

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»**

Факультет прикладної математики

Кафедра прикладної математики

«До захисту допущено»

Завідувач кафедри

\_\_\_\_\_ О.Р. Чертов

“ \_\_\_ ” \_\_\_\_\_ 2015 р.

**Дипломна робота**  
**на здобуття ступеня бакалавра**

зі спеціальності 6.040301 «Прикладна математика»

на тему: Математичне та програмне забезпечення автоматичної системи з підбору розмірів одягу за фотографією

Виконав студент 4 курсу, групи КМ-11

Барбинягра Артем Михайлович \_\_\_\_\_

Керівник к.т.н. доцент Сирота С.В. \_\_\_\_\_

Консультант з нормоконтролю ст. викладач Мальчиков В.В. \_\_\_\_\_

Рецензент к.т.н. доцент Замятін Д.С. \_\_\_\_\_

Засвідчую, що у цій дипломній роботі  
немає запозичень з праць інших  
авторів без відповідних посилань.  
Студент \_\_\_\_\_

Київ – 2015

**Національний технічний університет України  
«Київський політехнічний інститут»**

Факультет прикладної математики

Кафедра прикладної математики

Рівень вищої освіти – перший (бакалаврський)

Спеціальність 6.040301 «Прикладна математика»

ЗАТВЕРДЖУЮ

Завідувач кафедри

\_\_\_\_\_ О. Р. Чертов

« \_\_\_\_ » \_\_\_\_\_ 2015р.



**на дипломну роботу студенту  
Барбинягрі Артему Михайловичу**

1. Тема роботи «Математичне та програмне забезпечення автоматичної системи підбору розмірів одягу за фотографією»

керівник роботи – доцент Сирота С.В.

затверджені наказом по університету від "19" травня 2015 р. № 1039-С.

2. Строк подання студентом роботи "12" червня 2015р.

3. Вихідні дані до роботи:

– Фотографія користувача, що стоїть в заданій позі на зеленому фоні

4. Зміст розрахунково-пояснювальної записки (перелік завдань, які потрібно розробити):

- вивчити літературні джерела за тематикою дослідження;
- провести порівняльний аналіз математичних методів для пошуку контуру людини на фотографії
- провести порівняльний аналіз математичних методів для визначення антропометричних розмірів за існуючим контуром
- обґрунтувати вибір обраних методів
- реалізувати математичне програмне забезпечення на основі клієнт-серверної архітектури;
- Оформити документацію до дипломної роботи;

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

- ілюстрації вхідних фотографій
- ілюстрації результатів роботи програми

6. Консультанти розділів проекту (роботи):

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Нормоконтроль	ст.викладач Мальчиков В.В.		

7. Дата видачі завдання «28» жовтня 2014

### Календарний план

№ з/п	Назва етапів роботи та питань, які мають бути розроблені відповідно до завдання	Термін виконання	Позначки керівника про виконання завдань
1.	Вивчення літератури за тематикою роботи	20.11.2014	
2.	Порівняльний аналіз методів виділення контуру людини на фотографії	30.11.2014	
3.	Порівняльний аналіз методів пошуку антропометричних розмірів за існуючим контуром	30.12.2014	
4.	Проектування архітектури розроблюваних програмних засобів	01.03.2015	
5.	Визначення складу та форматів вихідних даних та результатів для кожної програми	09.03.2015	
6.	Програмна реалізація прототипу	13.03.2015	
7.	Перевірка та тестування моделі	16.04.2015	
8.	Розробка клієнт-серверного додатку	20.04.2015	
9.	Підготовка графічної частини дипломної роботи	25.04.2015	
10.	Оформлення документації дипломної роботи	20.05.2015	

Студент \_\_\_\_\_ Барбинягра А.М.

Керівник роботи \_\_\_\_\_ Сирота С.В.

## АНОТАЦІЯ

Дана дипломна робота виконана на здобуття освітньо-кваліфікаційного рівня “Бакалавр” та присвячена розробці математичного та програмного забезпечення автоматичної системи підбору розмірів одягу за фотографією.

Метою роботи є створення програмного забезпечення для підбору розмірів одягу та визначення антропометричних розмірів людини за фотографією на основі клієнт-серверної архітектури для використання у реальних задачах.

В рамках дипломної роботи проведено аналіз існуючих методів розпізнавання контурів людини на зображенні, а також визначення антропометричних показників за існуючим контуром. Було виділено їх недоліки та переваги.

На основі отриманих результатів було розроблено власний математичний метод, розроблено його програмну реалізацію, яку було протестовано. Тестування показало, що похибка розробленого алгоритму – 1%, що є більш ніж достатнім для поставленої задачі.

Робота виконана на 74 аркушах, має посилання на список використаних літературних джерел з 22 найменувань, а також наведено 16 рисунків, 2 таблиць та 7 додатків.

Ключові слова: хромакей, контур, антропометрія, людина, одяг

## ABSTRACT

This aim of this thesis is to develop the mathematical and software parts of the automatic clothes sizes recommendation system.

The main goal of this work is to develop the system for the automatic clothes sizes recommendation and anthropometric sizes recognition of the human on the photo. The system should be based on the client-server architecture and should have be ready-to-use in real life applications.

The analysis of existing methods of human body recognition and anthropometric measures based on the existing contour was made in the thesis. Positive and negative were analysed

New mathematical method was designed based on the results of analysis. Software imlementation of the method was developed and tested. Testing shown that developed method has 1% error which is more than plausible for the task.

The work consists of 74 pages, contains references to the 22 sources, 16 figures, 2 tables and 7 supplementials.

Keywords: chromakey, contour, anthrompetry, human, clothes

## ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ ТА ТЕРМІНІВ .....	9
ВСТУП .....	10
1 ПОСТАНОВКА ЗАДАЧІ .....	12
2 ОГЛЯД ІСНУЮЧИХ РІШЕНЬ .....	13
2.1 Огляд існуючих рішень з вимірювання антропометричних показників	13
2.1.1 Методи визначення антропометричних показників, засновані на ручних вимірах.....	13
2.1.2 Методи визначення антропометричних показників, засновані на комп'ютерних вимірах, що працюють із 3D-моделями .....	14
2.1.3 Методи визначення антропометричних показників, засновані на комп'ютерних вимірах, що працюють із 2D-зображеннями .....	15
2.2 Огляд технологій розробки.....	19
2.2.1 Огляд мов програмування .....	19
2.2.2 Огляд фреймворків для реалізації серверних аплікацій.....	22
2.3 Висновки.....	24
3 ОПИС МАТЕМАТИЧНОЇ МОДЕЛІ.....	26
3.1 Зміна кольорового простору.....	27
3.2 Розпізнавання контуру .....	29
3.3 Апроксимація контуру .....	30
3.4 Пошук ключових точок.....	31
3.5 Знаходження абсолютних розмірів тіла .....	32
3.6 Знаходження розмірів одягу .....	33
3.7 Висновки.....	33
4 СТРУКТУРА І ОПИС ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ.....	34
4.1 Опис системи.....	34
4.2 Опис функціоналу.....	34
4.3 Керівництво користувача.....	35
4.4 Проектування інтерфейсу користувача.....	35

4.5 Висновки.....	38
5 ТЕСТУВАННЯ КОМПОНЕНТІВ ІНФОРМАЦІЙНОЇ СИСТЕМИ.....	39
5.1 Контрольні приклади.....	39
5.2 Висновки.....	43
ВИСНОВКИ.....	44
СПИСОК ЛІТЕРАТУРИ.....	46



## ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ ТА ТЕРМІНІВ

Chromakey – хромакей, зелений екран

Frontend – частина програмної аплікації, що відповідає за роботу з користувачем

HSV – Hue Saturation Value, кольоровий простір, де колір кожного пікселя визначається через тон, насиченість та яскравість

L – Large, великий

M – Medium, середній

MVC – Model View Controller, архітектурний шаблон

RGB – red green blue, цвітовий простір, де колір кожного пікселя визначається через червоний, зелений та синій кольори

S – Small, малий

XL – Extra Large, екстра великий

ГГц – гігагерц

МБ – мегабайт

## ВСТУП

У наш час, нема людей які хоча б раз на рік купували одяг. Магазины одягу є одним із найуспішніших видів бізнесу – наприклад, станом на 2013, бренд Zara мав 16.743 мільярдів євро виторгу [1].

Але, через велику кількість покупців та велику кількість часу, що потрібна кожному на примірку одягу, виникає проблема зайнятості примірочних. У середньому, покупець проводить по 14 хвилин у примірочній [1].

Завдяки дуже швидкому росту потужності обчислювальної техніки, з'явилась можлива вирішення цієї проблеми за допомогою апарату обробки зображень.

Обробка зображень - будь-яка форма обробки інформації, для якої вхідні дані представлені зображенням, наприклад, фотографіями або відеокадрами. Обробляння зображень може здійснюватися як для отримання зображення на виході (наприклад, підготовка до поліграфічного тиражування, до телетрансляції і т.д.), так і для отримання іншої інформації (наприклад, розпізнання тексту, підрахунок числа і типу клітин в полі мікроскопа і т. д.) [2].

Обробка зображень застосовується у багатьох сферах діяльності – у медицині (наприклад, розпізнавання клітин), у робототехніці тощо. Однією із сфер застосування обробки зображень є антропометрія.

Антропометрія - один з основних методів антропологічного дослідження, який полягає у вимірюванні тіла людини та його частин з метою встановлення вікових, статевих, расових та ін. особливостей фізичної будови.

Потреба в антропометричних дослідженнях зумовлюється великою мінливістю розмірів тіла людини. Межі коливання розмірів людей однієї групи, як правило, заходять за межі коливань розмірів людей іншої групи. Це т. з. трансгресивна мінливість, яка зумовлює необхідність кількісних визначень. Результати антропометричних вимірів порівнюються за спеціально розробленими правилами, що ґрунтуються на принципах варіаційної статистики.

Зібрані в процесі антропометричного обстеження дані піддають варіаційно-статистичній обробці і оформляють у вигляді таблиць, графіків і схем.

Антропометричні методи мають велике значення в прикладній антропології; до широкого впровадження ідентифікації за відбитками пальців антропометрія використовувалась в криміналістиці для ідентифікації людей (так званий «Бетрильонаж»).

Масовість антропометричних досліджень дозволяє оцінювати і порівнювати мінливість ознак різних расових, вікових, професійних, статевих груп на основі вимірів великого числа індивідуумів.

На антропометричних матеріалах засновані також стандартизація предметів масового виробництва (одяг, взуття) [3].

## 1 ПОСТАНОВКА ЗАДАЧІ

Метою даного дипломної роботи є створення математичного та програмного забезпечення системи визначення розмірів одягу людини за фотографією.

Фотографія має містити на собі людину, що стоїть рівно, з витягнутими перпендикулярно тілу руками. Також зображення має містити на собі червоний квадрат розміром 15x15 сантиметрів. Фон зображення має бути зеленого кольору.

Математичне забезпечення має виконувати наступні функції:

- а) пошук контуру людини на фотографії;
- б) пошук антропометричних розмірів людини;
- в) підбір розміру одягу для людини.

Програмне забезпечення має надавати користувачеві функції завантаження зображення з користувачем.

Вихідними параметрами програмного забезпечення є:

- а) розміри одягу (наприклад, S, M, L, тощо);
- б) антропометричні розміри людини.

Всі розміри одягу мають бути приведені до європейської системи розмірів одягу.

Програмне забезпечення має виконуватися в операційній системі Windows 7 на IBM-сумісному комп'ютері, до складу якого входять:

- а) x86-сумісний процесор із тактовою частотою, не меншою за 1,5 ГГц;
- б) оперативна пам'ять обсягом, не меншим за 256 МБ.

## 2 ОГЛЯД ІСНУЮЧИХ РІШЕНЬ

### 2.1 Огляд існуючих рішень з вимірювання антропометричних показників

Алгоритми визначення антропометричних розмірів поділяють на два типи: засновані на ручних або комп'ютерних обчисленнях.

#### 2.1.1 Методи визначення антропометричних показників, засновані на ручних вимірах

Вимір діаметрів тіла проводиться великим циркулем. За його допомогою вимірюються: акроміальний діаметр (ширина плечей) – відстань між правою і лівою акроміальними точками; вертикальний діаметр – відстань між найбільш виступаючими точками великих вертелів стегнових кісток.

Так звані обхватні розміри тіла людини вимірюються сантиметровою стрічкою, площина якої розташовується паралельно площині тіла, а її нульовий розподіл знаходиться попереду випробуваного. Вимірюються обхват голови, грудей, талії, таза (через сідниці), стегна, гомілки, плеча, передпліччя.

Одним із найважливіших показників, що характеризують фізичний розвиток, вважають площу поверхні тіла. Серед множини методів її визначення

найбільш популярні аналітичні методи, що передбачають використання формул Boyd і Jssakson.

На думку багатьох дослідників, площу поверхні тіла як ознаку фізичного розвитку доцільно розглядати не в абсолютних значеннях, а у відносних, у співвідношенні з вагою тіла (при цьому визначається розмір ваги, що припадає на одиницю поверхні). У фізично слабких осіб на одиницю площі поверхні тіла припадає менше ваги, ніж у фізично сильних.

У 1964 Л.К. Щекочихіна на підставі ретельного аналізу парних і множинних кореляцій тотальних розмірів тіла з площами поверхонь сегментів тіла, розрахованих на великих вибірках чоловіків і жінок, установила залежність площі поверхні окремих частин тіла від довжини й ваги тіла одночасно. Виявлена залежність дозволила їй розрахувати рівняння множинної регресії для визначення площ поверхонь окремих частин тіла по двох тотальних розмірах тіла, а також розробити номограми [3].

### 2.1.2 Методи визначення антропометричних показників, засновані на комп'ютерних вимірах, що працюють із 3D-моделями

Методи визначення антропометричних показників, засновані на комп'ютерних вимірах поділяються на два типи: ті, що засновані на трьовимірних обчисленнях та потребують спеціального обладнання, такого як 3D-сканери, а також ті, що працюють зі звичайними двовимірними зображеннями.

Найбільш популярними 3D-сканерами є сканери, розроблені компанією Cyberware. Наприклад, Cyberware WB-4 [4].



Рисунок 2.1 - Cyberware WB-4

### 2.1.3 Методи визначення антропометричних показників, засновані на комп'ютерних вимірах, що працюють із 2D-зображеннями

Але найбільша потреба є у алгоритмах, що працюють зі звичайними 2D-зображеннями, фотографіями.

Прикладом такого алгоритму є алгоритм, запропонований у [5]. Для вимірювання використовуються фотографії, зроблені за допомогою фотокамери Canon IXUS з розширенням 1600x1200 пікселів. За допомогою камери роблять 3 фотографії – фронтальної частини тіла, бокової та тилової.

Після цього контур тіла апроксимується за допомогою прямокутників та овалів, якими і вимірюються антропометричні показники.

Для визначення точних значень тіла, на фотографії розміщено прямокутник, розміром 15x15 сантиметрів, за допомогою якого знаходиться масштаб точних розмірів до розмірів на фотографії.

Інший підхід запропоновано у [4]. Автори використовують зображення з розширенням 480x640 пікселів. Алгоритм використовує заздалегідь заданий контур людського тіла, що має декілька заздалегідь розміщених ключових точок, таких як: верхня точка голови, найбільш ліва та права точки рук, найбільш нижні точки ніг. Всі інші точки, що потребуються алгоритмом, користувач виставляє вручну.

Метод, запропонований у [6], складається з двох етапів – пошуку ключових точок та пошуку характеристичних точок, на основі ключових. В ключові точки входять точки рук, ніг та голови. В характеристичні точки входять точки паху, пахв та плечей.

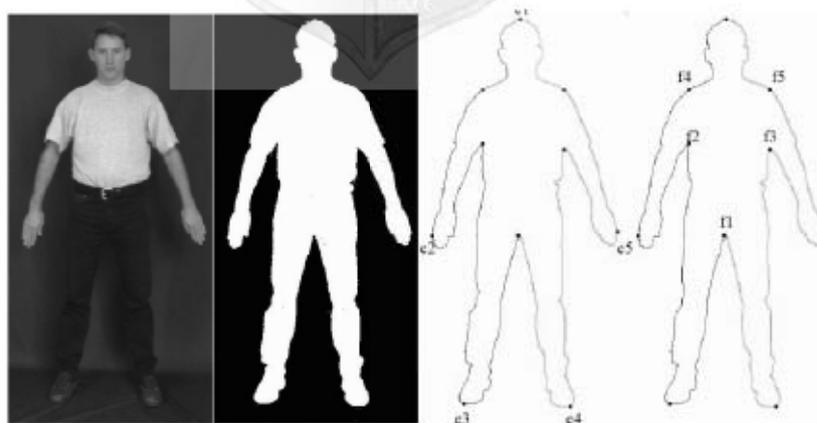


Рисунок 2.2 - Точки алгоритму, запропонованого у [7].

У методі, запропонованому у запропоновано вимірювання антропометричних показників за допомогою канону Колмана [7].

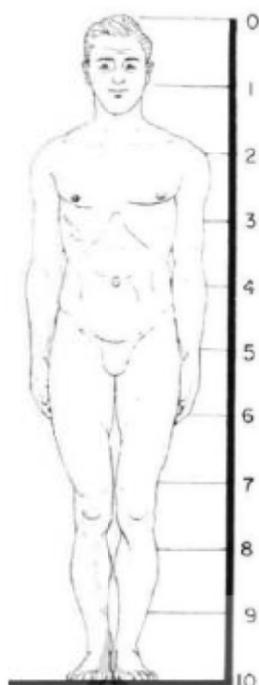


Рисунок 2.3 - Канон Колмана

Тіло людини поділяють на десять рівних частин. На основі експериментальних даних було отримано, що, наприклад, рука людини займає довжину 4 відрізків (2-6), а нога – 5 (5-10).

Для визначення точних розмірів, на фоні розміщено 6 червоних точок, відстань між якими заздалегідь відома.

Наведемо таблицю з порівнянням алгоритмів, що працюють із звичайними фотографіями:

Таблиця 2.1 - Коротке порівняння алгоритмів, що працюють з фотографіями

Автори методу	Короткий опис	Мінуси
Seo, H., Yeo, Y., Wohn, K.	Заздалегідь відомий контур, що підставляється під людину. Червона карта 15x15 см, для визначення точних розмірів	Низька точність, потребує людської взаємодії
Hung, P. C.-Y., Witana, C. P., Goonetilleke	Виділяються 5 ключових точок (руки, ноги, голова), за якими знаходяться характеристичні	Потребує людської взаємодії для калібровки камери для отримання точних розмірів
Tomáš Kohlschütter	Використовується канон Колмана, тобто, тіло ділиться на 10 рівних частин	Низька точність
Criminisi, A., Zisserman, A., van Gool, L. J., Bramble, S. K., Compton	Користувач виставляє точки на зображенні руками	Потребує взаємодії з людиною

## 2.2 Огляд технологій розробки

### 2.2.1 Огляд мов програмування

Python — інтерпретована об'єктно-орієнтована мова програмування високого рівня з динамічною семантикою. Розроблена в 1990 році Гвідо ван Россумом. Структури даних високого рівня разом із динамічною семантикою та динамічним зв'язуванням роблять її привабливою для швидкої розробки програм, а також як засіб поєднання існуючих компонентів. Python підтримує модулі та пакети модулів, що сприяє модульності та повторному використанню коду. Інтерпретатор Python та стандартні бібліотеки доступні як у скомпільованій так і у вихідній формі на всіх основних платформах. В мові програмування Python підтримується декілька парадигм програмування, зокрема: об'єктно-орієнтована, процедурна, функціональна та аспектно-орієнтована [8].

Переваги застосування для поставленої задачі:

- а) велика кількість документації;
- б) велика кількість фреймворків та бібліотек для наукових обчислень.

Недоліки застосування для поставленої задачі:

- а) потребує встановлення віртуальної машини.

Ruby - це інтерпретована, повністю об'єктно-орієнтована мова програмування з чіткою динамічною типізацією. Мова відрізняється високою ефективністю розробки програм і увібрала в себе кращі риси Perl, Java, Python, Smalltalk, Eiffel, Ada і Lisp. Ruby поєднує в собі Perl-подібний синтаксис із

об'єктно-орієнтованим підходом мови програмування Smalltalk. Також деякі риси запозичено із мов програмування Python, Lisp, Dylan та CLU.

Багатоплатформова реалізація інтерпретатора мови Ruby поширюється як вільне програмне забезпечення. Сирцевий код проекту розповсюджується під ліцензіями BSD ("2-clause BSD") і "Ruby", яка посилається на останній варіант ліцензії GPL і повністю сумісна з GPLv3 [9].

Переваги застосування для поставленої задачі:

- а) велика кількість документації;
- б) велике ком'юніті.

Недоліки застосування для поставленої задачі:

- а) потребує встановлення віртуальної машини;
- б) занадто повільний для наукових обчислень.

C++ — мова програмування високого рівня з підтримкою декількох парадигм програмування: об'єктно-орієнтованої, узагальненої та процедурної. Розроблена Б'ярном Страуструпом в AT&T Bell Laboratories (Мюррей-Хілл, Нью-Джерсі) у 1979 році та початково отримала назву «Сі з класами». Згодом Страуструп перейменував мову у C++ у 1983 р. Базується на мові С. Визначена стандартом ISO/IEC 14882:2003.

У 1990-х роках C++ стала однією з найуживаніших мов програмування загального призначення. Мову використовують для системного програмування, розробки програмного забезпечення, написання драйверів, потужних серверних та клієнтських програм, а також для розробки розважальних програм таких як відеоігри. C++ суттєво вплинула на інші, популярні сьогодні, мови програмування: C# та Java [10].

Переваги застосування для поставленої задачі:

- а) швидкість;
- б) велика кількість бібліотек.

Недоліки застосування для поставленої задачі:

- а) не є мультиплатформеним;
- б) досить складний.

Java — об'єктно-орієнтована мова програмування, випущена компанією Sun Microsystems у 1995 році як основний компонент платформи Java. Зараз мовою займається компанія Oracle, яка придбала Sun Microsystems у 2009 році. Синтаксис мови багато в чому схожий на C та C++. У офіційній реалізації, Java програми компілюються у байт-код, який при виконанні інтерпретується віртуальною машиною для конкретної платформи.

Oracle надає компілятор Java та віртуальну машину Java, які задовольняють специфікації Java Community Process, під ліцензією GNU General Public License.

Мова значно запозичила синтаксис із C і C++. Зокрема, взято за основу об'єктну модель C++, проте її модифіковано. Усунуто можливість появи деяких конфліктних ситуацій, що могли виникнути через помилки програміста та полегшено сам процес розробки об'єктно-орієнтованих програм. Ряд дій, які в C/C++ повинні здійснювати програмісти, доручено віртуальній машині. Передусім, Java розроблялась як платформи-незалежна мова, тому вона має менше низькорівневих можливостей для роботи з апаратним забезпеченням. За необхідності таких дій java дозволяє викликати підпрограми, написані іншими мовами програмування [11].

Переваги застосування для поставленої задачі:

- а) велика кількість готових рішень;
- б) популярність.

Недоліки застосування для поставленої задачі:

- а) потребує встановлення віртуальної машини;
- б) повільна.

## 2.2.2 Огляд фреймворків для реалізації серверних аплікацій

Sinatra — об'єктно-орієнтований програмний каркас для створення веб-застосунків, написаний на мові програмування Ruby.

Sinatra є відкритим програмним забезпеченням і розповсюджується під ліцензією MIT.

Розроблена та створена Blake Mizerany, Sinatra є крихітною (близько 1500 рядків коду, що майже 1/100-а розміру Rails) та надзвичайно гнучкою. Вона не слідує типовій архітектурі Модель-Вид-Контролер (Model-View-Controller, MVC), яка використовується в інших фреймвоках, таких як Ruby on Rails. Натомість, Sinatra зосереджена на "швидкому створенню веб-додатків на Ruby із мінімальними зусиллями". Sinatra використовує всі можливості Ruby і є його оригінальним розширенням [12].

Переваги застосування для поставленої задачі:

- а) популярність;
- б) легкість.

Недоліки застосування для поставленої задачі:

- а) досить складним є написання великих додатків.

Ruby on Rails — об'єктно-орієнтований програмний каркас (фреймворк) для створення веб-застосунків, написаний на мові програмування Ruby. Ruby on Rails надає каркас модель-вид-контролер (Model-View-Controller) для веб-застосунків, а також забезпечує їхню інтеграцію з веб-сервером і сервером бази даних.

Ruby on Rails був створений Девідом Гайнемаєр Генссоном (англ. David Heinemeier Hansson) на основі його роботи над засобом керування проектами

Basecamp і був випущений в липні 2004 року. Ruby on Rails є відкритим програмним забезпеченням і розповсюджується за ліцензією MIT [13].

Переваги застосування для поставленої задачі:

- а) MVC;
- б) велике ком'юніті;
- в) велика кількість документації.

Недоліки застосування для поставленої задачі:

- а) є дуже важким рішенням – займає багато пам'яті.

Django (Джанго) — високорівневий відкритий Python-фреймворк для розробки веб-систем. Названо його було на честь джазмена Джанго Рейнхардта (відповідно до музичних смаків одного зі засновників проекту).

Сайт на Django будується з однієї або декількох частин, які рекомендується робити модульними. Це одна з істотних архітектурних відмінностей цього фреймворку від деяких інших (наприклад Ruby on Rails).

Архітектура Django подібна на «Модель-Вид-Контролер» (MVC). Однак, те що називається «контролером» в класичній моделі MVC, в Django називається «вид» (англ. view), а те, що мало б бути «видом», називається «шаблон» (англ. template). Таким чином, MVC розробники Django називають MTV («Модель-Шаблон-Вид»). [14]

Переваги застосування для поставленої задачі:

- а) велика кількість документації;
- б) популярність.

Недоліки застосування для поставленої задачі:

- а) специфічна архітектура.

PyLons – високорівневий модульний веб-фреймворк написаний на мові Python. Він призначений для швидкої розробки веб-сайтів та порталів, а також для створення простих веб-інтерфейсів. Фреймворк не є цілісним, а

складається з багатьох різних відкритих компонентів, які в разі потреби можна замінити аналогами [15].

Переваги застосування для поставленої задачі:

а) модульність;

Недоліки застосування для поставленої задачі:

а) специфічна архітектура;

б) мала кількість документації.

### 2.3 Висновки

Для реалізації математичного методу було обрано мову програмування Python та фреймворк OpenCV.

OpenCV (англ. Open Source Computer Vision Library, бібліотека комп'ютерного зору з відкритим вихідним кодом) — бібліотека функцій та алгоритмів комп'ютерного зору, обробки зображень і чисельних алгоритмів загального призначення з відкритим кодом. Бібліотека розроблена Intel і нині підтримується Willow Garage та Itseez. Реалізована на C / C++, також розробляється для Python, Java, Ruby, Matlab, Lua та інших мов. Може вільно використовуватися в академічних та комерційних цілях — поширюється на умовах ліцензії BSD [16].

Головним моментом у виборі було те, що Python є досить швидкий для роботи із зображеннями у реальному часі, а також велика кількість навчальних матеріалів. Фреймворк OpenCV було обрано через відсутність конкурентів.

Для реалізації веб-додатку було обрано мову програмування Ruby, а також фреймворк Sinatra. Вибір обґрунтовано легковісністю фреймворку

Sinatra (наприклад, на відміну від інших фреймворків, для підключення Sinatra до проекту потрібна лише 1 строчка коду), але, так як Sinatra є лише під мову програмування Ruby, було вирішено обрати її.

Для написання frontend-частини веб-аплікації, було обрано фреймворк Twitter Bootstrap.

Bootstrap — це набір інструментів від Twitter (відноситься до класу інструментів: CSS-фреймворк), створений для полегшення розробки web застосунків та сайтів. Він включає CSS та HTML для типографії, форм, кнопок, таблиць, сіток, навігації тощо, а також додаткові розширення JavaScript.

Репозиторій з фреймворком є одним з найбільш популярних на GitHub і, серед інших, його використовують NASA і MSNBC [17].

Вибір обумовлено через відсутність значних конкурентів, велику кількість документації, а також попередній стаж праці з ним.

Вибраний для розробки математичний метод є поєднанням методів [4, 5, 7]. Для пошуку контуру людини на зображенні було обрано технологію хромакей [7]. Для пошуку відносних антропометричних вимірів вирішено використовувати ключові точки [5]. Для пошуку абсолютних розмірів вирішено використати порівняння з відносними розмірами об'єкта, абсолютні розміри якого відомі заздалегідь (наприклад, червоний квадрат зі стороною 15см).

### 3 ОПИС МАТЕМАТИЧНОЇ МОДЕЛІ

Обраний для реалізації метод було побудовано на основі методів, запропонованих у [4, 5, 7].

Алгоритм складається з трьох етапів: розпізнавання контуру, знаходження відносних розмірів та знаходження абсолютних розмірів.

Для розпізнавання контуру існують два підходи. Перший – Background subtraction. Його засновано на порівнянні двох фотографій – однієї з чистим фоном і другої, з об'єктом на цьому ж фоні.

Для цього шукають об'єкти на фоновому зображенні за наступною формулою: ( $I$  – інтенсивність пікселя з координатами  $x$  та  $y$ ,  $B$  – інтенсивність пікселя на зображенні з чистим фоном з тими ж координатами,  $T$  – threshold, максимальний поріг)

$$|I(x, y) - B(x, y)| > T \quad (3.1)$$

Другий – chromakey. У даному підході використовується зеленій (або синій) екран.

У даній дипломній роботі було обрано другий метод, тому ще він потребує лише однієї фотографії, а також є більш зручним для використання у місцях, де камера є зафіксованою – наприклад, у магазинах одягу.



Рисунок 3.1 - Приклад вихідного зображення

### 3.1 Зміна кольорового простору

Вхідна фотографія конвертується з кольорового простору RGB (Red Green Blue) до кольорового простору HSV (Hue Saturation Value) для того, щоб була можливість виділення маски зображення.

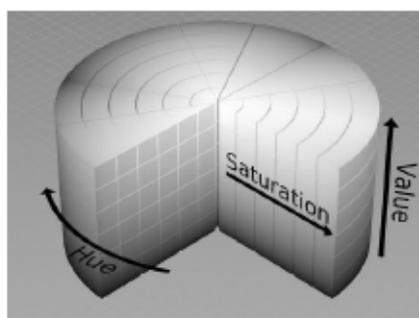


Рисунок 3.2 - Кольоровий простір HSV

Конвертування виконується за наступними формулами [18]:

$$R' = \frac{R}{255}, G' = \frac{G}{255}, B' = \frac{B}{255}$$

$$C_{max} = \max(R', G', B'), C_{min} = \min(R', G', B') \quad (3.2)$$

$$\Delta = C_{max} - C_{min}$$

Для пошуку тону (Hue), насиченості (Saturation) та яскравості (Value) використовуються наступні формули:

$$H = \begin{cases} 60^\circ * \left( \frac{G' - B'}{\Delta} \text{mod} 6 \right), C_{max} = R' \\ 60^\circ * \left( \frac{B' - R'}{\Delta} + 2 \right), C_{max} = G' \\ 60^\circ * \left( \frac{R' - G'}{\Delta} + 4 \right), C_{max} = B' \end{cases} \quad (3.3)$$

$$S = \begin{cases} 0, C_{max} = 0 \\ \frac{\Delta}{C_{max}}, C_{max} \neq 0 \end{cases} \quad (3.4)$$

$$V = C_{max} \quad (3.5)$$

Отримані значення HSV кожного з пікселів оригінального зображення формують нове зображення.

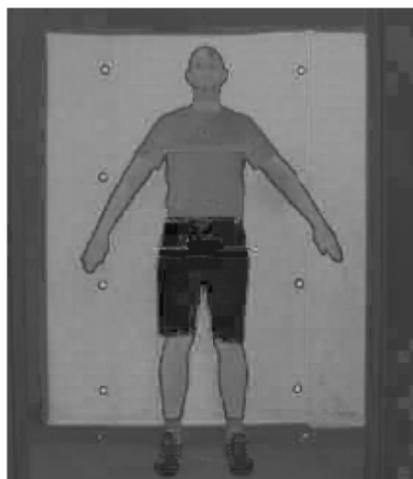


Рисунок 3.3 - Фотографія у форматі HSV.

### 3.2 Розпізнавання контуру

За отриманим зображенням будується так звана «бінарна маска». Тобто всі сегменти зображення, що не входять в задану множину фарбуються чорним кольором, а всі інші – білим.



Рисунок 3.4 - Приклад бінарної маски

По отриманій масці виділяють контури. Для пошуку контурів використовується алгоритм, запропонований у [19].

### 3.3 Апроксимація контуру

Знайдений контур має дуже велику кількість точок, адже людське тіло має багато кутів. З таким контуром важко працювати, тому зменшення кількості точок на контурі, він апроксимується за допомогою полігонів. Для цього було обрано метод Дугласа-Пекера [20]. Він працює наступним чином.

Контур представляє собою масив впорядкованих, послідовно поєднаних точок. Він рекурсивно ділиться на підконтури – точками входу слугують усі точки, що знаходяться між першою та останньою. Після цього будується відрізок між вхідною та останньою точками контуру. За знайденим відрізком шукають відстань від відрізка до кожної з вершин за наступною формулою ( $V$  – відповідна координата вершини,  $i$  – номер початкової точки,  $j$  – номер кінцевої точки,  $k$  – номер точки, відстань від якої шукають у даний момент):

$$dist = \left( \frac{\begin{vmatrix} 1 & V_i x & V_i y \\ 1 & V_j x & V_j y \\ 1 & V_k x & V_k y \end{vmatrix}}{(V_i x + V_j x)^2 + (V_i y + V_j y)^2} \right)^{1/2} \quad (3.6)$$

Усі точки, відстань яких більша за заданий поріг, відкидаються, а алгоритм повторюється рекурсивно для всіх точок контуру, починаючи від першої [21].

### 3.4 Пошук ключових точок

За алгоритмом [7], контур містить 5 ключових точок – найбільш ліва та найбільш права точки, тобто кінці рук, найбільш висока точка – кінець голови, а також найбільш нижні точки лівої та правої половин тіла, тобто кінці ніг.

За допомогою знайдених ключових точок виконується пошук характеристичних точок, тобто точок пахв, паху, плечей та кінців шиї.

При їх пошуку використовується канон Колмана, тобто метод, запропонований у [4].

Точки плечей є максимумами контурів, що знаходиться між відповідними точкою руки та ноги.

Точки шиї є найбільш лівою та найбільш правою точками тіла на другій частині тіла, відповідно до канону Колмана.

Точки пахв є найбільш високими точками на контурі, що знаходяться на третій частині тіла.



Рисунок 3.5 - Приклад знайдених точок

### 3.5 Знаходження абсолютних розмірів тіла

Для знаходження абсолютних розмірів тіла, використовується спосіб, запропонований у [5]. На фоні закріплюється червона картка розміром 15x15 сантиметрів, відносні розміри якої знаходяться за фотографією. Після цього за наступною формулою, знаходиться її масштаб, який використовується для знаходження абсолютних розмірів людського тіла:

$$\frac{15 \text{ см}}{\text{довжина сторони картки}} = \frac{\text{справжній розмір}}{\text{розмір на зображенні}} \quad (3.7)$$

### 3.6 Знаходження розмірів одягу

Для знаходження розмірів одягу використовуються таблиці, що містять відомі значення.

Приведемо приклад розмірів футболок [22]:

Таблиця 3.1 - Розміри футболок

Розмір	Висота (см)	Ширина (см)
S	69	50
M	72	503
L	74	56
XL	75	59

### 3.7 Висновки

У цьому розділі було визначено математичну модель та головний алгоритм. Для пошуку контуру було використано технологію хромакей. Знайдений контур апроксимується полігонами за допомогою методу Дугласа-Пекера, після чого на ньому знаходяться ключові точки – руки, ноги, голова, пахви, плечі.

Для визначення точних розмірів було вирішено використати червоний квадрат зі стороною 15см, що розміщено на фотографії.

Розміри одягу знаходяться за допомогою порівняння знайдених антропометричних показників із табличними значеннями.

## 4 СТРУКТУРА І ОПИС ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

### 4.1 Опис системи

Для реалізації системи було обрано мову програмування Python та фреймворк OpenCV.

Інтерфейс програмного засобу буде спроектовано у вигляді веб-додатку, використовуючи мови програмування JavaScript, Ruby, а також мову розмітки HTML. Для написання серверного коду додатку, було використано фреймворк Sinatra.

### 4.2 Опис функціоналу

Головною функцією системи є рекомендація розмірів одягу. Система приймає на вхід фотографію користувача, що стоїть в заданій позі на зеленому фоні. На вихід система виводить антропометричні показники людини, а також рекомендований розмір одягу.



### 4.3 Керівництво користувача

Для користування системою АСПРО, користувачу потрібно мати фотографію із зображенням людини на зеленому фоні, що стоїть у позі з витягнутими, нахиленими вниз руками. Зображення повинно мати фон зеленого кольору. Допустимими типами зображень є \*.jpeg, \*.jpg або \*.png.

Для початку роботи, користувач повинен натиснути кнопку "Завантажити файл" та обрати відповідний файл, після чого натиснути кнопку "Завантажити".

У разі, якщо на зображенні не було знайдено людину, або людина стоїть у іншій, аніж заданій, позі, програмний продукт повідомить про це відповідним текстовим повідомленням на екрані.

### 4.4 Проектування інтерфейсу користувача

Інтерфейс користувача було розроблено за допомогою мови розмітки HTML, а також каскадної таблиці стилів (CSS). Для покращення якості інтерфейсу було використано frontend framework Bootstrap.

Інтерфейс користувача містить 4 вікна – початкове вікно, через яке користувач завантажує фотографію, вікно для відображення помилки, вікно з результатами, а також вікно з керівництвом користувача.

При початку роботи із системою, користувач попадає на головне вікно

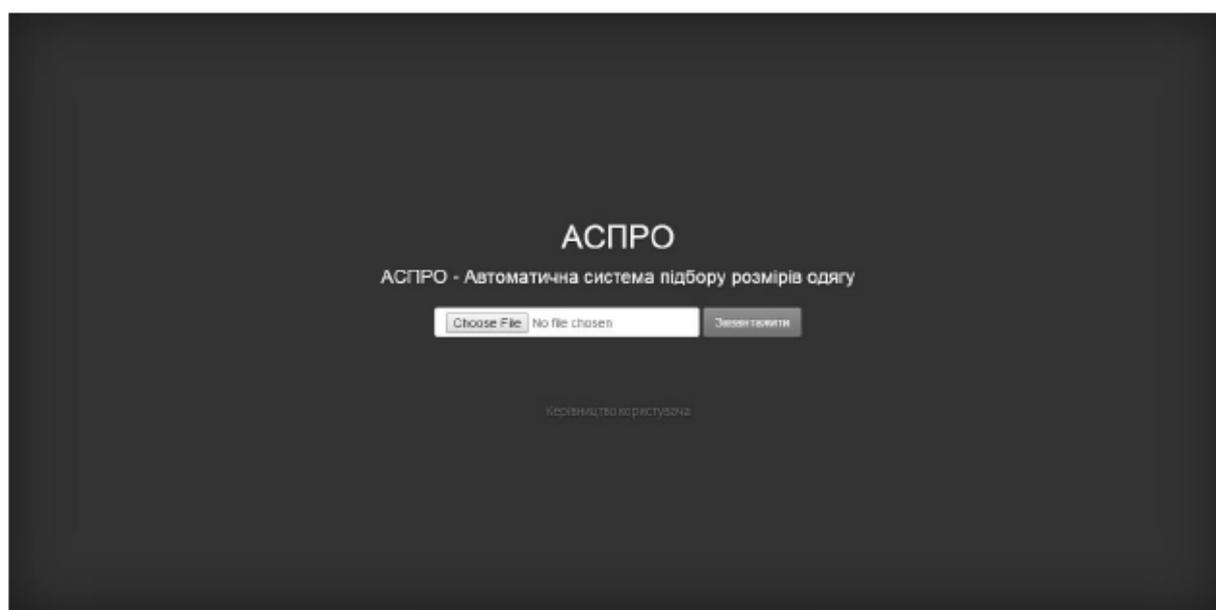


Рисунок 4.1 - Головне вікно

Після обрання зображення користувач натискає на кнопку «Завантажити» та, у разі якщо зображення було неправильним, попадає на вікно із помилкою

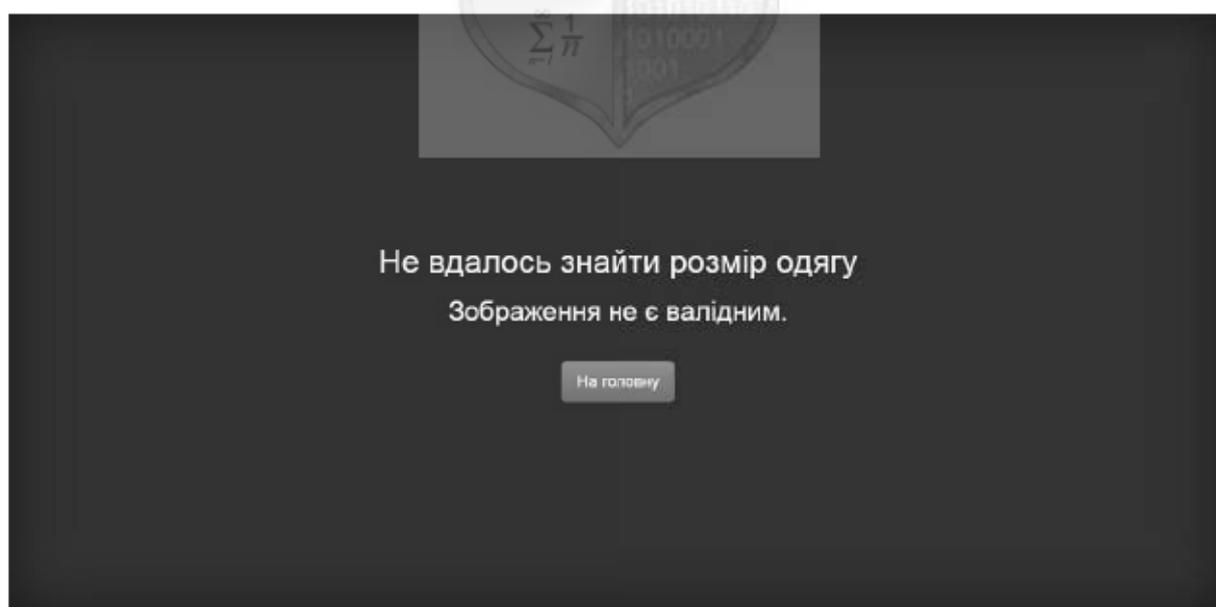


Рисунок 4.2 - Приклад екранної форми з повідомленням про помилку

Або, якщо зображення було вірним, попадає на сторінку з результатами аналізу

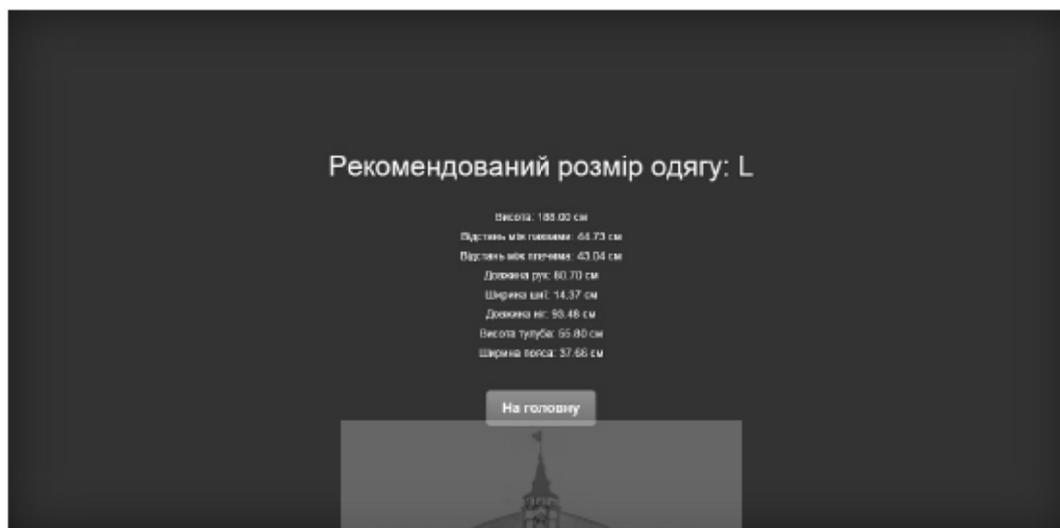


Рисунок 4.3 - Вікно з результатами

У разі якщо користувач натискає кнопку «Керівництво користувача», користувач потрапляє на відповідну сторінку

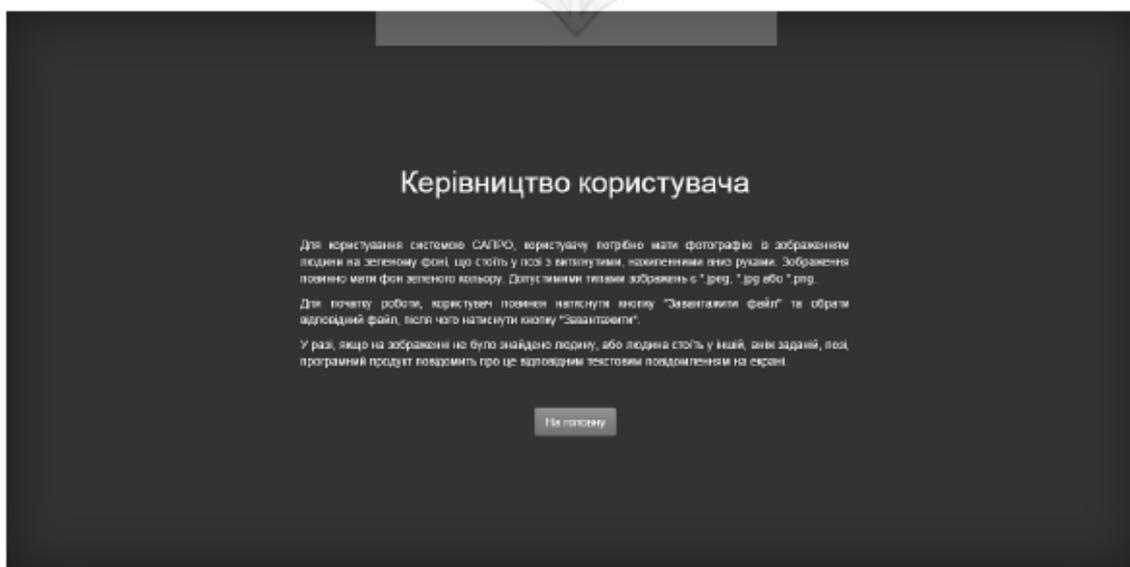


Рисунок 4.4 - Керівництво користувача

## 4.5 Висновки

При розробці програмного забезпечення було використано дві мови програмування – Ruby та Python. Python було використано при реалізації алгоритмічної частини. У зв'язці з Python було використано фреймворк OpenCV. Це дозволило виконувати всі обчислення при швидкості, що є близькою до реального часу.

Мову програмування Ruby було використано при розробці веб-аплікації. Разом з Ruby було використано фреймворк Sinatra. Це дозволило використати клієнт-серверну архітектуру, а також MVC-модель у аплікації.

Для розробки клієнтської частини було використано мову розмітки HTML, а також каскадні таблиці стилів CSS. Завдяки цьому, користувачі можуть користуватись аплікацією з будь-якого пристрою з будь-якої точки світу.

## 5 ТЕСТУВАННЯ КОМПОНЕНТІВ ІНФОРМАЦІЙНОЇ СИСТЕМИ

У цьому розділі на чотирьох фотографіях (5.1.1-5.1.4) наведено кілька прикладів результатів визначення розмірів одягу, а також антропометричних показників.

### 5.1 Контрольні приклади

На першій фотографії знаходиться людина, що стоїть потрібній позі.  
Зріст – 192 см.



Рисунок 5.1 - Тестова фотографія 1

Результати аналізу першого зображення:

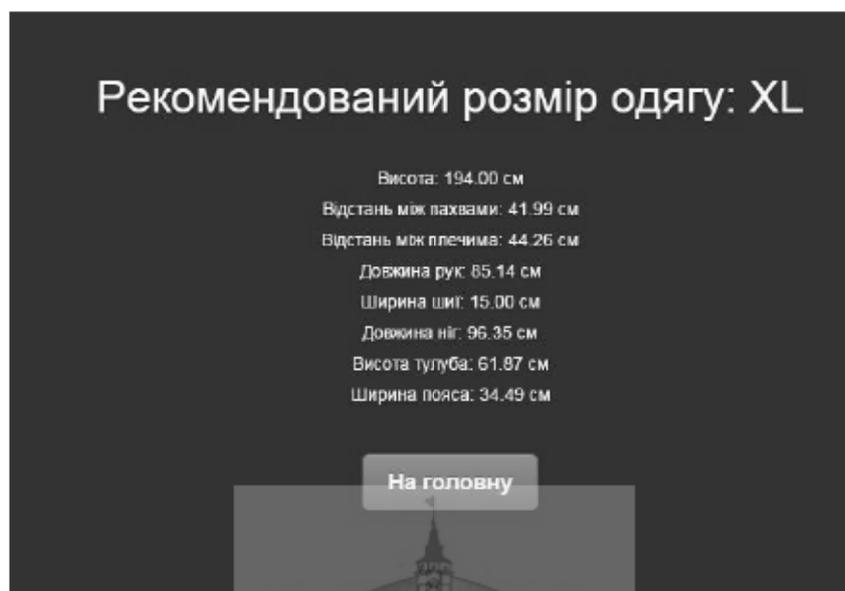


Рисунок 5.2 – Результати аналізу тестової фотографії 1

Зображення розпізналось вірно. Похибка – 1%

На другій фотографії знаходиться людина, що стоїть потрібній позі.  
Зріст – 186 см.



Рисунок 5.3 - Тестова фотографія 2

Результати аналізу другого зображення:

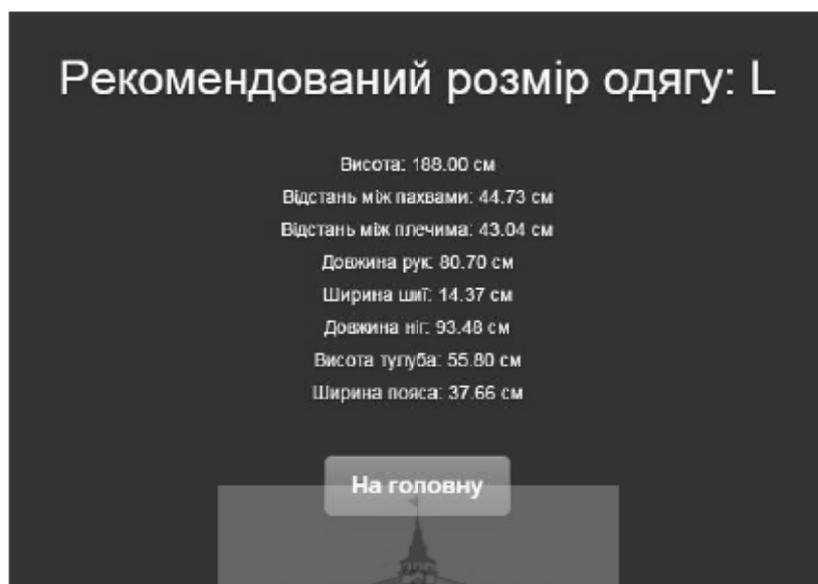


Рисунок 5.4 – Результати аналізу тестової фотографії 2

Зображення розпізналось вірно. Похибка – 1%

На третій фотографії знаходиться людина, що прийняла іншу від потрібної позу.



Рисунок 5.5 - Тестова фотографія 3

Результати аналізу третього зображення:

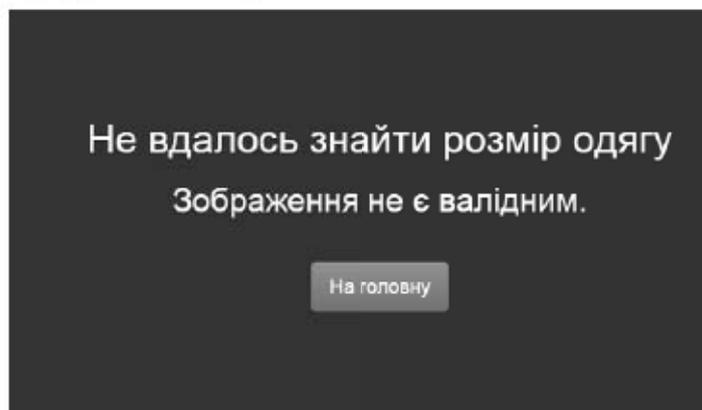


Рисунок 5.6 – Результати аналізу тестової фотографії 3

Людина розпізналась на зображенні, але через те, що вона стоїть у неправильній позі – її розміри розпізнати не вдалось. Для того, щоб це виправити, людина повинна зайняти вірну позу.

На четвертій фотографії знаходиться людина, що стоїть потрібній позі, але фонове зображення не є зеленим.



Рисунок 5.7 - Тестова фотографія 4

## Результати аналізу четвертого зображення

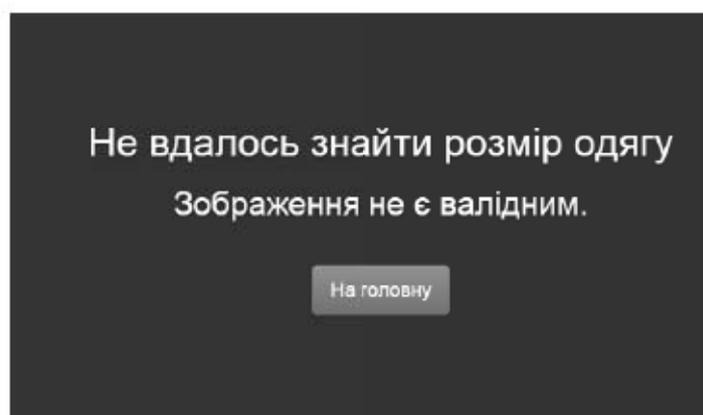


Рисунок 5.8 – Результати тестової фотографії 4

Людину не вдалось розпізнати на зображенні через те, що фонове зображення не є зеленим. Для того, щоб людині вдалось розпізнати – потрібно, щоб вона стала на зеленому фоні.

## 5.2 Висновки

Результати тестування показали, що розроблене математичне забезпечення мають досить високу точність, яка є більш ніж достатня для поставленої задачі. Порівняння реальних розмірів людей з даними, що були отримані системою показало, що система має точність близьку до 1%. Наприклад, реальний зріст людини на 4 тестовій фотографії – 186см, тоді як система оцінила його у 188см.

## ВИСНОВКИ

У даній роботі проведено огляд існуючих методів визначення антропометричних показників людини, включаючи ручні методи, методи що використовують обладнання для побудови 3D-моделі, а також методи, що працюють із 2D-зображеннями (фотографіями).

На основі розглянутих методів було побудовано власний.

Алгоритм було засновано на основі трьох методів, запропонованих у [4, 5, 7] через те, що вони працюють із звичайними фотографіями, не потребують додаткових пристроїв, а також потребують мінімальної взаємодії з користувачем, тобто, є найбільш автоматизованими.

Для розпізнавання контуру людини було обрано алгоритм, що працює із хромакей-зображеннями, тобто із зображеннями, що містять зелений фон. Хоча це призводить до потреби у додатковому обладнанні, але завдяки цьому точність пошуку контуру є максимально точним.

Для апроксимації контуру було обрано алгоритм Дугласа-Пекера [20]. Завдяки цьому швидкість апроксимації контуру є близькою до реального часу, а всі обчислення, що працюють з контуром – виконуються набагато швидше.

На основі побудованого алгоритму було написано програмне забезпечення, після чого було проведено його тестування.

Для написання алгоритмічної частини було використано мову Python та фреймворк OpenCV.

При написанні веб-аплікації було використано мову програмування Ruby та фреймворк Sinatra. Завдяки цьому ПЗ є дуже легким та дозволяє користувачам з будь-якої точки світу користуватись системою.

Тестування показало, що розроблене математичне та програмне забезпечення мають досить високу точність (близько 1%), яка є більш ніж достатньою для задачі рекомендації розмірів одягу.



## СПИСОК ЛІТЕРАТУРИ

1. Zara Annual Report, 2013 [Електронний ресурс] - Режим доступу: [http://www.inditex.com/documents/10279/18789/Inditex\\_Group\\_Annual\\_Report\\_2013.pdf/88b623b8-b6b0-4d38-b45e-45822932ff72](http://www.inditex.com/documents/10279/18789/Inditex_Group_Annual_Report_2013.pdf/88b623b8-b6b0-4d38-b45e-45822932ff72)
2. Бонч-Бруєвич Г.Ф., Носенко Т.І. Організація та обробка електронної інформації / Бонч-Бруєвич Г.Ф., Носенко Т.І // Київ – 2013 – С. 5-6.
3. Антропометрія [Електронний ресурс] - Режим доступу: [http://dic.academic.ru/dic.nsf/enc\\_medicine/2815/%D0%90%D0%BD%D1%82%D1%80%D0%BE%D0%BF%D0%BE%D0%BC%D0%B5%D1%82%D1%80%D0%B8%D1%8F/](http://dic.academic.ru/dic.nsf/enc_medicine/2815/%D0%90%D0%BD%D1%82%D1%80%D0%BE%D0%BF%D0%BE%D0%BC%D0%B5%D1%82%D1%80%D0%B8%D1%8F/)
4. Seo, H., Yeo, Y., Wohn, K. 3D body reconstruction from photos based on range scan. Lecture Notes in Computer Science / Seo, H., Yeo, Y., Wohn, K. // China – 2006 – С. 849-860
5. Hung, P. C.-Y., Witana, C. P., Goonetilleke, R. S. Anthropometric Measurements from Photographic Images. Computing Systems / Hung, P. C.-Y., Witana, C. P., Goonetilleke, R. S. // China – 2004 – С. 764-769
6. Hilton, A., Beresford, D., Gentils, T., Smith, R., Sun, W., Illingworth, J. Wholebody modelling of people from multiview images to populate virtual worlds. / Hilton, A., Beresford, D., Gentils, T., Smith, R., Sun, W., Illingworth, J. // California – 2000 – С. 411-436
7. Tomáš Kohlschütter (2012), Human Body Modelling by Development of the Automatic Landmarking Algorithm / Tomáš Kohlschütter // West Bohemia – 2012 С. 24-41

8. Python [Електронний ресурс] - Режим доступу:  
<https://www.python.org/>
9. Ruby programming language [Електронний ресурс] - Режим доступу:  
<https://www.ruby-lang.org/>
10. Програмування на C++. Семантичний конспект розділу [Електронний ресурс] - Режим доступу:  
<http://www.znannya.org/?view=summary:Cpp>
11. Java [Електронний ресурс] - Режим доступу: <http://java.com>
12. Ruby Sinatra [Електронний ресурс] - Режим доступу:  
<http://www.sinatrarb.com/>
13. Записки про Ruby [Електронний ресурс] - Режим доступу:  
<http://rubyonrails.org/>
14. Django Project [Електронний ресурс] - Режим доступу:  
<https://www.djangoproject.com/>
15. Pylons Project [Електронний ресурс] - Режим доступу:  
<http://www.pylonsproject.org/>
16. OpenCV бібліотека комп'ютерного зору з відкритим вихідним кодом [Електронний ресурс] - Режим доступу:  
<http://engineer.org.ua/page/opencv-biblioteka-kompjuternogo-zoru-z-vidkritim-vihidnim-kodom>
17. Курс - Фреймворк Bootstrap [Електронний ресурс] - Режим доступу:  
<http://edu.cbsystematics.com/ua/courses/description/bootstrap>
18. RGB to HSV Color Conversion Algorithm [Електронний ресурс] - Режим доступу: <http://math.stackexchange.com/questions/556341/rgb-to-hsv-color-conversion-algorithm>

19. Suzuki, S., Abe, K. Topological Structural Analysis of Digitized Binary Images by Border Following. / Suzuki, S., Abe, K. // Tokyo – 1985 – С. 32-46
20. Urs Ramer "An iterative procedure for the polygonal approximation of plane curves", Computer Graphics and Image Processing / Urs Ramer // New York – 1972 – С. 244-256
21. John Hershberger, Jack Snoeyink "Speeding Up the Douglas-Peucker Line-Simplification Algorithm" / John Hershberger, Jack Snoeyink // Department of Computer Science University of British Columbia - 2000 – С. 1-10
22. Як вибрати правильний розмір футболки? [Електронний ресурс] - Режим доступу: <http://www.vddruk.lviv.ua/tsikavo-znaty/yak-vybraty-pravylnyy-rozmir-futbolky/>

