmarks_pixels and 358 in landmarks_pixels:
        point1 = landmarks_pixels[129]
        point2 = landmarks_pixels[358]
        nose_width = calculate_distance(point1, point2)
        nose_measurements["nose_width"] = nose_width
        nose_text += f"***Ширина носа:** {nose_width:.2f} пікселів\n"
        cv2.circle(annotated_image_nose, point1, 3, (0, 0, 255), -1)
        cv2.circle(annotated_image_nose, point2, 3, (255, 0, 0), -1)
        cv2.line(annotated_image_nose, point1, point2, (255, 255, 0), 1)
    else:
        nose_text += "***Ширина носа:** Точки не знайдено\n"

    if 6 in landmarks_pixels and 1 in landmarks_pixels:
        point1 = landmarks_pixels[6]
        point2 = landmarks_pixels[1]
        nose_length = calculate_distance(point1, point2)
        nose_measurements["nose_length"] = nose_length

```

```

nose_text += f"***Довжина носа:** {nose_length:.2f} пікселів\n"
cv2.circle(annotated_image_nose, point1, 3, (0, 0, 255), -1)
cv2.circle(annotated_image_nose, point2, 3, (255, 0, 0), -1)
cv2.line(annotated_image_nose, point1, point2, (255, 255, 0), 1)
else:
    nose_text += "***Довжина носа:** Точки не знайдено\n"

return nose_measurements, nose_text

```

Функція для вимірювання підборіддя

```

def measure_chin(landmarks_pixels, annotated_image_chin):
    chin_measurements = {}
    chin_text = ""

    # Підборіддя: висота (17, 152)
    if 17 in landmarks_pixels and 152 in landmarks_pixels:
        point1 = landmarks_pixels[17]
        point2 = landmarks_pixels[152]
        chin_height = calculate_distance(point1, point2)
        chin_measurements["chin_height"] = chin_height
        chin_text += f"***Висота підборіддя:** {chin_height:.2f} пікселів\n"
        cv2.circle(annotated_image_chin, point1, 3, (0, 0, 255), -1)
        cv2.circle(annotated_image_chin, point2, 3, (255, 0, 0), -1)
        cv2.line(annotated_image_chin, point1, point2, (255, 255, 0), 1)
    else:
        chin_text += "***Висота підборіддя:** Точки не знайдено\n"

    return chin_measurements, chin_text

```

Функція для вимірювання лоба

```

def measure_forehead(landmarks_pixels, annotated_image_forehead):
    forehead_measurements = {}
    forehead_text = ""

```

```

# Лоб: висота (середина між 105 і 336, до 10)
if 105 in landmarks_pixels and 336 in landmarks_pixels and 10 in landmarks_pixels:
    brow_mid_x = (landmarks_pixels[105][0] + landmarks_pixels[336][0]) // 2
    brow_mid_y = (landmarks_pixels[105][1] + landmarks_pixels[336][1]) // 2
    point1 = (brow_mid_x, brow_mid_y)
    point2 = landmarks_pixels[10]
    forehead_height = calculate_distance(point1, point2)
    forehead_measurements["forehead_height"] = forehead_height
    forehead_text += f"**Висота лоба:** {forehead_height:.2f} пікселів\n"
    cv2.circle(annotated_image_forehead, point1, 3, (0, 0, 255), -1)
    cv2.circle(annotated_image_forehead, point2, 3, (255, 0, 0), -1)
    cv2.line(annotated_image_forehead, point1, point2, (255, 255, 0), 1)
else:
    forehead_text += "**Висота лоба:** Точки не знайдено\n"

return forehead_measurements, forehead_text

# Функція для базової діагностики шкіри
def analyze_skin(image_bgr, landmarks_pixels):
    skin_text = ""

    # Визначаємо область шкіри (прямокутник між очима та ротом)
    if 133 in landmarks_pixels and 362 in landmarks_pixels and 61 in landmarks_pixels and
291 in landmarks_pixels:
        left_eye = landmarks_pixels[362]
        right_eye = landmarks_pixels[133]
        left_mouth = landmarks_pixels[61]
        right_mouth = landmarks_pixels[291]

        # Координати прямокутника (між очима та ротом)
        x_min = min(left_eye[0], right_eye[0], left_mouth[0], right_mouth[0])
        x_max = max(left_eye[0], right_eye[0], left_mouth[0], right_mouth[0])
        y_min = min(left_eye[1], right_eye[1])

```

```

y_max = max(left_mouth[1], right_mouth[1])

# Обрізаємо область шкіри
skin_region = image_bgr[y_min:y_max, x_min:x_max]
if skin_region.size == 0:
    return "Область шкіри не визначена."

# Функція для розширеної діагностики шкіри
def analyze_skin(image_bgr, landmarks_pixels):
    skin_text = ""

    # Визначаємо область шкіри (між очима та ротом)
    if 133 in landmarks_pixels and 362 in landmarks_pixels and 61 in landmarks_pixels and
291 in landmarks_pixels:
        left_eye = landmarks_pixels[362]
        right_eye = landmarks_pixels[133]
        left_mouth = landmarks_pixels[61]
        right_mouth = landmarks_pixels[291]

        # Координати прямокутника
        x_min = min(left_eye[0], right_eye[0], left_mouth[0], right_mouth[0])
        x_max = max(left_eye[0], right_eye[0], left_mouth[0], right_mouth[0])
        y_min = min(left_eye[1], right_eye[1])
        y_max = max(left_mouth[1], right_mouth[1])

        skin_region = image_bgr[y_min:y_max, x_min:x_max]

        if skin_region.size == 0:
            return "Область шкіри не визначена (порожній регіон)."
```

HSV та LAB колірні простори

```

skin_hsv = cv2.cvtColor(skin_region, cv2.COLOR_BGR2HSV)
skin_lab = cv2.cvtColor(skin_region, cv2.COLOR_BGR2Lab)
```

Середній колір у BGR → RGB

```

mean_bgr = np.mean(skin_region, axis=(0, 1))
mean_rgb = mean_bgr[::-1]
skin_text += f"Середній колір (RGB):** ({mean_rgb[0]:.0f}, {mean_rgb[1]:.0f},
{mean_rgb[2]:.0f})\n"

# HSV
hue = skin_hsv[:, :, 0]
saturation = skin_hsv[:, :, 1]
value = skin_hsv[:, :, 2]

mean_hue = np.mean(hue)
mean_saturation = np.mean(saturation)
mean_value = np.mean(value)
skin_text += f"Середній HSV:** Hue={mean_hue:.1f}°,
Saturation={mean_saturation:.1f}, Value={mean_value:.1f}\n"

var_value = np.var(value)
var_hue = np.var(hue)
skin_text += f"Дисперсія Value (однорідність освітлення):** {var_value:.2f}\n"
skin_text += f"Дисперсія Hue (однорідність відтінку):** {var_hue:.2f}\n"

# LAB
# Розрахунок середнього кольору для виводу як плашки
skin_bgr = cv2.cvtColor(skin_crop, cv2.COLOR_RGB2BGR)
mean_color_bgr = np.mean(skin_bgr, axis=(0, 1))
mean_color_rgb = mean_color_bgr[::-1].astype(np.uint8)

# Створення однотонного зображення (плашка кольору)
color_patch = np.zeros((100, 300, 3), dtype=np.uint8)
color_patch[:, :] = mean_color_rgb

st.image(color_patch, caption=f"Середній колір шкіри (RGB):
{tuple(mean_color_rgb)}")
l = skin_lab[:, :, 0]
a = skin_lab[:, :, 1]

```

```

b = skin_lab[:, :, 2]

mean_l = np.mean(l)
mean_a = np.mean(a)
mean_b = np.mean(b)
skin_text += f"Середній колір (LAB):** L={mean_l:.1f}, A={mean_a:.1f},
B={mean_b:.1f}\n"

# Орієнтовна оцінка тону шкіри
if mean_hue < 15 or mean_hue > 160:
    tone = "теплий (рожевий/червоний)"
elif 15 <= mean_hue <= 45:
    tone = "жовтуватий/золотистий"
elif 45 < mean_hue <= 90:
    tone = "оливковий"
else:
    tone = "нейтральний/холодний"

skin_text += f"Попередня оцінка тону шкіри:** {tone}\n"

# Додаткове зауваження
skin_text += "**Примітка:** Аналіз виконується лише за пікселями на
зображенні. Для точного визначення потрібні стандартизовані умови.\n"

else:
    skin_text = "**Аналіз шкіри:** Не вдалося знайти всі необхідні точки обличчя."

return skin_text

# Конвертуємо в HSV для аналізу відтінку
skin_hsv = cv2.cvtColor(skin_region, cv2.COLOR_BGR2HSV)

# Обчислюємо середній колір у RGB
mean_color_rgb = np.mean(skin_region, axis=(0, 1))
skin_text += f"Середній колір шкіри (RGB):** ({mean_color_rgb[2]:.0f},

```

```
{mean_color_rgb[1]:.0f}, {mean_color_rgb[0]:.0f})\n"
```

```
# Обчислюємо середній відтінок у HSV
```

```
mean_hue = np.mean(skin_hsv[:, :, 0])
```

```
skin_text += f"Середній відтінок (HSV Hue): {mean_hue:.2f} °\n"
```

```
# Оцінюємо однорідність (дисперсія кольору в каналі Value)
```

```
variance_value = np.var(skin_hsv[:, :, 2])
```

```
skin_text += f"Однорідність (дисперсія Value): {variance_value:.2f}\n"
```

```
skin_text += "Примітка: Висока дисперсія може вказувати на неоднорідності  
(плями, почервоніння), але це не професійна діагностика.\n"
```

```
else:
```

```
skin_text = "Не вдалося визначити область шкіри (точки не знайдено)."
```

```
return skin_text
```

```
# --- Streamlit інтерфейс ---
```

```
st.set_page_config(page_title="Антропометрія та Симетрія Лиця (пікселі),  
layout="wide")
```

```
st.title("Вимірювання Відстаней та Симетрії Лиця")
```

```
st.warning("""
```

```
Важливо: Ця програма вимірює відстані в пікселі на 2D-зображенні,  
оцінює симетрію лица та аналізує шкіру.
```

```
Результати НЕ є реальними антропометричними вимірами чи професійною  
дерматологічною діагностикою.
```

```
Точність залежить від якості фото, ракурсу, освітлення та відсутності спотворень.
```

```
Для точних вимірів або діагностики потрібне спеціалізоване обладнання або 3D-  
сканування.
```

```
""")
```

```
# Чеклісти для вибору категорій аналізу
```

```
st.sidebar.header("Налаштування аналізу")
```

```
show_Lips = st.sidebar.checkbox("Вимірювання губ", value=True)
```

```
show_Skin = st.sidebar.checkbox("Діагностика шкіри", value=True)
```

```
show_Eyebrows = st.sidebar.checkbox("Вимірювання брів", value=True)
```

```
show_Nose = st.sidebar.checkbox("Вимірювання носа", value=True)
```

```

show_Chin = st.sidebar.checkbox("Вимірювання підборіддя", value=True)
show_Forehead = st.sidebar.checkbox("Вимірювання лоба", value=True)

uploaded_file = st.file_uploader("Завантажте зображення лица:", type=["jpg", "jpeg",
"png"])

if uploaded_file is not None:
    # Читання зображення
    file_bytes = np.asarray(bytearray(uploaded_file.read()), dtype=np.uint8)
    image_bgr = cv2.imdecode(file_bytes, 1)
    if image_bgr is None:
        st.error("Не вдалося завантажити зображення. Перевірте формат або цілісність
файлу.")
        st.stop()
    image_rgb = cv2.cvtColor(image_bgr, cv2.COLOR_BGR2RGB)

    st.sidebar.header("Оригінальне зображення")
    st.sidebar.image(image_rgb, use_container_width=True)

    # Ініціалізація та обробка зображення
    with mp_face_mesh.FaceMesh(
        max_num_faces=5, # Підтримка кількох осіб
        refine_landmarks=True,
        min_detection_confidence=0.5,
        min_tracking_confidence=0.5) as face_mesh:

        results = face_mesh.process(image_rgb)
        height, width, _ = image_bgr.shape

        if results.multi_face_landmarks:
            for face_idx, face_landmarks in enumerate(results.multi_face_landmarks):
                # Створюємо окремі зображення для кожного обличчя
                annotated_image_measurements = image_bgr.copy() # Для вимірювань
                annotated_image_symmetry = image_bgr.copy() # Для симетрії

```

```

annotated_image_lips = image_bgr.copy() # Для вимірювань губ
annotated_image_eyebrows = image_bgr.copy() # Для брів
# Збір координат усіх точок у словник
for idx, landmark in enumerate(face_landmarks.landmark):
    x = int(landmark.x * width)
    y = int(landmark.y * height)
    landmarks_pixels[idx] = (x, y)

# --- Аналіз шкіри ---
if show_Skin:
    skin_analysis_text = analyze_skin(image_bgr, landmarks_pixels)
    st.subheader(f" 🧐 Аналіз шкіри (особа #{face_idx + 1})")
    st.markdown(skin_analysis_text)

# Відображення області шкіри
if 133 in landmarks_pixels and 362 in landmarks_pixels and 61 in
landmarks_pixels and 291 in landmarks_pixels:
    left_eye = landmarks_pixels[362]
    right_eye = landmarks_pixels[133]
    left_mouth = landmarks_pixels[61]
    right_mouth = landmarks_pixels[291]
    x_min = min(left_eye[0], right_eye[0], left_mouth[0], right_mouth[0])
    x_max = max(left_eye[0], right_eye[0], left_mouth[0], right_mouth[0])
    y_min = min(left_eye[1], right_eye[1])
    y_max = max(left_mouth[1], right_mouth[1])
    skin_crop = image_rgb[y_min:y_max, x_min:x_max]

    st.image(skin_crop, caption="Область шкіри для аналізу",
use_container_width=False)

# Побудова гістограм
import matplotlib.pyplot as plt

fig, ax = plt.subplots(1, 3, figsize=(15, 4))
colors = ['r', 'g', 'b']

```

```

titles = ['Red', 'Green', 'Blue']
for i in range(3):
    channel = skin_crop[:, :, i]
    ax[i].hist(channel.ravel(), bins=50, color=colors[i], alpha=0.7)
    ax[i].set_title(f"Гістограма {titles[i]}")
    ax[i].set_xlabel("Інтенсивність")
    ax[i].set_ylabel("Кількість пікселів")
st.pyplot(fig)

annotated_image_nose = image_bgr.copy() # Для носа
annotated_image_chin = image_bgr.copy() # Для підборіддя
annotated_image_forehead = image_bgr.copy() # Для лоба
landmarks_pixels = {}
measurements_pixels = {}

# Малюємо контури лиця, очей, губ на всіх зображеннях
for annotated_image in [annotated_image_measurements,
annotated_image_symmetry,
                        annotated_image_lips, annotated_image_eyebrows,
                        annotated_image_nose, annotated_image_chin,
                        annotated_image_forehead]:
    mp_drawing.draw_landmarks(
        image=annotated_image,
        landmark_list=face_landmarks,
        connections=mp_face_mesh.FACEMESH_CONTOURS,
        landmark_drawing_spec=None,

connection_drawing_spec=mp_drawing_styles.get_default_face_mesh_contours_style())

    mp_drawing.draw_landmarks(
        image=annotated_image,
        landmark_list=face_landmarks,
        connections=mp_face_mesh.FACEMESH_IRISES,
        landmark_drawing_spec=None,

```

```
connection_drawing_spec=mp_drawing_styles.get_default_face_mesh_iris_connections_style())
```

```
{tuple(mean_color_rgb)})

l = skin_lab[:, :, 0]
a = skin_lab[:, :, 1]
b = skin_lab[:, :, 2]

mean_l = np.mean(l)
mean_a = np.mean(a)
mean_b = np.mean(b)
skin_text += f"Середній колір (LAB):** L={mean_l:.1f}, A={mean_a:.1f},
B={mean_b:.1f}\n"

# Орієнтовна оцінка тону шкіри
if mean_hue < 15 or mean_hue > 160:
    tone = "теплий (рожевий/червоний)"
elif 15 <= mean_hue <= 45:
    tone = "жовтуватий/золотистий"
elif 45 < mean_hue <= 90:
    tone = "оливковий"
else:
    tone = "нейтральний/холодний"

skin_text += f"Попередня оцінка тону шкіри:** {tone}\n"

# Додаткове зауваження
skin_text += "**Примітка:** Аналіз виконується лише за пікселями на
зображенні. Для точного визначення потрібні стандартизовані умови.\n"

else:
    skin_text = "**Аналіз шкіри:** Не вдалося знайти всі необхідні точки обличчя."

return skin_text

# Конвертуємо в HSV для аналізу відтінку
```

```

skin_hsv = cv2.cvtColor(skin_region, cv2.COLOR_BGR2HSV)

# Збираємо координати ключових точок і малюємо всі точки
for landmark_id, point in enumerate(face_landmarks.landmark):

# --- Аналіз кольору шкіри ---
if skin_region is not None and skin_region.size > 0:
    # Конвертація зображення шкіри в колірний простір HSV
    skin_hsv = cv2.cvtColor(skin_region, cv2.COLOR_BGR2HSV)

    # Конвертація в LAB — для додаткового аналізу освітленості та тону
    skin_lab = cv2.cvtColor(skin_region, cv2.COLOR_BGR2Lab)

    # Обчислення середнього кольору у BGR та переведення в RGB
    mean_color_bgr = np.mean(skin_region, axis=(0, 1))
    mean_color_rgb = mean_color_bgr[::-1] # BGR → RGB
    skin_text += f"***Середній колір шкіри (RGB):** ({mean_color_rgb[0]:.0f},
{mean_color_rgb[1]:.0f}, {mean_color_rgb[2]:.0f})\n"

    # HSV компоненти: Hue (відтінок), Saturation (насиченість), Value (яскравість)
    mean_hue = np.mean(skin_hsv[:, :, 0])
    mean_saturation = np.mean(skin_hsv[:, :, 1])
    mean_value = np.mean(skin_hsv[:, :, 2])
    skin_text += f"***Середній HSV:** Hue={mean_hue:.2f}°,
Saturation={mean_saturation:.2f}, Value={mean_value:.2f}\n"

    # Дисперсія HSV Value — оцінка неоднорідності яскравості
    variance_value = np.var(skin_hsv[:, :, 2])
    skin_text += f"***Однорідність (дисперсія Value):** {variance_value:.2f}\n"

    # Дисперсія Hue — оцінка однорідності відтінку
    variance_hue = np.var(skin_hsv[:, :, 0])
    skin_text += f"***Відтінкова однорідність (дисперсія Hue):** {variance_hue:.2f}\n"

    # Аналіз LAB (освітленість і тон)
    mean_l = np.mean(skin_lab[:, :, 0])

```

```

mean_a = np.mean(skin_lab[:, :, 1])
mean_b = np.mean(skin_lab[:, :, 2])
skin_text += f"Середній колір у LAB:** L={mean_l:.1f}, A={mean_a:.1f},
B={mean_b:.1f}\n"

# Класифікація загального відтінку (приблизно)
if mean_hue < 15 or mean_hue > 160:
    tone_description = "теплий (рожевий або червонуватий)"
elif 15 <= mean_hue <= 45:
    tone_description = "золотисто-жовтий"
elif 45 < mean_hue <= 90:
    tone_description = "оливковий"
else:
    tone_description = "нейтральний/холодний"

skin_text += f"Попередня оцінка тону шкіри:** {tone_description}\n"

# Примітка щодо достовірності
skin_text += "**Примітка:** Аналіз базується на пікселях зображення і не замінює
професійне освітлення чи спектрофотометрію.\n"

else:
    skin_text = "Не вдалося визначити область шкіри (точки не знайдено або регіон
порожній)."

    x = int(point.x * width)
    y = int(point.y * height)
    landmarks_pixels[landmark_id] = (x, y)
    # Малюємо маленькі зелені точки для всіх landmarks
    for annotated_image in [annotated_image_measurements,
annotated_image_symmetry,
                                annotated_image_lips, annotated_image_eyebrows,
                                annotated_image_nose, annotated_image_chin,
                                annotated_image_forehead]:
        cv2.circle(annotated_image, (x, y), 1, (0, 255, 0), -1)

```

```

# --- Розрахунок та візуалізація основних вимірів ---
st.subheader(f"Обличчя {face_idx + 1}: Результати вимірювань (в пікселях)")
col1, col2, col3 = st.columns(3)
with col1:
    st.write("Виміри по точкам")
    results_text = ""
    # Вимірювання відстаней (обмежено до 3)
    for name, (idx1, idx2) in FACIAL_LANDMARKS_IDXS.items():
        if idx1 in landmarks_pixels and idx2 in landmarks_pixels:
            point1 = landmarks_pixels[idx1]
            point2 = landmarks_pixels[idx2]
            cv2.circle(annotated_image_measurements, point1, 3, (0, 0, 255), -1)
            cv2.circle(annotated_image_measurements, point2, 3, (255, 0, 0), -1)
            cv2.line(annotated_image_measurements, point1, point2, (255, 255, 0), 1)
            distance = calculate_distance(point1, point2)
            measurements_pixels[name] = distance
            results_text += f"***{name.replace('_', ' ').capitalize()}** {distance:.2f}
підселив\n"

        else:
            results_text += f"***{name.replace('_', ' ').capitalize()}** Точки не
знайдено\n"

    # Аналіз симетрії та додавання до результатів
    symmetry_results, symmetry_text =
calculate_face_symmetry(landmarks_pixels,
                        annotated_image_measurements.shape,
                        annotated_image_symmetry)

    if symmetry_results is not None:
        results_text += "\n---\n**Симетрія лиця:**\n" + symmetry_text
    else:
        results_text += "\n---\n**Симетрія лиця:** " + symmetry_text + "\n"

    st.markdown(results_text)

with col2:

```

```

        st.image(cv2.cvtColor(annotated_image_measurements,
cv2.COLOR_BGR2RGB),
                caption=f"Обличчя {face_idx + 1}: Оброблено з вимірами",
use_container_width=True)

        with col3:
            st.image(cv2.cvtColor(annotated_image_symmetry, cv2.COLOR_BGR2RGB),
                    caption=f"Обличчя {face_idx + 1}: Оброблено з віссю симетрії",
use_container_width=True)

            # --- Додаткові аналізи з чеклістом ---
            selected_categories = []
            if show_Lips:
                selected_categories.append(("Вимірювання губ", measure_lips,
annotated_image_lips,
                                        f"Обличчя {face_idx + 1}: Оброблено з вимірами губ"))
            if show_Skin:
                selected_categories.append(("Діагностика шкіри", analyze_skin, None, None))
            if show_Eyebrows:
                selected_categories.append(("Вимірювання брів", measure_eyebrows,
annotated_image_eyebrows,
                                        f"Обличчя {face_idx + 1}: Оброблено з вимірами брів"))
            if show_Nose:
                selected_categories.append(("Вимірювання носа", measure_nose,
annotated_image_nose,
                                        f"Обличчя {face_idx + 1}: Оброблено з вимірами носа"))
            if show_Chin:
                selected_categories.append(("Вимірювання підборіддя", measure_chin,
annotated_image_chin,
                                        f"Обличчя {face_idx + 1}: Оброблено з вимірами
підборіддя"))
            if show_Forehead:
                selected_categories.append(("Вимірювання лоба", measure_forehead,
annotated_image_forehead,
                                        f"Обличчя {face_idx + 1}: Оброблено з вимірами лоба"))

```

```

if selected_categories:
    st.subheader(f"Обличчя {face_idx + 1}: Додаткові аналізи")
    cols = st.columns(len(selected_categories))
    for col, (category_name, analysis_func, annotated_image, image_caption) in
zip(cols,
                                     selected_categories):
        with col:
            st.write(category_name)
            if category_name == "Діагностика шкіри":
                text = analysis_func(image_bgr, landmarks_pixels)
            else:
                measurements, text = analysis_func(landmarks_pixels,
annotated_image)

            st.markdown(text)
            if annotated_image is not None:
                st.image(cv2.cvtColor(annotated_image, cv2.COLOR_BGR2RGB),
                        caption=image_caption, use_container_width=True)
            else:
                st.warning(f"Обличчя {face_idx + 1}: Не вибрано жодної категорії для
додаткового аналізу.")

def measure_eyebrows(landmarks_pixels, annotated_image_eyebrows):
    eyebrows_measurements = {}
    eyebrows_text = ""
    # Права брова: ширина (70-105), висота (середина - до 159)
    if 70 in landmarks_pixels and 105 in landmarks_pixels:
        pt1 = landmarks_pixels[70]
        pt2 = landmarks_pixels[105]
        width = calculate_distance(pt1, pt2)
        eyebrows_measurements["right_eyebrow_width"] = width
        eyebrows_text += f"***Ширина правої брови:** {width:.2f} пікселів\n"
        cv2.line(annotated_image_eyebrows, pt1, pt2, (255, 255, 0), 1)
        cv2.putText(annotated_image_eyebrows, "Right Width", ((pt1[0]+pt2[0])/2,
(pt1[1]+pt2[1])/2 - 5),
                    cv2.FONT_HERSHEY_SIMPLEX, 0.4, (255, 255, 0), 1)

```

```

# Висота правої брови
mid_x = (pt1[0] + pt2[0]) // 2
mid_y = (pt1[1] + pt2[1]) // 2
if 159 in landmarks_pixels:
    eye_point = landmarks_pixels[159]
    height = calculate_distance((mid_x, mid_y), eye_point)
    eyebrows_measurements["right_eyebrow_height"] = height
    eyebrows_text += f"***Висота правої брови:** {height:.2f} пікселів\n"
    cv2.line(annotated_image_eyebrows, (mid_x, mid_y), eye_point, (0, 255, 255), 1)
    cv2.putText(annotated_image_eyebrows, "Right Height", ((mid_x + eye_point[0]) //
2, (mid_y + eye_point[1]) // 2),
                cv2.FONT_HERSHEY_SIMPLEX, 0.4, (0, 255, 255), 1)
else:
    eyebrows_text += "***Права брова:** Недостатньо точок\n"
# Ліва брова: ширина (300-336), висота (до 386)
if 300 in landmarks_pixels and 336 in landmarks_pixels:
    pt1 = landmarks_pixels[300]
    pt2 = landmarks_pixels[336]
    width = calculate_distance(pt1, pt2)
    eyebrows_measurements["left_eyebrow_width"] = width
    eyebrows_text += f"***Ширина лівої брови:** {width:.2f} пікселів\n"
    cv2.line(annotated_image_eyebrows, pt1, pt2, (255, 255, 0), 1)
    cv2.putText(annotated_image_eyebrows, "Left Width", ((pt1[0]+pt2[0])//2,
(pt1[1]+pt2[1])//2 - 5),
                cv2.FONT_HERSHEY_SIMPLEX, 0.4, (255, 255, 0), 1)
# Висота лівої брови
mid_x = (pt1[0] + pt2[0]) // 2
mid_y = (pt1[1] + pt2[1]) // 2
if 386 in landmarks_pixels:
    eye_point = landmarks_pixels[386]
    height = calculate_distance((mid_x, mid_y), eye_point)
    eyebrows_measurements["left_eyebrow_height"] = height
    eyebrows_text += f"***Висота лівої брови:** {height:.2f} пікселів\n"
    cv2.line(annotated_image_eyebrows, (mid_x, mid_y), eye_point, (0, 255, 255), 1)
    cv2.putText(annotated_image_eyebrows, "Left Height", ((mid_x + eye_point[0]) //

```

```

2, (mid_y + eye_point[1]) // 2),
        cv2.FONT_HERSHEY_SIMPLEX, 0.4, (0, 255, 255), 1)
else:
    eyebrows_text += "***Ліва брова:** Недостатньо точок\n"
return eyebrows_measurements, eyebrows_text

# Оновлення пояснень залежно від обраних категорій
explanations = ""
**Пояснення:**
- **Mouth:** Відстань між куточками рота.
- **Right/Left Eye:** Відстань між куточками відповідного ока.
- **Симетрія:** Оцінка симетрії для куточків рота, щелепи, зовнішніх куточків
очей та середнє значення (100% = ідеальна симетрія).
""

if show_Lips:
    explanations += "- **Розміри губ:** Ширина (між куточками рота) та висота
(між верхньою та нижньою губами).\n"
if show_Skin:
    explanations += "- **Діагностика шкіри:** Середній колір, відтінок і оцінка
однорідності шкіри (не є професійною діагностикою).\n"
if show_Eyebrows:
    explanations += "- **Брови:** Ширина правої та лівої брови (між зовнішніми
та внутрішніми точками).\n"
if show_Nose:
    explanations += "- **Ніс:** Ширина (між крилами носа) та довжина (від
перенісся до кінчика).\n"
if show_Chin:
    explanations += "- **Підборіддя:** Висота (від нижньої точки губ до нижньої
точки підборіддя).\n"
if show_Forehead:
    explanations += "- **Лоб:** Висота (від брів до лінії росту волосся).\n"
st.info(explanations)
else:
    st.error("На зображенні не знайдено обличч.")

```

Додаток Г – Ілюстративна частина

Розробка програмного застосунку для діагностування захворювань на основі антропологічних вимірювань голови людини

Виконав: студент 4 курсу

Групи 5ПІ-216

Спеціальності 121 – Інженерія програмного забезпечення

Мазур Вадим Вячеславович

Керівник: д.т.н., проф. каф. ПЗ Романюк О.Н.

Рисунок Г.1 – Титульний аркуш

Актуальність системи діагностування захворювань на основі антропологічних вимірювань голови людини

- У процесі діагностування захворювань на основі антропологічних вимірювань голови людини важливо враховувати потреби всіх категорій користувачів, включаючи нейровідмінних осіб.
- Більшість медичних застосунків створюються з орієнтацією на типову поведінку користувача та можуть не враховувати особливості сенсорного сприйняття або когнітивні труднощі. Це створює бар'єри для ефективного використання системи частиною пацієнтів або фахівців.
- Тому при проектуванні інтерфейсу важливо дотримуватись принципів інклюзивного дизайну: забезпечувати простоту взаємодії, знижене сенсорне навантаження, передбачувану навігацію та адаптивність до різних стилів сприйняття. Такий підхід сприятиме кращому доступу до технологій діагностики та підвищить загальну ефективність використання програмного продукту.



Рисунок Г.2 – Актуальність діагностування захворювань на основі
антропологічних вимірювань голови людини



Мета, предмет та об'єкт дослідження

Метою дослідження є підвищення оперативності діагностики генетичних захворювань за рахунок розробки геометричних та морфологічних моделей комп'ютерного аналізу обличчя.

Об'єкт дослідження є процес діагностування захворювань на основі антропологічних вимірювань.

Предмет дослідження є методи і засоби антропологічних вимірювань для задач діагностування.

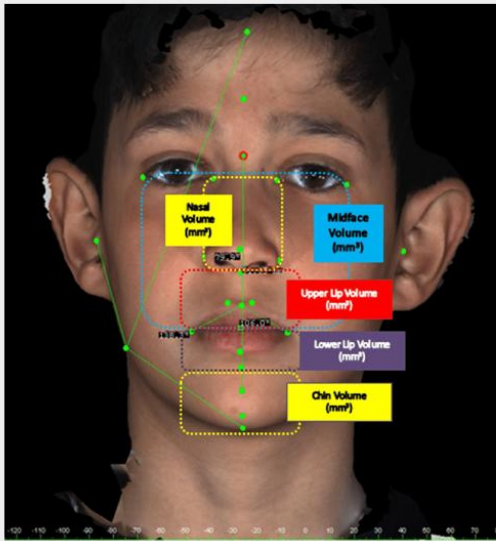
Рисунок Г.3 – Мета, предмет та об'єкт дослідження

Наукова новизна розробки

1. Вперше розроблено скелетну полігональну геометричну модель взаємопов'язаних елементів обличчя, особливість якої полягає у можливості комп'ютерного аналізу видозміни форми та стану лицьових параметрів, що дає можливість оперативного діагностування генетичних захворювань таких як: синдром Дауна, синдром Тернена, прогресивний параліч, нейрофіброматоз, синдром Вільямса, синдром Нунан, синдром Аперта та інші.
2. Розробка моделей аналізу зображень обличчя для виявлення ознак генетичних захворювань, що поєднує метрики з локалізованим морфологічним аналізом ключових зон голови, зокрема очей, лоба, носа, рота та підборіддя. Метод включає сегментацію обличчя з подальшим виявленням характерних морфологічних ознак, таких як асиметрія, диспропорція, або аномальні співвідношення між елементами обличчя.

Рисунок Г.4 – Наукова новизна розробки

1. 3dMD Face System



3dMDface System — це високоточна система тривимірного (3D) сканування обличчя, розроблена компанією 3dMD (США). Вона широко використовується в медицині, зокрема в щелепно-лицевій хірургії, ортодонції, пластичній хірургії та антропології, для отримання детальних цифрових моделей обличчя.

Основні характеристики 3dMDface System:

- Система використовує метод стереофотограмметрії, що забезпечує високу точність вимірювань без необхідності в опроміненні пацієнта.
- Геометрична точність системи становить до 0,2 мм, що дозволяє отримувати високоякісні 3D-моделі обличчя.
- Автоматична обробка: Програмне забезпечення автоматично створює безперервну 3D-поверхню з координатами x, y, z та накладає кольорову текстуру без необхідності ручного зшивання зображень.

Рисунок Г.5 – Аналог 3dMD Face System

2. Face Analyzer

Face Analyzer — це узагальнена назва для систем або програм, що здійснюють автоматичний аналіз обличчя на основі його зображення або відео. Залежно від розробника та призначення, такі системи можуть мати різну функціональність.

Основні характеристики Face Analyzer :

- Визначення форми обличчя, розміру та розташування очей, носа, губ, брів тощо.
- Системи можуть прогнозувати вік, стать та поточний емоційний стан особи (наприклад, щасливий, сумний, злий).
- Ідентифікація особи на основі біометричних характеристик.
- У медицині чи антропології такі системи можуть автоматично вимірювати відстані між ключовими точками обличчя (наприклад, ширина між зіницями, висота губ тощо).

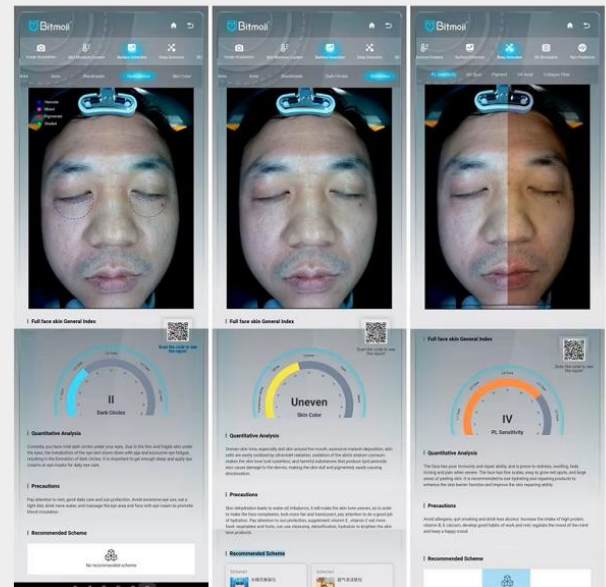


Рисунок Г.6 – Аналог Face Analyzer

3. Bellus3D Face Camera Pro



Bellus3D Face Camera Pro — це професійна камера для 3D-сканування обличчя, яка була розроблена компанією Bellus3D для точного створення високоякісних тривимірних моделей обличчя. Вона використовувалася у медицині, стоматології, косметології, розробці масок, AR/VR, біометрії та дослідженнях.

Основні характеристики 3dMDface System:

- Час сканування ~10 секунд (швидке захоплення + обробка в реальному часі)
- Система сканування інфрачервоне стереоскопічне сканування + RGB
- Кількість сенсорів: 2 ІЧ-сенсори (глибина) + 1 RGB-камера

Рисунок Г.7 – Аналог Bellus3D Face Camera Pro



Варіантний аналіз розробки

Система побудована на Python в середовищі PyCharm з використання MediaPipe, OpenCV, NumPy, Streamlit, Matplotlib. Усі модулі мають гнучку структуру, адаптовану під різні сценарії використання.

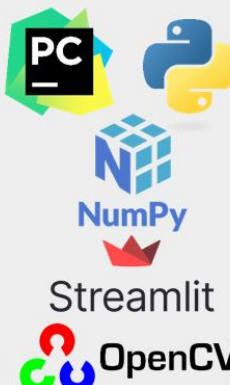
- MediaPipe — для детекції ключових точок;
- OpenCV — для роботи з зображенням;
- NumPy — для математичних обчислень;
- Streamlit — для створення веб-інтерфейсу;
- Matplotlib — для графічної візуалізації.



MediaPipe



matplotlib



OpenCV

Рисунок Г.8 – Варіантний аналіз розробки

Аналіз методу антропологічних вимірювань голови з використанням кореляційних зв'язків між різними морфометричними параметрами голови

Метод базується на встановленні статистичних зв'язків між морфометричними параметрами голови для створення прогнозних моделей і виявлення індивідуальних особливостей. Вимірюють основні показники: довжину, ширину та окружність голови, висоту і ширину обличчя, міжзіничну відстань тощо.

Дані збираються з репрезентативної вибірки, після чого обчислюються коефіцієнти кореляції Пірсона між параметрами. Сильні кореляції дозволяють будувати регресійні рівняння, наприклад, окружність голови + ширина. Такий підхід корисний, коли повний набір вимірювань недоступний — у судовій медицині, ортопедії, біометрії.

Метод дає змогу досліджувати відмінності за віком, статтю, етнічною належністю, а також використовує PCA для виділення найважливіших параметрів. Він широко застосовується у медичній практиці, ергономіці, судовій антропології та системах розпізнавання.

Рисунок Г.9 – Аналіз методу антропологічних вимірювань голови з використанням кореляційних зв'язків між різними морфометричними параметрами голови

Аналіз гібридного методу аналізу зображень обличчя для виявлення ознак генетичних захворювань

Гібридний метод аналізу зображень обличчя для виявлення ознак генетичних захворювань поєднує автоматизовані алгоритми комп'ютерного зору з експертними клінічними оцінками. Такий підхід дозволяє виявляти характерні риси, пов'язані з генетичними синдромами, з підвищеною точністю та надійністю.

Об'єктивна частина аналізу включає геометричні та текстурні метрики: відстані між ключовими точками обличчя, схожість текстур шкіри та рівень асиметрії. Ці показники обчислюються алгоритмами комп'ютерного зору та подаються у вигляді кількісних значень.

Суб'єктивна частина ґрунтується на оцінках медичних експертів, які аналізують риси обличчя за шкалами MOS та Лайкерта, а також узгодженість думок визначається коефіцієнтом Кендала. Це дозволяє включити клінічну інтуїцію у процес діагностики. Усі метрики нормалізуються та поєднуються в єдину гібридну оцінку, де вагові коефіцієнти підбираються з урахуванням їх впливу на точність діагнозу. Це забезпечує адаптивність методу під різні генетичні порушення.

Рисунок Г.10 – Аналіз гібридного методу аналізу зображень обличчя для виявлення ознак генетичних захворювань

Алгоритм роботи системи



Рисунок Г.11 – Алгоритм роботи системи

Діаграма алгоритму функції вимірювання розмірів губ

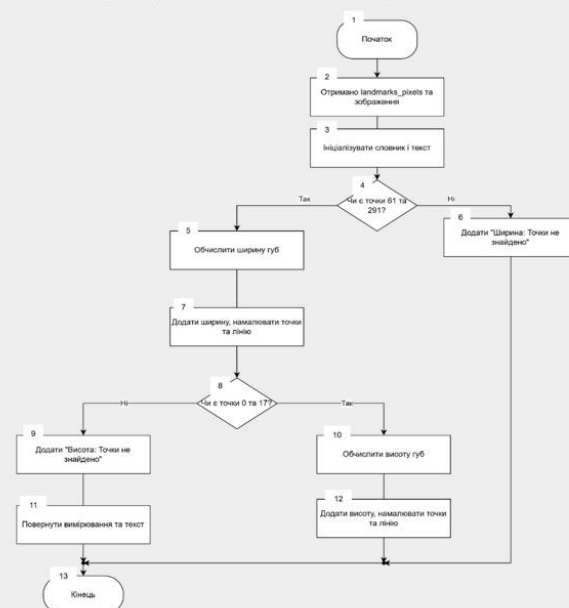


Рисунок Г.12 – Діаграма алгоритму функції вимірювання розмірів губ

Діаграма алгоритму функції для оцінки симетрії обличчя



Рисунок Г.13 – Діаграма алгоритму функції для оцінки симетрії обличчя

Діаграма алгоритму обчислення аналізу кольору шкіри

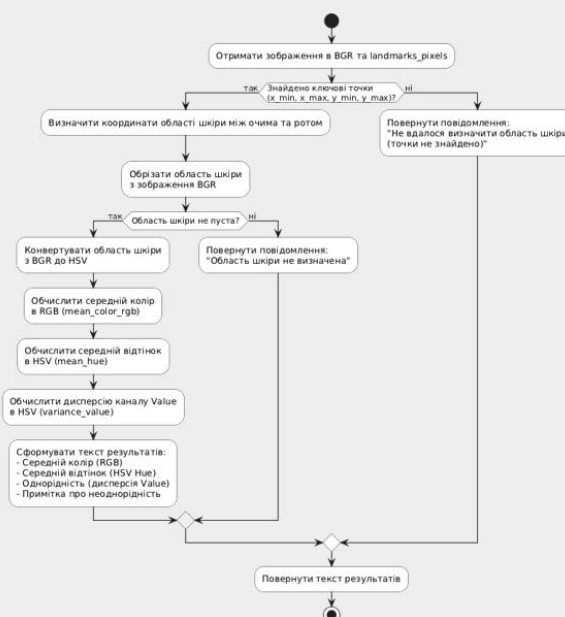


Рисунок Г.14 – Діаграма алгоритму обчислення аналізу кольору шкіри

Діаграма алгоритму обробки координат ключових точок



Рисунок Г.15 – Діаграма алгоритму обробки координат ключових точок

Система з завантаженням зображенням

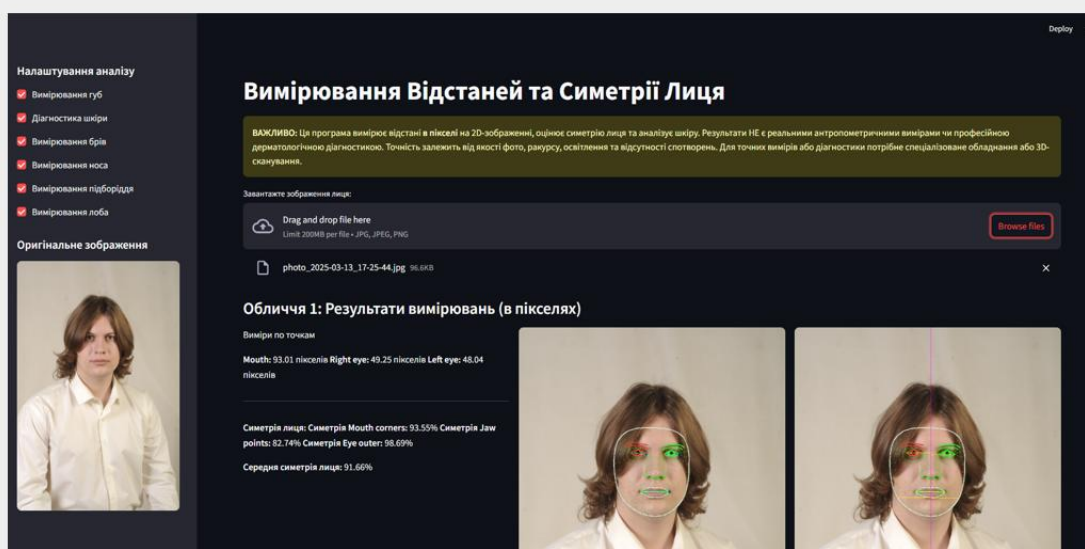


Рисунок Г.16 – Система з завантаженням зображенням

Відображення додаткових параметрів вимірювання

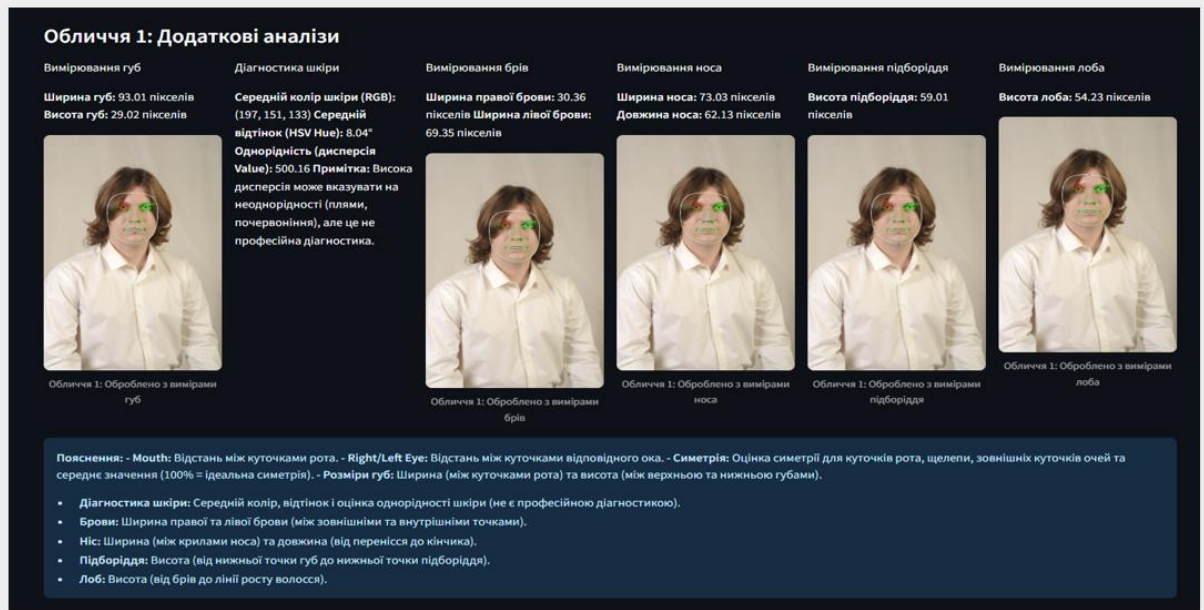


Рисунок Г.17 – Відображення додаткових параметрів вимірювання

Висновки

1. Проаналізовано сучасний стан систем антропометричних вимірювань, зокрема методи визначення геометричних параметрів голови та обличчя, що використовуються у медичних, біометричних і дослідницьких цілях;
2. Виконано порівняльний аналіз програмних аналогів та в результаті встановлено, що наявні рішення або є надто дорогими та складними у використанні, або мають обмежену функціональність і недостатню точність. Це обґрунтувало необхідність розробки нового рішення, яке поєднувало б точність, доступність і гнучкість;
3. Розроблено методи комп'ютерного аналізу морфологічних ознак обличчя, що дозволяють ідентифікувати характерні зміни, пов'язані з генетичними синдромами (Дауна, Аперта, Тернера тощо). Зокрема реалізовано скелетні полігональні моделі обличчя, алгоритми симетричного аналізу та кореляційні методи для визначення ключових параметрів (довжина, ширина, відстані між антропометричними точками);
4. Розроблено алгоритми та програмні модулі, що реалізують запропоновані методи діагностування захворювань на основі антропологічних вимірювань голови людини;
5. Проведено тестування розроблених програмних модулів на різних зображеннях та вимірюваннях голови людини, що підтвердило їхню ефективність та коректність роботи.

Рисунок Г.18 – Висновки

Апробація та публікація

- [1] В. В. Мазур і О. Н. Романюк, «Застосування методів машинного навчання для автоматичного виявлення та класифікації антропометричних точок на зображеннях голови», Матеріали IV Всеукраїнської науково-технічної конференції факультету інформаційних технологій та комп'ютерної інженерії, Вінниця, Україна, 24–27 вересня 2025 р.
- [2] В. В. Мазур і О. Н. Романюк, «Комп'ютерне діагностування захворювань за зображенням голови та обличчя», XVIII Всеукраїнська науково-практична веб конференція аспірантів, студентів та молодих вчених. Кривий Ріг, Україна, 25–27 березня 2025 р., Видавництво КНУ, 2025, с. 231–235.
- [3] О. Н. Романюк, Н. В. Тітова, В. В. Мазур, С. О. Романюк і С. В. Котлик, «Аналіз систем контролю версій в програмуванні», XXIV Всеукраїнська науково-технічна конференція молодих вчених, аспірантів та студентів, Одеса, Україна, 18–19 квітня 2024 р., Видавництво ОНТУ, 2024, с. 261–263.
- [4] В. В. Мазур, О. Н. Романюк і А. Р. Глоба, «Формування зображення рельєфної поверхні тривимірних об'єктів», Матеріали VI Всеукраїнської науково-практичної Інтернет-конференції студентів, аспірантів та молодих вчених, Хмельницький, Україна, 30 листопада 2023 р., Видавництво ХНТУ, 2023, с. 29–30.
- [5] О. Н. Романюк, Р. Ю. Чехмиструк і В. В. Мазур, «Вибір методу зафарбовування для задач рендерингу», Матеріали VI Всеукраїнської науково-практичної Інтернет-конференції студентів, аспірантів та молодих вчених «Сучасні інформаційні системи та технології», Херсон, Україна, Книжкове видавництво ФОП Вишемирський В. С., 2023, с. 51–52.
- [6] А. Р. Глоба, В. Г. Дмитрієв, В. В. Мазур, О. Н. Романюк і Р. Ю. Чехмиструк, «Аналіз растрової графіки та засобів її формування та оброблення», Міжнародна науково-практична Інтернет-конференція «Електронні інформаційні ресурси: створення, використання, доступ», Суми/Вінниця, Україна, 20–21 листопада 2023 р., с. 85–86.
- [7] Р. Ю. Чехмиструк, О. Н. Романюк, В. В. Мазур, А. Р. Глоба і Т. В. Тітова, «Метод скінченних елементів для симуляції тканин», Міжнародна науково-практична Інтернет-конференція, Суми–Вінниця, Україна, 20–21 листопада 2023 р., Видавництво НІКО/КЗВО, 2023, с. 309–310.
- [8] О. Н. Романюк, В. В. Мазур і О. В. Романюк, «Використання індексів голови людини для комп'ютерного діагностування захворювань», XXXIII Міжнародна науково-практична конференція MicroCAD-2025, Харків, Україна, НТУ «ХПІ», 2025, с. 1574.
- [9] О. Н. Романюк, В. П. Майданюк, В. В. Мазур, Н. В. Тітова і С. О. Романюк, «Особливості антропологічних вимірювань обличчя людини за тривимірною моделлю», XXVII Міжнародна науково-технічна конференція «Технологія-2025», Київ, Україна, 2025.
- [10] О. Н. Романюк, В. В. Мазур, Н. В. Тітова і С. О. Романюк, «Комп'ютерний аналіз симетрії обличчя», Інформаційні технології – 2025: збірник тез XII Всеукраїнської науково-практичної конференції молодих учених, Київ, Україна, 15 травня 2025 р., Київський столичний університет імені Бориса Грінченка, 2025, с. 244–247.

Рисунок Г.19 –Апробація та публікація

Дякую за увагу

Рисунок Г.20 – Закінчення презентації

Додаток Д – Акт впровадження

Директор ТОВ «ДРЕССЛАБ ЮЕЙ»



Чехместрук Р.Ю.

«10» березня 2025 р.

АКТ

впровадження програмних засобів для тестування якості формування графічних зображень автора Мазура Вадима Вячеславовича

Комісія, у складі директора Чехместрука Р.Ю., референта Чехместрук Л.Т. склали цей акт про те, що у ТОВ «ДРЕССЛАБ ЮЕЙ» впроваджено програмні засоби для тестування якості формування цифрових графічних зображень автора Мазура В.В.

Використання результатів роботи Мазура В.В. дозволило отримати метричні оцінки PSNR, MSE, SSIM, VIF, FSIM, NIQE, BRISQUE, PIQE для оцінювання якості формування графічних зображень.

Мазуром В.В. передано в ТОВ «ДРЕССЛАБ ЮЕЙ» програмне забезпечення для оцінки якості формування графічних зображень.

Члени комісії: Чехместрук Р.Ю.

Чехместрук Л.Т.