Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

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

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти перший (бакалаврський)

Розробка інтелектуальної системи віртуальної примірки одягу на основі аналізу пози людини та геометричних перетворень з використанням генеративних моделей
(тема)

Виконав:
здобувач четвертого року навчання,
групи ІТШІ-22-3

Микита ФЕСЕНКО
(власне ім'я, прізвище)

Спеціальність 122 Комп'ютерні науки

(код і повна назва спеціальності)
Тип програми освітньо-професійна
Освітня програма Штучний інтелект

(повна назва освітньої програми)

Керівник ас. Максим ПОЛІЇТ
(посада, власне ім'я, прізвище)

Допускається до захисту

Завідувач кафедри ШІ _____ Лариса ЧАЛА
(підпис) (власне ім'я, прізвище)

2026 р.

Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук

Кафедра Штучного інтелекту

Рівень вищої освіти перший (бакалаврський)

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

Тип програми освітньо-професійна

Освітня програма Штучний інтелект
(повна назва)

ЗАТВЕРДЖУЮ:
Зав. кафедри _____
(підпис)
«_____» _____ 20 ___ р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві Фесенку Микиті Богдановичу
(прізвище, ім'я, по батькові)

1. Тема роботи Розробка інтелектуальної системи віртуальної примірки одягу на основі аналізу пози людини та геометричних перетворень з використанням генеративних моделей

затверджена наказом університету від 5 червня 20 26 р. № 488Ст
2. Термін подання здобувачем роботи до екзаменаційної комісії 10 липня 20 26 р.
3. Вихідні дані до роботи науково-технічні джерела з віртуальної примірки одягу; матеріали про методи оцінювання пози людини, сегментації зображень, геометричних перетворень і генеративних моделей; програмний проєкт ai_virtual_tryon; знімки інтерфейсу локального застосунку; середовище Python, FastAPI, Next.js, Gradio API та IDM-VTON.

4. Перелік питань, що потрібно опрацювати в роботі
1) Аналіз предметної галузі віртуальної примірки одягу
2) Математичне та алгоритмічне забезпечення системи
3) Проєктування та реалізація програмної системи
4) Експериментальні дослідження та аналіз результатів

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Строк / термін виконання етапів роботи	Примітка
1	Отримання завдання на кваліфікаційну роботу	08.06.2026	виконано
2	Аналіз предметної області та джерел	09.06.2026	виконано
3	Уточнення постановки задачі та вимог	18.06.2026	виконано
4	Розробка математичного забезпечення	23.06.2026	виконано
5	Проєктування архітектури системи	27.06.2026	виконано
6	Реалізація backend, frontend та AI-модулів	04.07.2026	виконано
7	Експериментальна перевірка системи	05.07.2026	виконано
8	Оформлення пояснювальної записки	06.07.2026	виконано
9	Підготовка презентаційних матеріалів	07.07.2026	виконано
10	Захист перед ЕК	10.07.2026	

Дата видачі завдання 8 червня 2026 р.

Здобувач _____
(підпис)

Керівник роботи _____ ас. Максим ПОЛІЇТ
(підпис) (посада, власне ім’я, прізвище)

РЕФЕРАТ

Пояснювальна записка: 65 с., 11 рис., 11 табл., 1 дод., 29 джерел.

АНАЛІЗ ПОЗИ ЛЮДИНИ, ВІРТУАЛЬНА ПРИМІРКА, ГЕНЕРАТИВНІ МОДЕЛІ, ГЕОМЕТРИЧНІ ПЕРЕТВОРЕННЯ, ГЛИБОКЕ НАВЧАННЯ, ЕЛЕКТРОННА КОМЕРЦІЯ, КОМП'ЮТЕРНИЙ ЗІР, ОДЯГ, СЕГМЕНТАЦІЯ ЗОБРАЖЕНЬ, ШТУЧНИЙ ІНТЕЛЕКТ.

Об'єкт дослідження – процес віртуальної примірки одягу на цифрових зображеннях користувача.

Предмет дослідження – методи аналізу пози людини, сегментації зображень, геометричного узгодження предметів одягу з тілом користувача та генеративного покращення результатів віртуальної примірки.

Мета роботи – розроблення інтелектуальної системи віртуальної примірки одягу, що поєднує модулі аналізу пози людини, геометричних перетворень, асинхронної серверної обробки та генеративної моделі для формування реалістичного результату примірки.

Методи дослідження – аналіз науково-технічної літератури, методи комп'ютерного зору, оцінювання ключових точок тіла людини, сегментації зображень, афінних та нелінійних геометричних перетворень, дифузійного генеративного моделювання, проєктування програмних систем і функціонального тестування.

У роботі проаналізовано предметну область віртуальної примірки, розроблено постановку задачі, запропоновано математичне та алгоритмічне забезпечення системи, спроєктовано архітектуру web-застосунку, описано реалізацію backend-частини на FastAPI, frontend-частини на Next.js та модуля генерації результату через IDM-VTON. Практична значимість полягає у можливості інтеграції системи в електронні магазини одягу, маркетплейси та сервіси персоналізованого підбору стилю.

ABSTRACT

Bachelor’s thesis contains: 65 pp., 11 fig., 11 tabl., 1 ann., 29 references.

ARTIFICIAL INTELLIGENCE, CLOTHING, COMPUTER VISION, DEEP LEARNING, E-COMMERCE, GENERATIVE MODELS, GEOMETRIC TRANSFORMATIONS, HUMAN POSE ANALYSIS, IMAGE SEGMENTATION, VIRTUAL TRY-ON.

The object of research is the process of virtual clothing try-on using digital images of a user. The subject of research is a set of methods for human pose estimation, image segmentation, geometric garment alignment and generative refinement of the final virtual try-on result.

The purpose of the thesis is to develop an intelligent virtual clothing try-on system that combines pose analysis, geometric transformations, asynchronous server-side processing and generative image synthesis to produce a realistic visualization of a selected garment on a user photo.

The research methods include analysis of scientific and technical sources, computer vision techniques, keypoint-based human pose estimation, image segmentation, affine and non-linear transformations, diffusion-based generative modeling, software architecture design and functional testing.

The thesis presents an analytical review of virtual try-on methods, formalizes the task, proposes mathematical and algorithmic support, designs a web-based architecture, and describes implementation of the FastAPI backend, Next.js frontend and IDM-VTON based AI processing module. The practical value of the work is the possibility of integrating the developed system into online fashion stores, marketplaces and personalized styling applications.

ЗМІСТ

Перелік умовних позначень, символів, одиниць, скорочень і термінів 7
Вступ 8
1 Аналіз предметної галузі віртуальної примірки одягу 10
1.1 Загальна характеристика задачі віртуальної примірки 10
1.2 Аналіз пози людини та сегментація зображень 13
1.3 Геометричні перетворення в задачі узгодження одягу 17
1.4 Використання генеративних моделей у задачі примірки 21
1.5 Постановка задачі розробки системи 24
1.6 Практичні умови використання віртуальної примірки в онлайн-магазині 27
2 Математичне та алгоритмічне забезпечення системи 30
2.1 Формальна постановка задачі 30
2.2 Модель представлення зображень і ключових точок 32
2.3 Афінні перетворення, гомографія та tps-викривлення 34
2.4 Обробка перекриттів, композиція та оцінювання якості 35
2.5 Алгоритм підготовки даних перед запуском генеративної моделі ... 37
3 Проєктування та реалізація програмної системи 39
3.1 Вимоги до системи та сценарії використання 39
3.2 Архітектура застосунку та структура проєкту 40
3.3 Реалізація backend, api та асинхронної обробки 42
3.4 Реалізація ai-модуля та інтеграція з idm-vton 44
3.5 Реалізація frontend та користувацького інтерфейсу 46
3.6 Модель даних, збереження файлів та розгортання 49
3.7 Обробка помилок, захист файлів і журналювання запитів 51

4 Експериментальні дослідження та аналіз результатів 53
4.1 Методика тестування 53
4.2 Перевірка інтерфейсу та основних функцій 54
4.3 Аналіз якості, обмежень і напрямів покращення 55
4.4 Практична придатність прототипу для інтеграції в електронну комерцію 57
Висновки 59
Перелік джерел посилання 61
Додаток А 64

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

ШІ – штучний інтелект;

AI – Artificial Intelligence – штучний інтелект;

API – Application Programming Interface – програмний інтерфейс застосунку;

CNN – Convolutional Neural Network – згорткова нейронна мережа;

CV – Computer Vision – комп’ютерний зір;

FastAPI – web-фреймворк для створення API мовою Python;

GAN – Generative Adversarial Network – генеративно-змагальна мережа;

IDM-VTON – Improving Diffusion Models for Authentic Virtual Try-on;

JSON – JavaScript Object Notation – формат обміну даними;

REST – Representational State Transfer – архітектурний стиль API;

SAM – Segment Anything Model – модель сегментації зображень;

TPS – Thin-Plate Spline – тонкопластинне сплайнове перетворення;

UI – User Interface – користувацький інтерфейс;

VTON – Virtual Try-On – віртуальна примірка.

ВСТУП

Сучасний розвиток електронної комерції, мобільних застосунків і технологій штучного інтелекту зумовив появу нових сервісів у сфері онлайн-продажу одягу. Однією з найбільш актуальних проблем цієї сфери є те, що користувач не має можливості фізично приміряти товар перед купівлею. Через це покупець приймає рішення на основі фотографій моделі або товару, які не завжди відповідають його власним пропорціям, кольоротипу, стилю та очікуванням.

Віртуальна примірка одягу є перспективним напрямом комп'ютерного зору, який поєднує аналіз зображень, оцінювання пози людини, сегментацію, геометричні перетворення та генеративне моделювання. На відміну від простого накладання зображення одягу, інтелектуальна система повинна враховувати положення тіла, форму силуету, перекриття рук і волосся, освітлення, масштаб і текстуру тканини. Саме така комплексність визначає наукову та практичну актуальність теми.

Актуальність роботи полягає у необхідності розроблення програмної системи, здатної автоматизувати процес віртуальної примірки та підвищити якість користувацького досвіду в онлайн-магазині. Очікується, що використання такого сервісу може зменшити невизначеність під час вибору товару, підвищити довіру користувача до електронного каталогу та створити основу для персоналізованого підбору стилю.

Об'єктом дослідження є процес віртуальної примірки одягу на цифрових зображеннях користувача. Предметом дослідження є методи аналізу пози людини, геометричного узгодження предметів одягу з тілом користувача, генеративного покращення результату та програмні засоби реалізації відповідної web-системи.

Метою кваліфікаційної роботи є розроблення інтелектуальної системи віртуальної примірки одягу на основі аналізу пози людини, геометричних перетворень та генеративних моделей, а також перевірка

працездатності цієї системи на прикладі web-застосунку з каталогом товарів, сторінкою товару, кошиком і модулем AI-примірки.

Для досягнення мети необхідно вирішити такі задачі: виконати аналіз предметної галузі; дослідити сучасні підходи до віртуальної примірки; формалізувати задачу обробки зображень; розробити математичне та алгоритмічне забезпечення; спроєктувати архітектуру програмної системи; реалізувати backend, frontend і AI-модуль; провести функціональне тестування; проаналізувати отримані результати та сформулювати напрями подальшого розвитку.

Методами дослідження є аналіз наукових джерел, порівняльний аналіз існуючих підходів, методи комп'ютерного зору, оцінювання пози людини, сегментації, афінних і нелінійних перетворень, дифузійного генеративного моделювання, web-програмування, асинхронної серверної обробки та тестування програмного забезпечення.

Практичне значення роботи полягає у створенні системи, яку можна використовувати як прототип інтелектуального модуля для інтернет-магазинів одягу, fashion-маркетплейсів і мобільних застосунків. Система дозволяє користувачеві переглядати каталог, обирати товар, запускати процес примірки та отримувати згенероване зображення, що демонструє обраний предмет одягу на фотографії користувача.

Пояснювальна записка складається зі вступу, чотирьох розділів, висновків, переліку джерел посилання та додатків. У першому розділі проаналізовано предметну галузь. У другому розділі наведено математичне та алгоритмічне забезпечення. У третьому розділі описано проєктування та реалізацію програмної системи. У четвертому розділі наведено експериментальну перевірку, аналіз результатів, обмеження та рекомендації щодо розвитку системи.

1 АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ ВІРТУАЛЬНОЇ ПРИМІРКИ ОДЯГУ

1.1 Загальна характеристика задачі віртуальної примірки

Віртуальна примірка одягу є напрямом комп’ютерного зору та штучного інтелекту, який спрямований на створення зображення людини в обраному предметі одягу без фізичної примірки товару. Основна ідея полягає в тому, що користувач завантажує власне фото, обирає предмет одягу з каталогу, після чого система автоматично аналізує зображення людини, визначає положення тіла, виділяє потрібні області та формує результат примірки.

На відміну від звичайного накладання зображення одягу на фотографію, інтелектуальна віртуальна примірка повинна враховувати позу людини, пропорції тіла, нахил корпусу, положення рук, форму силуету, видимі й перекриті частини тіла, а також особливості самого предмета одягу. Саме тому така задача є складною та потребує використання декількох груп методів, зокрема аналізу пози, сегментації, геометричного перетворення та генеративного моделювання.

Процес віртуальної примірки зазвичай починається з отримання вхідного зображення користувача та зображення предмета одягу. Після цього система визначає ключові точки тіла людини, аналізує положення основних частин тіла, виконує сегментацію потрібних областей і готує предмет одягу до подальшого перетворення. На наступному етапі одяг масштабується, повертається, деформується та узгоджується з тілом користувача. Завершальним етапом є формування фінального зображення та його генеративне покращення для усунення візуальних артефактів.

У межах цієї задачі вхідними даними виступають фотографія користувача та зображення товару, а проміжними результатами — карта ключових точок, маска тіла, трансформований контур одягу й сформоване

зображення. Така послідовність дозволяє контролювати, на якому етапі виникають відхилення, наприклад зміщення предмета, неточне виділення меж або втрата деталей тканини [1], [4].

У реальному сценарії вхідне фото може бути зроблене з різної відстані, у вертикальній або горизонтальній орієнтації, з неоднорідним фоном чи частковими перекриттями. Тому перед перенесенням одягу потрібно оцінити придатність зображення: чи видно тіло повністю, чи достатня роздільна здатність і чи не заважають сторонні об’єкти ключовим областям плечей, талії та рук. Така попередня перевірка зменшує ймовірність того, що подальші етапи працюватимуть з некоректними даними.

Для практичної реалізації було обрано модульний підхід. Він дозволяє ізолювати логіку завантаження файлів, перевірки формату, аналізу пози, сегментації, геометричного перетворення, генеративного покращення та оцінювання якості. Така декомпозиція спрощує тестування, підвищує надійність і створює можливість заміни окремого алгоритму без повного перепроєктування системи.

З погляду користувацького досвіду процесу перенесення одягу на фотографію користувача має бути прихованим за простим сценарієм взаємодії: користувач відкриває каталог, обирає товар, завантажує фотографію та очікує результат. Уся складність обчислень переноситься на серверну частину, де виконується асинхронна обробка запиту, збереження проміжних артефактів і повернення фінального зображення через API.

У роботі зроблено акцент на балансі між математично обґрунтованими геометричними операціями та генеративним моделюванням. Геометричний етап забезпечує контрольованість і пояснюваність процесу, а генеративний етап дає змогу підвищити фотореалістичність зображення, усунути локальні артефакти та узгодити текстуру одягу з контекстом фотографії.

Практична цінність розглянутого підходу полягає в тому, що його можна інтегрувати в електронний магазин або маркетплейс без радикальної зміни бізнес-логіки. Каталог товарів, сторінка товару, кошик і замовлення залишаються звичними компонентами, а віртуальна примірка додається як інтелектуальний сервіс, що покращує прийняття рішення користувачем.

Ключовими поняттями для цього підрозділу є цифрове зображення, силует, цільовий предмет одягу, фотореалістичне відтворення. Вони визначають спосіб перетворення початкових даних у проміжні представлення, придатні для подальшої генерації. Якщо на етапі побудови цих представлень не врахувати пропорції тіла, межі одягу та перекриття рук або волосся, модель може сформувати візуально привабливе, але технічно неправильне зображення.

Окрему увагу приділено можливості масштабування. Навіть якщо система використовується локально під час розробки, її архітектура повинна підтримувати подальший перехід до серверного розгортання. Для цього запити примірки виконуються асинхронно, а результат зберігається у файловому сховищі з прив’язкою до унікального ідентифікатора.

Для забезпечення повторюваності результатів у роботі передбачено фіксацію основних параметрів обробки: режим генерації, кількість спроб, рівень реалістичності, необхідність збереження обличчя та фону. Наявність таких параметрів важлива для експериментального порівняння, оскільки дозволяє аналізувати якість не лише за суб’єктивним враженням, а й за умовами отримання результату. Порівняння основних підходів до віртуальної примірки наведено в таблиці 1.1.

Таблиця 1.1 – Порівняння основних підходів до віртуальної примірки

Підхід	Переваги	Обмеження	Доцільність використання
2D-накладання	Простота реалізації, низькі вимоги до ресурсів	Низька реалістичність, відсутність адаптації до пози	Демонстраційні прототипи

Продовження таблиці 1.1

Підхід	Переваги	Обмеження	Доцільність використання
Геометричне викривлення	Контрольованість, збереження текстури	Потреба в якісних ключових точках і масках	Системи з пояснюваною логікою
GAN-підходи	Фотореалістичність і гнучкість	Можлива втрата деталей одягу	Дослідницькі VTON-системи
Дифузійні моделі	Висока якість генерації, робота з in-the-wild фото	Висока обчислювальна складність	Сучасні продукційні системи
Гібридний підхід	Поєднання геометричного контролю та генеративної якості	Складніша архітектура	Обрана стратегія роботи

1.2 Аналіз пози людини та сегментація зображень

Таким чином, задача віртуальної примірки є комплексною, оскільки вона поєднує аналіз зображень, просторове узгодження об’єктів, обробку перекриттів і генерацію фотореалістичного результату. Її складність полягає в тому, що система повинна не лише механічно накласти зображення одягу на фотографію користувача, а й врахувати особливості будови тіла, положення людини у кадрі, масштаб зображення, перспективу, освітлення та взаємне розташування окремих елементів. У реальних умовах користувачі можуть завантажувати фотографії різної якості, зроблені під різними кутами, з неоднорідним фоном або частковим перекриттям тіла руками, волоссям чи іншими об’єктами. Через це задача віртуальної примірки потребує використання кількох взаємопов’язаних етапів обробки, кожен з яких впливає на кінцеву якість результату.

Аналіз пози людини як основа віртуальної примірки.

Одним із ключових етапів віртуальної примірки є аналіз пози людини. Поза визначає просторове розташування основних частин тіла: голови, плечей, рук, тулуба, стегон і ніг. Для системи примірки особливо важливими є ключові точки плечей, ліктів, зап’ясть, шиї, грудної області, талії та стегон, оскільки саме вони дозволяють правильно розмістити предмет одягу на тілі

користувача. Наприклад, положення плечей і шиї використовується для коректного розміщення верхньої частини одягу, положення талії та стегон — для визначення меж нижньої частини виробу, а координати рук допомагають зрозуміти, які ділянки одягу можуть бути частково перекриті. Якщо ці точки визначено неточно, система може неправильно масштабувати або змістити одяг, що призведе до неприродного вигляду результату.

Аналіз пози людини дає змогу визначити, у якому положенні знаходиться тіло користувача, а також обчислити приблизні геометричні параметри фігури. До таких параметрів можна віднести ширину плечей, нахил корпусу, положення рук і співвідношення основних частин тіла. Ці дані надалі використовуються для геометричного узгодження предмета одягу з тілом людини. Крім того, на основі виявленої пози можна оцінити, чи підходить зображення для якісної примірки, оскільки надто сильний поворот корпусу, закриті руки, нахил тіла або неповне потрапляння людини в кадр можуть ускладнювати подальшу обробку. У такому випадку система може або виконати додаткову корекцію, або повідомити користувача про необхідність завантажити більш придатне фото.

У системі віртуальної примірки аналіз пози може виконуватися за допомогою моделей оцінювання ключових точок тіла людини. Такі моделі отримують на вхід зображення та повертають набір координат, які відповідають анатомічно важливим точкам. На основі цих координат система може побудувати умовну скелетну модель користувача та використати її для вирівнювання одягу. Скелетна модель не є повною тривимірною копією тіла, однак вона дозволяє достатньо точно визначити основні пропорції та напрямки розташування частин тіла на двовимірному зображенні. Саме тому оцінювання пози є важливим проміжним етапом між початковим аналізом фотографії та подальшою генерацією нового зображення з підібраним одягом.

Для модуля оцінювання пози критичним є не лише саме виявлення ключових точок, а й перевірка їхньої узгодженості між собою. Якщо координати плечей, шиї або зап’ясть визначені неточно, наступні етапи отримують помилкові орієнтири для масштабування і побудови маски. Тому результати pose-estimation використовуються як контрольна основа для сегментації та геометричного вирівнювання [5], [8].

Наприклад, неправильне визначення лінії плечей може спричинити некоректне розміщення коміра або рукавів, а помилка у визначенні області талії може вплинути на посадку сукні, сорочки чи верхнього одягу. Тому модуль аналізу пози повинен працювати узгоджено з іншими компонентами системи, зокрема з модулями сегментації, побудови маски, масштабування та генерації фінального результату.

Для оцінювання ключових точок важливо враховувати не лише координати окремих суглобів, а й загальну узгодженість скелетної моделі. Якщо рука перекриває тулуб, людина стоїть під кутом або частина тіла виходить за межі кадру, деякі точки можуть визначатися з нижчою достовірністю. У таких випадках доцільно поєднувати результати pose-estimation із маскою силуету та додатковою перевіркою симетрії плечей, положення шиї і меж тулуба. Це дозволяє відрізнити придатне фото від зображення, на якому подальша примірка буде нестабільною. Окремо слід враховувати положення рук, оскільки саме вони часто перекривають область нового одягу і впливають на порядок накладання шарів.

Ключові точки, що використовуються для віртуальної примірки, наведено в таблиці 1.2.

Таблиця 1.2 – Ключові точки для віртуальної примірки

Ключова точка	Призначення	Використання в алгоритмі
Ліве та праве плече	Визначення ширини верхньої частини тіла	Масштабування і поворот верхнього одягу

Продовження таблиці 1.2

Ключова точка	Призначення	Використання в алгоритмі
Шия	Орієнтир для верхньої межі виробу	Формування точки прив’язки футболки або сорочки
Лікті та зап’ястя	Виявлення положення рук	Обробка перекриттів і збереження рук поверх одягу
Ліве та праве стегно	Оцінювання положення нижньої частини тулуба	Узгодження довжини виробу
Контур силуету	Виділення меж тіла	Корекція маски та композиції

У контексті теми «аналіз пози та сегментація» важливим є збереження відповідності між візуальною інформацією та структурними даними системи. Зображення користувача, зображення товару, ідентифікатор завдання, статус обробки та шлях до результату мають розглядатися не як окремі файли, а як взаємопов’язані сутності одного процесу. Це дозволяє відстежувати життєвий цикл примірки від моменту завантаження фото до формування URL готового зображення (рисунок 1.1). Такий підхід є важливим для забезпечення цілісності даних, оскільки кожен етап обробки повинен бути пов’язаний з конкретним запитом користувача та відповідним результатом. Якщо система зберігає лише окремі зображення без зв’язку між ними, ускладнюється контроль процесу, пошук помилок і повторне використання отриманих результатів. Натомість використання єдиної структури завдання дає змогу точно визначити, яке фото користувача було використане, який товар застосовувався для примірки, на якому етапі перебуває обробка та де збережено фінальний результат. У межах такого процесу аналіз пози та сегментація виконують роль проміжних, але критично важливих етапів, оскільки саме вони формують основу для подальшого коректного накладання або генерації одягу. Статус обробки дозволяє frontend-частині системи інформувати користувача про поточний стан операції, наприклад про створення завдання, перевірку вхідних даних, виконання AI-обробки, успішне завершення або виникнення помилки.

Шлях до результату або URL готового зображення, у свою чергу, є завершальним елементом цього ланцюга, оскільки він підтверджує, що всі попередні етапи були виконані та користувач може переглянути отриманий результат. Таким чином, взаємозв’язок між вхідними зображеннями, службовими параметрами, статусами та результатом забезпечує керованість процесу віртуальної примірки, підвищує надійність роботи системи та спрощує подальший аналіз помилок або повторну обробку запиту.



Рисунок 1.1 – Потік даних під час аналізу зображень

1.3 Геометричні перетворення в задачі узгодження одягу

Основною проблемою аналізу пози є те, що фотографії користувачів можуть мати різну якість, освітлення, масштаб, фон і положення тіла. Крім того, частини тіла можуть бути перекриті волоссям, руками, аксесуарами або іншими предметами. Тому система повинна бути стійкою до неточностей і мати механізми перевірки якості вхідних даних. Така перевірка може включати оцінювання наявності людини у кадрі, повноти видимості тіла, достатньої роздільної здатності зображення, правильності орієнтації фотографії та відсутності значних перекриттів у ключових зонах.

Якщо вхідне зображення має низьку якість або не дозволяє коректно визначити ключові точки, система повинна зафіксувати цю ситуацію як потенційну помилку обробки або запропонувати користувачеві завантажити інше фото. Це дає змогу зменшити кількість некоректних результатів і підвищити загальну надійність віртуальної примірки.

Після визначення пози людини необхідно узгодити зображення предмета одягу з тілом користувача. Для цього застосовуються геометричні перетворення, призначення яких полягає у зміні масштабу, положення, нахилу та форми одягу відповідно до пози й пропорцій людини на фотографії. На цьому етапі система використовує отримані раніше ключові точки тіла, контури силуету та результати сегментації, щоб визначити, де саме має розміщуватися предмет одягу. Наприклад, для верхнього одягу важливими є координати плечей, шиї, грудної області та талії, а для нижнього одягу — положення талії, стегон і ніг. Завдяки цьому зображення товару може бути не просто накладене поверх фотографії, а просторово узгоджене з тілом користувача.

Найпростішими є афінні перетворення, які дозволяють виконати масштабування, поворот, зсув і нахил зображення. Вони можуть бути використані для базового вирівнювання предмета одягу, наприклад футболки, сорочки або сукні, за координатами плечей і тулуба користувача. Однак афінні перетворення не завжди достатні, оскільки реальний одяг має складну форму, складки та локальні деформації. Крім того, тіло людини на фотографії може бути повернуте під кутом, нахилене або частково перекрите, через що просте масштабування чи поворот не забезпечує природного вигляду. Афінне перетворення є доцільним для початкового вирівнювання, але воно не враховує складної поведінки тканини, зміну форми рукавів, прилягання до тіла та особливості посадки одягу на різних типах фігури.

Для більш точного узгодження можуть застосовуватися нелінійні перетворення. Вони дозволяють деформувати зображення одягу за

контрольними точками, що краще відповідає реальній зміні форми тканини на тілі людини. Контрольні точки можуть визначатися на основі ключових точок тіла, контурів одягу та сегментованих областей. Такий підхід дає змогу змінювати форму окремих частин предмета одягу незалежно одна від одної, наприклад коригувати положення рукавів, лінію плечей, нижній край виробу або область талії. Нелінійні перетворення є особливо важливими у випадках, коли поза користувача відрізняється від фронтального положення або коли одяг повинен повторювати контури тіла. Вони дозволяють отримати більш реалістичний результат, оскільки враховують не лише загальний розмір і нахил об’єкта, а й локальні зміни його форми.

Геометричний етап пов’язує абстрактні ознаки пози з реальним розташуванням предмета одягу на зображенні. Його результатом є набір перетворень, які визначають масштаб, кут повороту, локальне викривлення та область накладання виробу. Якість цих параметрів безпосередньо впливає на природність рукавів, коміра, нижнього краю та контурів тканини [2], [11], [12].

Наприклад, якщо координати плечей визначено неточно, подальше масштабування може призвести до неправильного розташування рукавів або коміра. Якщо некоректно виконано сегментацію тіла, то частина одягу може перекривати відкриті ділянки шкіри або, навпаки, зникати в областях, де вона повинна бути видимою. Тому геометричне узгодження має виконуватися у взаємодії з модулями аналізу пози, сегментації, побудови маски та генерації фінального зображення.

Під час масштабування та деформації одягу система повинна орієнтуватися на конкретні геометричні ознаки зображення: відстань між плечима, нахил лінії корпусу, положення талії та видиму довжину виробу. Для фронтальних фото достатньо базового вирівнювання, тоді як для нахилених або частково повернутих фігур потрібне локальне уточнення форми за контрольними точками. Важливо, щоб перетворення не руйнувало характерні деталі товару: форму коміра, довжину рукава, принт, фактуру

або нижній край виробу. Через це геометричне узгодження має поєднувати загальне позиціонування з обмеженнями на допустиме викривлення, інакше реалістичність може бути досягнута ціною втрати відповідності реальному одягу.

Ключовими поняттями для цього підрозділу є афінне перетворення, гомографія, TPS, контрольні точки, деформація тканини. Принцип нелінійного TPS-викривлення за контрольними точками наведено на рисунку 1.2. Афінне перетворення використовується для базового вирівнювання об’єкта, гомографія дає змогу враховувати перспективні зміни площини, а TPS-викривлення забезпечує плавну нелінійну деформацію зображення відповідно до заданих контрольних точок. У задачі віртуальної примірки ці поняття застосовуються разом, оскільки спочатку необхідно приблизно розмістити одяг на тілі користувача, а потім уточнити його форму з урахуванням контурів фігури, пози та видимих областей. Саме поєднання базових і більш складних перетворень дозволяє отримати результат, який виглядає природніше та краще відповідає реальній посадці одягу на людині.

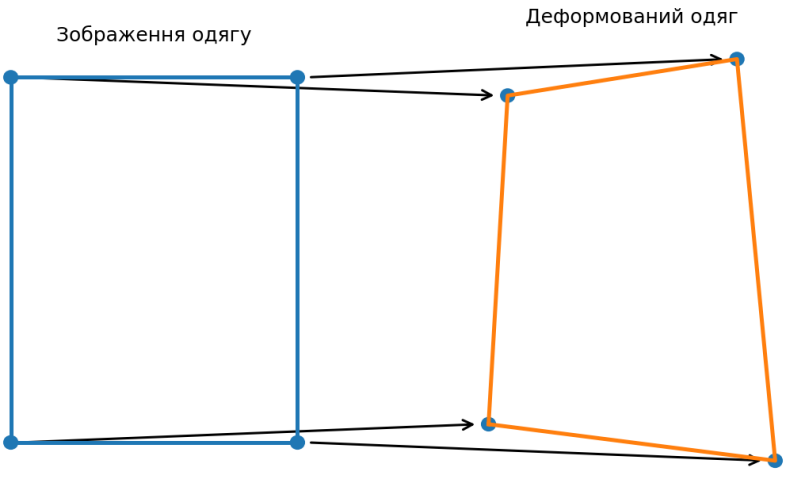


Рисунок 1.2 – Принцип нелінійного TPS-викривлення за контрольними точками

1.4 Використання генеративних моделей у задачі примірки

Геометричне перетворення є важливим етапом, оскільки саме від нього залежить природність розташування предмета одягу на тілі користувача. Якщо одяг неправильно масштабовано, зміщено або деформовано, фінальний результат виглядатиме нереалістично навіть за умови подальшого використання генеративної моделі. Це пояснюється тим, що генеративна модель не завжди може повністю виправити грубі помилки попередніх етапів, особливо якщо предмет одягу розташований не відповідно до анатомічних особливостей тіла або не збігається з ключовими точками користувача. Тому геометричне узгодження має виконуватися максимально точно ще до етапу фінальної генерації, оскільки воно формує основу для подальшого покращення зображення.

Генеративні моделі застосовуються для покращення реалістичності фінального зображення. Після геометричного накладання одягу можуть виникати різні артефакти, зокрема неприродні межі, розриви текстури, некоректні тіні, спотворення тканини або проблеми в областях перекриття руками та волоссям. Генеративна модель може використовуватися для усунення таких дефектів і створення більш цілісного зображення. Вона здатна доповнювати відсутні фрагменти, згладжувати переходи між тілом користувача та новим предметом одягу, а також узгоджувати освітлення між вихідною фотографією і накладеним об’єктом. Завдяки цьому фінальне зображення може виглядати не як простий монтаж, а як більш природна фотографія, у якій предмет одягу візуально поєднаний із тілом користувача та навколишнім середовищем.

У контексті цієї роботи генеративна модель розглядається не як єдиний інструмент розв’язання задачі, а як модуль фінального покращення результату. Основна логіка системи має базуватися на аналізі пози людини, сегментації та геометричних перетвореннях. Такий підхід дозволяє зробити

систему більш керованою, пояснюваною та придатною для подальшого дослідження. Якщо весь процес побудовано лише на генеративній моделі, складніше визначити, на якому саме етапі виникла помилка та чому результат виявився некоректним. Натомість поділ системи на окремі модулі дає змогу аналізувати роботу кожного етапу окремо, перевіряти якість вхідних і проміжних даних, а також удосконалювати систему поступово без повної зміни її архітектури.

Генеративний модуль може покращувати межі між тілом і предметом одягу, відновлювати частково перекриті області, коригувати тіні та освітлення, узгоджувати текстуру одягу з формою тіла й усувати дрібні візуальні артефакти. Водночас використання генеративних моделей має певні обмеження. Такі моделі можуть змінювати деталі зовнішності користувача або самого предмета одягу, що є небажаним для задачі електронної комерції. Тому важливо передбачити контроль якості результату та обмежити генеративну модель так, щоб вона не змінювала суттєві характеристики товару. До таких характеристик можна віднести колір, фасон, довжину, форму рукавів, наявність принтів, декоративних елементів, застібок або інших деталей, які мають значення для покупця. У системах електронної комерції результат примірки повинен не лише виглядати реалістично, а й відповідати реальному товару, оскільки спотворення зовнішнього вигляду виробу може призвести до неправильного сприйняття товару користувачем.

Генеративний компонент доцільно запускати після попередньої підготовки масок і геометричного вирівнювання. У такому разі модель працює не з невпорядкованим монтажем, а з уже сформованою областю редагування, що зменшує ризик зміни обличчя, фону або важливих характеристик товару. Це особливо важливо для електронної комерції, де збереження реального вигляду виробу має практичне значення [4], [9], [13], [14].

Наприклад, якщо на етапі сегментації неправильно визначено межі рук або тулуба, генеративна модель може некоректно відновити область перекриття або змінити частину одягу. Якщо ж геометричне перетворення виконано з помилкою, модель може намагатися приховати дефект, але результат усе одно залишатиметься візуально неприродним. Саме тому генеративний модуль повинен працювати не ізольовано, а у зв'язку з попередніми етапами аналізу пози, сегментації, побудови маски та геометричного узгодження.

Для дифузійного покращення важливо обмежити область редагування, щоб модель не змінювала зайві частини фотографії. Маска повинна охоплювати переважно зону нового одягу та прилеглі межі, а фон, обличчя й відкриті частини тіла мають залишатися стабільними. Це зменшує ризик появи сторонніх деталей і допомагає зберегти автентичність товару. Також важливо передавати моделі достатньо інформації про тип виробу, його колір і бажаний рівень реалістичності. Якщо область редагування надто широка, модель може домалювати зайві елементи; якщо надто вузька — на межах одягу залишаються помітні шви або різниця в освітленні.

Порівняння відомих підходів VTON наведено в таблиці 1.3. Таке порівняння дає змогу оцінити переваги та обмеження різних методів віртуальної примірки, зокрема підходів, що базуються на сегментації, геометричних перетвореннях, генеративних мережах і дифузійних моделях. Аналіз цих підходів є важливим для вибору архітектури системи, оскільки різні методи можуть по-різному працювати з фотографіями користувачів, типами одягу, складними позами та випадками часткового перекриття тіла. Саме тому доцільно розглядати VTON не як одну окрему технологію, а як сукупність методів, які можуть комбінуватися залежно від вимог до якості, швидкодії, керованості та точності збереження характеристик товару.

Таблиця 1.3 – Порівняння відомих підходів VTON

Метод	Основна ідея	Сильна сторона	Обмеження
VITON	coarse-to-fine перенесення одягу	рання формалізація задачі	обмежена роздільна здатність
CP-VTON	TPS-перетворення та маска композиції	краще збереження ознак одягу	чутливість до помилок вирівнювання
VITON-HD	ALIAS normalization для високої роздільності	якість 1024×768	складність підготовки даних
IDM-VTON	дифузійна модель з семантикою одягу	вища автентичність і збереження деталей	потреба в GPU або зовнішньому сервісі

1.5 Постановка задачі розробки системи

На основі аналізу предметної галузі було сформульовано задачу розробки інтелектуальної системи віртуальної примірки одягу. Система повинна отримувати на вхід фотографію користувача та зображення предмета одягу, після чого виконувати аналіз пози людини, сегментацію необхідних областей, геометричне узгодження одягу з тілом і генеративне покращення результату. Такий підхід дозволяє розглядати віртуальну примірку не як просте накладання одного зображення на інше, а як послідовний процес інтелектуальної обробки візуальних даних. Кожний етап у цьому процесі має власне призначення: аналіз пози визначає просторове розташування тіла, сегментація відокремлює важливі області зображення, геометричне перетворення забезпечує правильне розміщення одягу, а генеративний модуль покращує природність фінального результату. У сукупності ці етапи формують основу програмної системи, здатної працювати з різними фотографіями користувачів і різними видами товарів.

Основними вимогами до системи є можливість завантаження фотографії користувача, вибір предмета одягу з каталогу, автоматичне визначення ключових точок тіла людини, виділення областей тіла, одягу та фону, геометричне перетворення предмета одягу відповідно до пози користувача, формування фінального зображення віртуальної примірки та

перевірка якості отриманого результату. Крім цього, система повинна забезпечувати збереження інформації про створене завдання, його поточний статус, вхідні файли та шлях до сформованого результату. Важливою вимогою є також зручність використання, оскільки користувач повинен мати можливість виконати примірку в межах зрозумілого сценарію онлайн-магазину без необхідності вручну налаштовувати параметри обробки. Система має автоматично виконувати основні операції, а у випадку помилки або недостатньої якості вхідного зображення — повідомляти користувача про проблему та надавати зрозумілий результат обробки.

Очікувана система має складатися з модуля завантаження даних, модуля аналізу пози, модуля сегментації, модуля геометричного перетворення, генеративного модуля, модуля оцінювання якості та користувацького інтерфейсу. Така структура дозволить розділити процес обробки на окремі етапи та забезпечити можливість подальшого вдосконалення кожного з них. Модуль завантаження даних відповідатиме за приймання фотографії користувача та зображення товару, модуль аналізу пози — за визначення ключових точок тіла, модуль сегментації — за виділення необхідних областей, а модуль геометричного перетворення — за узгодження предмета одягу з тілом користувача. Генеративний модуль має виконувати фінальне покращення зображення, усунення артефактів і візуальне узгодження результату, тоді як модуль оцінювання якості повинен перевіряти, чи відповідає отримане зображення базовим вимогам реалістичності та коректності. Користувацький інтерфейс, у свою чергу, забезпечуватиме взаємодію користувача із системою, відображення стану обробки та перегляд фінального результату.

Таким чином, у межах передатестаційної практики було підготовлено теоретичну та проєктну основу для подальшої реалізації кваліфікаційної роботи. Було визначено основні проблеми предметної галузі, проаналізовано ключові етапи віртуальної примірки, розглянуто роль аналізу пози, сегментації, геометричних перетворень і генеративних

моделей у формуванні фінального результату. Також було уточнено загальну структуру майбутньої системи та визначено функціональні вимоги до її основних модулів. Це дозволяє перейти від теоретичного аналізу до практичного проєктування та реалізації програмного рішення, яке може бути використане як частина web-системи онлайн-магазину з підтримкою AI-примірки.

Постановка задачі фіксує межі відповідальності кожного рівня системи: користувацький інтерфейс ініціює сценарій примірки, backend реєструє завдання та контролює обробку, а AI-модуль формує результат на основі підготовлених файлів. Такий поділ робить процес перевірюваним і дозволяє окремо аналізувати помилки завантаження, обробки або генерації [25].

Наприклад, некоректне визначення пози людини може призвести до помилкового масштабування одягу, а неточна сегментація — до неправильного відображення областей перекриття. Тому під час постановки задачі необхідно не лише описати кінцеву мету системи, а й визначити логіку взаємодії між модулями, порядок передавання даних, можливі стани обробки та способи контролю помилок. Такий підхід підвищує керованість системи та спрощує її подальше тестування.

Для інтегрованої web-системи важливо врахувати не лише якість зображень, а й поведінку користувача: повторні натискання кнопки примірки, завантаження надто великих файлів, переривання запиту або тимчасову недоступність AI-сервісу. Тому завдання примірки повинно мати статуси, а frontend — показувати зрозумілий стан обробки без блокування основного інтерфейсу. Окремо необхідно передбачити збереження ідентифікатора запиту, щоб користувач міг отримати результат після завершення генерації, а розробник — відстежити причину помилки. Такий підхід робить систему придатною не тільки для демонстрації, а й для подальшого використання в реальному каталозі товарів.

У результаті аналізу було сформульовано основну задачу: розробити програмну систему, яка отримує фотографію користувача та зображення товару, формує завдання обробки, передає дані в AI-модуль, очікує генерації результату та відображає його користувачу в межах цілісного сценарію онлайн-магазину. Така система повинна поєднувати функції обробки зображень, керування станами завдань, збереження результатів і взаємодії з користувачем. Вона має забезпечувати логічний зв’язок між товаром у каталозі, фотографією користувача, запитом на примірку та сформованим результатом. Завдяки цьому користувач зможе не лише переглянути товар на сторінці магазину, а й оцінити його зовнішній вигляд на власному зображенні. Отже, сформульована задача охоплює як технічні аспекти реалізації AI-обробки, так і прикладний сценарій використання системи в електронній комерції.

1.6 Практичні умови використання віртуальної примірки в онлайн-магазині

Окремою особливістю задачі віртуальної примірки є те, що вона має оцінюватися не лише як технічна демонстрація комп’ютерного зору, а як елемент реального купівельного сценарію. Для користувача важливим є не сам факт роботи моделі, а відповідь на практичне питання: чи допомагає отримане зображення краще уявити товар на собі. Тому під час аналізу предметної галузі доцільно враховувати не тільки якість генерації, а й довіру до результату, сталість відтворення деталей товару та зрозумілість обмежень сервісу. У реальному інтернет-магазині користувач очікує, що система не просто створить візуально привабливе зображення, а збереже основні властивості товару та покаже його максимально наближено до реального вигляду. Якщо результат виглядає красиво, але не відповідає товару з каталогу, така примірка може ввести покупця в оману. Саме тому оцінювання якості віртуальної примірки повинно включати не лише

технічні показники реалістичності, а й відповідність результату практичним потребам користувача.

У випадку інтернет-магазину помилка віртуальної примірки може мати різні наслідки. Незначна неточність контуру або тіні зазвичай сприймається як технічний артефакт, однак зміна кольору, фактури тканини, довжини виробу або розташування декоративних елементів уже може вплинути на рішення про купівлю. Саме тому генеративна модель у такій системі не повинна безконтрольно «домальовувати» товар. Її роль полягає у покращенні стиків, освітлення та локальних дефектів, але не у зміні комерційно важливих характеристик одягу [1], [4].

До таких характеристик можна віднести силует виробу, форму коміра, довжину рукавів, посадку в області талії, наявність принтів, візерунків, гудзиків, блискавок або інших декоративних деталей. Якщо система змінює ці елементи, користувач фактично бачить не той товар, який доступний для купівлі. Тому для задачі електронної комерції важливо забезпечити баланс між реалістичністю зображення та збереженням ідентичності товару.

Для практичного використання сервісу важливо також розрізняти фотографії товару та фотографії користувача. Зображення товару в каталозі зазвичай створюється у контрольованих умовах, має правильне освітлення, чистий фон і стабільну роздільну здатність. Фото користувача, навпаки, може бути зроблене телефоном у кімнаті, на вулиці або перед дзеркалом. Через це система повинна мати механізм попередньої перевірки придатності зображення, а не одразу передавати кожний файл у генеративний модуль. Така перевірка може враховувати наявність людини у кадрі, видимість основних частин тіла, достатній рівень освітлення, відсутність сильного розмиття, правильну орієнтацію зображення та допустимий масштаб фігури. Якщо фотографія не відповідає базовим вимогам, система повинна повідомити користувача про проблему до початку складної AI-обробки. Це дозволяє зменшити кількість невдалих результатів, оптимізувати

використання обчислювальних ресурсів і зробити роботу сервісу більш передбачуваною.

Практичний сценарій використання віртуальної примірки має бути коротким і передбачуваним: вибір товару, завантаження фото, запуск обробки, очікування статусу та перегляд результату. Якщо користувач не розуміє, що відбувається після натискання кнопки, довіра до сервісу знижується навіть за умови якісного зображення. Тому важливо показувати проміжний стан обробки, повідомляти про помилки зрозумілою мовою та пояснювати, які фотографії підходять для кращого результату. Наприклад, система може відображати статуси створення завдання, перевірки зображення, виконання обробки, формування результату та завершення примірки. У разі помилки користувач повинен отримати не технічне повідомлення, а зрозуміле пояснення: недостатнє освітлення, неповна видимість тіла, занадто малий розмір фігури або неможливість визначити ключові точки. Такий підхід покращує взаємодію з сервісом і допомагає користувачеві самостійно підготувати якісніше фото для повторної спроби.

Таким чином, предметна галузь віртуальної примірки включає не тільки алгоритми аналізу пози та генерації зображень, а й вимоги електронної комерції: збереження ідентичності товару, контроль очікувань користувача, захист персональних фото та інтеграцію в наявну логіку каталогу. Саме поєднання цих умов визначає доцільність гібридного підходу, у якому геометричний контроль доповнюється генеративним покращенням, але не замінюється ним повністю. Геометричні методи забезпечують керованість процесу, дозволяють прив’язати одяг до пози та пропорцій користувача, а генеративні моделі покращують візуальну цілісність фінального результату. Окрім цього, система повинна враховувати питання безпеки даних, оскільки фотографії користувачів є персональною інформацією і мають оброблятися з дотриманням принципів конфіденційності.

2 МАТЕМАТИЧНЕ ТА АЛГОРИТМІЧНЕ ЗАБЕЗПЕЧЕННЯ СИСТЕМИ

2.1 Формальна постановка задачі

Формальна модель описує задачу як перетворення пари зображень у фінальну візуалізацію примірки. При цьому важливо окремо задавати вхідне фото користувача, зображення одягу, службові маски, ключові точки та параметри генерації, оскільки саме ці елементи визначають послідовність обчислень і умови повторюваності експерименту [1], [4].

Формальний опис дає змогу визначити, які саме дані повинні передаватися між етапами обробки, у якому вигляді вони зберігаються та який результат очікується на виході системи. Такий підхід спрощує подальше проєктування архітектури, оскільки дозволяє відокремити логіку завантаження файлів, аналізу пози, сегментації, геометричного узгодження, генеративного покращення та оцінювання якості. Крім того, формальна постановка задачі є необхідною для подальшого тестування системи, оскільки вона задає критерії, за якими можна перевірити правильність роботи окремих модулів і всього процесу віртуальної примірки загалом.

В описі вхідних і вихідних даних доцільно розділяти сирі файли, проміжні результати та службові параметри. Фото користувача, зображення товару, маска людини, маска одягу, координати ключових точок, ідентифікатор завдання та URL результату мають зберігатися як пов’язані елементи одного процесу, а не як ізольовані файли. Для цього у структурі системи потрібно фіксувати, який товар був обраний, яке фото використано, які параметри передано до AI-модуля та який файл сформовано після завершення обробки. Така організація даних спрощує повторну перевірку результату, дозволяє аналізувати помилки і створює основу для майбутнього розширення, наприклад історії примірок або порівняння різних товарів на одному фото.

Нехай на вхід системи надходить зображення користувача P, зображення предмета одягу G та метадані товару M, які включають назву, категорію, колір, розмір і текстовий опис. Необхідно побудувати зображення R, на якому користувач зображений у тому самому положенні, але з обраним предметом одягу, узгодженим із контуром тіла та візуальним контекстом початкової фотографії. При цьому результат R повинен зберігати основні характеристики користувача, зокрема позу, пропорції тіла, положення рук, обличчя та загальну композицію кадру. Одночасно система повинна зберегти комерційно важливі властивості товару G, а саме колір, фасон, довжину, форму, декоративні елементи та загальний вигляд виробу. Метадані M використовуються для уточнення категорії одягу, вибору відповідної логіки обробки та контролю того, щоб результат не суперечив опису товару. Отже, задача полягає не лише у створенні візуально реалістичного зображення, а й у забезпеченні відповідності між вхідними даними, проміжними результатами обробки та фінальним зображенням, яке може бути використане користувачем у сценарії онлайн-примірки.

$$P \in R\hat{}(H \times W \times 3), \quad G \in R\hat{}(h \times w \times 4), \quad R = F(P, G, M), \tag{2.1}$$

де P – зображення користувача;

G – зображення одягу з альфа-каналом або маскою;

M – метадані товару;

F – композиція алгоритмічних і генеративних перетворень;

R – результат віртуальної примірки.

Функція F не є єдиною нейронною мережею. У межах роботи вона розглядається як конвеєр, що складається з етапів перевірки, нормалізації, аналізу пози, сегментації, геометричного перетворення, обробки перекриттів, генеративного покращення і збереження результату. Такий підхід робить процес більш пояснюваним та придатним до налагодження.

2.2 Модель представлення зображень і ключових точок

Представлення зображень у системі має забезпечувати зв’язок між піксельними даними та семантичними областями. Фотографія користувача використовується не лише як растрове зображення, а як джерело ключових точок, контурів, масок тіла, меж фону та зон можливого перекриття. Такий опис полегшує подальше узгодження одягу з тілом [5], [8].

Зображення користувача не повинно розглядатися лише як звичайний графічний файл, оскільки для подальшої обробки з нього необхідно отримати структуровану інформацію: межі тіла, області одягу, положення основних частин фігури, фон, а також можливі зони перекриття. Саме тому представлення зображення у вигляді набору масок, контурів і ключових точок дозволяє перейти від неструктурованих піксельних даних до формалізованого опису, який може бути використаний іншими модулями системи. Такий підхід забезпечує більш точне геометричне узгодження предмета одягу, спрощує контроль проміжних результатів і дозволяє виявляти помилки ще до етапу генеративного покращення.

Побудова масок і контурів потребує окремої уваги до меж між тілом, одягом і фоном. Навіть якщо ключові точки визначені правильно, неточна маска може спричинити перекриття рук, втрату частини виробу або появу різкого шва на межі композиції. Тому результати сегментації мають уточнюватися з урахуванням силуету та області майбутнього накладання. Для верхнього одягу особливо важливими є лінія плечей, шия, руки та зона грудей, а для довших виробів — талія, стегна і нижній край. Якщо ці області оброблені неточно, генеративна модель може приховати частину помилки, але не завжди здатна відновити правильну посадку одягу. Саме тому маска розглядається як ключовий проміжний результат між аналізом пози та фінальною композицією.

Поза людини описується множиною ключових точок K. Кожна точка має координати у площині зображення та, за можливості, оцінку

впевненості. У практичній реалізації особливе значення мають точки плечей, шиї, ліктів, зап’ясть і стегон, оскільки саме вони формують каркас для верхнього одягу. Наприклад, координати плечей і шиї використовуються для визначення положення коміра, верхньої лінії виробу та ширини посадки, точки ліктів і зап’ясть допомагають врахувати положення рук, а точки стегон і талії дозволяють уточнити нижню межу сорочки, футболки, куртки або сукні. Оцінка впевненості для кожної ключової точки є важливою, оскільки вона дозволяє системі визначити, наскільки надійно знайдено відповідну частину тіла. Якщо певна точка має низьку впевненість або не визначена через перекриття, система може використати сусідні точки, контур тіла або результати сегментації для приблизного відновлення її положення. Таким чином, множина ключових точок K є основою для побудови умовної скелетної моделі користувача, яка надалі використовується для геометричного узгодження предмета одягу з тілом та перевірки коректності фінального результату.

$$K = \{(x_i, y_i, c_i)\}_(i = 1)\^n, \quad 0 \le x_i < W, \quad 0 \le y_i < H, \qquad (2.2)$$

де x_i, y_i – координати і-ї ключової точки;

c_i – оцінка впевненості;

n – кількість ключових точок, що використовуються алгоритмом.

Для оцінювання пропорцій тіла застосовується відстань між ключовими точками. Наприклад, ширина плечей може бути оцінена як евклідова відстань між лівим і правим плечем. Цей параметр використовується для масштабування одягу та визначення початкового положення верхньої межі виробу.

$$d(p, q) = sqrt((x_p - x_q)\^2 + (y_p - y_q)\^2), \qquad (2.3)$$

де p і q – дві ключові точки;

d(p, q) – відстань між ними у пікселях.

2.3 Афінні перетворення, гомографія та TPS-викривлення

Геометрична модель узгодження задає правила переходу від координат товару до координат тіла користувача. Для цього використовуються опорні точки, локальні області деформації та перетворення, які мають зберігати форму виробу настільки, наскільки це можливо після адаптації до пози людини. Помилки на цьому етапі найчастіше проявляються у зміщенні рукавів, перекосі коміра або неприродній лінії низу [2], [11], [12].

Під час перенесення одягу в координати тіла користувача система використовує не абсолютні розміри зображення, а відносні опорні точки. Це дозволяє адаптувати той самий товар до різних фотографій, не прив’язуючи алгоритм до одного масштабу або фіксованого положення людини у кадрі. Наприклад, ширина плечей може визначати базовий масштаб, нахил лінії плечей — кут повороту, а контур тулуба — допустиму область локального викривлення. У результаті предмет одягу отримує початкове положення, яке надалі може бути уточнене композиційним або генеративним модулем.

Найпростішим способом змінити положення зображення одягу є афінне перетворення. Воно охоплює масштабування, поворот, зсув і нахил. Для задачі верхнього одягу афінне перетворення може використовуватися як початкове наближення, коли відомі координати плечей і приблизний центр тулуба.

$$[x'\ y'\ 1]\text{^}T\ =\ A\ [x\ y\ 1]\text{^}T,\quad A\ =\ [[a11\ a12\ tx],[a21\ a22\ ty],[0\ 0\ 1]], \qquad (2.4)$$

де A – матриця афінного перетворення;

tx, ty – параметри зсуву;

a11, a12, a21, a22 – параметри масштабування, повороту та нахилу.

Для складніших випадків, коли поверхня одягу повинна змінити форму відповідно до силуету тіла, використовується нелінійне викривлення. Одним із поширених підходів є Thin-Plate Spline, який будує гладку функцію деформації за набором контрольних точок.

$$f(x,y) = a_1 + a_x\, x + a_y\, y + \Sigma_(i=1)\hat{}m\, w_i\, U(||(x,y) - p_i||), \quad (2.5)$$

де p_i – контрольні точки;

w_i – вагові коефіцієнти;

U(r)=r^2 ln(r^2) – радіальна базисна функція TPS;

m – кількість контрольних точок.

Узагальнений конвеєр алгоритму віртуальної примірки подано на рисунку 2.1.



Рисунок 2.1 – Узагальнений конвеєр алгоритму віртуальної примірки

2.4 Обробка перекриттів, композиція та оцінювання якості

Композиція фінального результату відповідає за те, які фрагменти зображення залишаються з початкової фотографії, а які замінюються або уточнюються після примірки. На цьому етапі особливо важливо коректно обробити межі маски, руки, волосся, відкриті ділянки шкіри та тіні, щоб результат не виглядав як механічне накладання [3], [4].

Поєднання деформованого одягу з початковим зображенням вимагає акуратної роботи з областями перекриття. Якщо руки або волосся мають

залишатися поверх нового виробу, відповідні фрагменти потрібно зберегти з оригінального фото або відновити під час генеративного уточнення, щоб композиція виглядала природно. Крім того, необхідно враховувати різницю в освітленні між фотографією користувача та зображенням товару. Без корекції яскравості, тіней і меж накладання навіть правильно деформований одяг може виглядати як сторонній об’єкт.

Після геометричного перетворення потрібно визначити, які частини одягу мають бути видимими, а які повинні бути перекриті руками, волоссям, обличчям або іншими об’єктами. Для цього використовується маска видимості, яка поєднує маску одягу, маску людини та правила пошарового відображення.

$$R(x,y) = \alpha(x,y) \cdot G_w(x,y) + (1 - \alpha(x,y)) \cdot P(x,y), \qquad (2.6)$$

де R(x, y) – піксель результату;

G_w(x, y) – піксель викривленого одягу;

P(x, y) – піксель початкової фотографії;

α(x, y) – маска змішування.

Оцінювання якості результату виконується за сукупністю критеріїв. До них належать коректність формату вхідних даних, наявність критичних ключових точок, адекватність маски, відсутність грубих артефактів, збереження обличчя та фону, а також візуальна узгодженість одягу з тілом користувача.

$$Q = w_1\, Q_pose + w_2\, Q_mask + w_3\, Q_align + w_4\, Q_real, \qquad (2.7)$$

де Q – інтегральна оцінка якості;

Q_pose – якість визначення пози;

Q_mask – якість маски;

Q_align – якість геометричного вирівнювання;

Q_real – реалістичність результату;

w_i – ваги критеріїв.

2.5 Алгоритм підготовки даних перед запуском генеративної моделі

Перед запуском генеративної моделі система повинна привести вхідні дані до узгодженого вигляду. Цей етап не є допоміжним, оскільки саме він визначає, чи отримає модель коректні межі тіла, область заміни одягу та зображення товару без зайвого фону. Якщо підготовка виконана поверхнево, навіть сучасна дифузійна модель може сформувати результат, у якому одяг виглядає реалістично, але розташований неточно або частково перекриває руки й обличчя користувача [4], [13].

У межах розроблюваної системи підготовка даних розглядається як послідовність перевірок. Спочатку файл перевіряється за типом, розміром і можливістю відкриття бібліотекою обробки зображень. Далі виконується нормалізація роздільної здатності, оскільки надто великі файли збільшують час обробки, а надто малі не містять достатньої кількості деталей. Після цього формується робоча копія зображення, з якою працюють модулі аналізу пози, сегментації та геометричного узгодження.

Наступним кроком є перевірка структурної придатності фотографії. Для верхнього одягу критичними є видимість плечей, грудної області та частини рук. Якщо ключові точки плечей або тулуба відсутні, система не повинна мовчки запускати генерацію, оскільки результат буде випадковим. Більш коректною поведінкою є повернення статусу помилки з поясненням, що потрібно завантажити фото людини у повний зріст або принаймні з добре видимою верхньою частиною тіла.

Підготовка зображення товару має іншу логіку. Для каталожного фото важливо зберегти форму, колір і декоративні елементи виробу. Якщо файл містить прозорий фон, використовується альфа-канал або маска. Якщо прозорість відсутня, необхідно виділити контур виробу, а зайві області фону не повинні потрапляти до генеративної моделі. Саме на цьому етапі

геометричні методи та сегментація працюють як обмеження для подальшого синтезу.

Після завершення перевірок система формує набір службових даних: ідентифікатор завдання, шляхи до нормалізованих файлів, параметри обробки, статус черги та очікуваний формат результату. Така організація дозволяє повторити обробку, порівняти різні параметри генерації та відокремити помилки вхідних даних від помилок AI-сервісу. Для кваліфікаційної роботи це важливо, оскільки підвищує відтворюваність експериментів і робить алгоритмічний конвеєр прозорим для аналізу.

3 ПРОЄКТУВАННЯ ТА РЕАЛІЗАЦІЯ ПРОГРАМНОЇ СИСТЕМИ

3.1 Вимоги до системи та сценарії використання

Вимоги до системи сформовано з урахуванням реального сценарію онлайн-магазину: користувач повинен швидко вибрати товар, завантажити фото, запустити примірку та отримати зрозумілий результат. Тому технічна частина має підтримувати перевірку файлів, збереження статусу завдання, обробку помилок і повернення посилання на сформоване зображення [25].

У сценарії онлайн-магазину система має працювати передбачувано навіть тоді, коли користувач виконує дії неідеально: обирає товар без завантаженого фото, надсилає файл неправильного формату або повертається на сторінку товару після запуску примірки. Для цього потрібні перевірки, повідомлення про помилки та збереження стану завдання. Інтерфейс не повинен створювати враження зависання, тому очікування результату супроводжується статусом обробки. Це дозволяє поєднати AI-функцію з типовою логікою e-commerce без ускладнення основного сценарію покупки.

Функціональні вимоги до системи наведено в таблиці 3.1.

Таблиця 3.1 – Функціональні вимоги до системи

№	Вимога	Опис реалізації
FR-1	Перегляд каталогу	Користувач може переглядати список товарів, виконувати пошук і фільтрацію
FR-2	Сторінка товару	Система відображає фото, назву, ціну, опис, розмір і колір товару
FR-3	Кошик	Користувач може додавати товар, змінювати кількість і видаляти позиції
FR-4	Завантаження фото	Користувач передає власне зображення для запуску віртуальної примірки
FR-5	AI-обробка	Сервер створює асинхронне завдання та передає дані в модуль генерації

Продовження таблиці 3.1

№	Вимога	Опис реалізації
FR-6	Отримання результату	Після завершення користувач отримує URL згенерованого зображення

Наведені функціональні вимоги визначають основні можливості, які повинна забезпечувати система віртуальної примірки одягу. Вони описують користувацький сценарій від перегляду каталогу товарів і роботи з кошиком до завантаження власного фото та отримання згенерованого результату. Особливу роль у системі відіграє AI-обробка, оскільки саме вона забезпечує формування зображення з приміркою обраного товару. Нефункціональні вимоги до системи наведено в таблиці 3.2.

Таблиця 3.2 – Нефункціональні вимоги

Категорія	Вимога	Обґрунтування
Надійність	Перевірка формату та розміру файлів	Захист AI-модуля від некоректних даних
Продуктивність	Асинхронна обробка задач	Запобігання блокуванню користувацького інтерфейсу
Масштабованість	Модульна структура	Можливість заміни AI-провайдера або бази даних
Зручність	Простий сценарій взаємодії	Користувач не повинен знати технічні деталі обробки
Безпека	Обмеження частоти запитів і валідація файлів	Зменшення ризику зловживань

3.2 Архітектура застосунку та структура проєкту

Архітектура системи побудована як взаємодія трьох рівнів: frontend забезпечує сценарій користувача, backend керує даними та запитами, а AI-рівень виконує ресурсоємну генерацію. Така структура дозволяє масштабувати окремі компоненти незалежно й не перевантажувати інтерфейс тривалими обчисленнями [25].

Організація frontend, backend та AI-рівня передбачає різну відповідальність кожного компонента. Клієнтська частина не повинна виконувати важкі обчислення, backend має контролювати життєвий цикл запиту, а AI-рівень — працювати як окремий сервіс генерації. Такий поділ спрощує тестування і подальше розгортання. Якщо AI-модуль потребує більше ресурсів або переноситься на інший сервер, інші частини застосунку можуть залишатися без суттєвих змін. Це також спрощує діагностику, оскільки помилки інтерфейсу, API або генерації можна аналізувати окремо.

Загальну архітектуру системи віртуальної примірки наведено на рисунку 3.1.

Компоненти користувацького застосунку та API подано на рисунку 3.2.

Репозиторій проєкту містить окремі каталоги для AI-ядра, backend-застосунку, frontend-застосунку, документації, тестів, сховища часу виконання та Docker-конфігурації. Така структура відповідає модульному підходу і дозволяє розвивати окремі частини незалежно одна від одної. Основні компоненти програмного проєкту наведені в таблиці 3.3.

Таблиця 3.3 – Основні компоненти програмного проєкту

Компонент	Технологія	Призначення
Frontend	Next.js, React, TypeScript	Інтерфейс каталогу, сторінки товару, кошика та AI-примірки
Backend	FastAPI, Python	REST API, маршрути, валідація, обробка запитів
AI Engine	Python, Gradio client, IDM-VTON	Генерація результату примірки
Storage	Файлова система	Збереження фото користувача, товарів, результатів і артефактів
Database	SQLite/PostgreSQL	Збереження товарів, замовлень, завантажень і завдань
Docker	Docker Compose	Розгортання сервісів на сервері

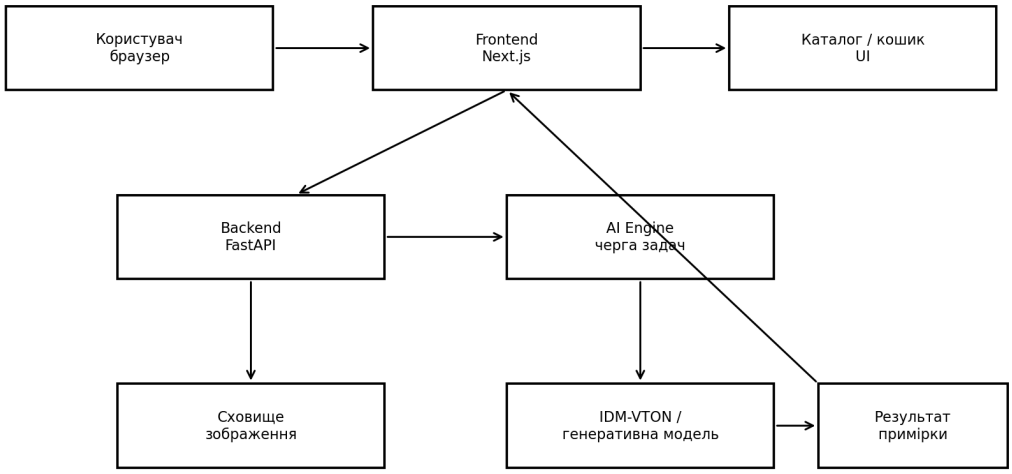


Рисунок 3.1 – Архітектура системи віртуальної примірки

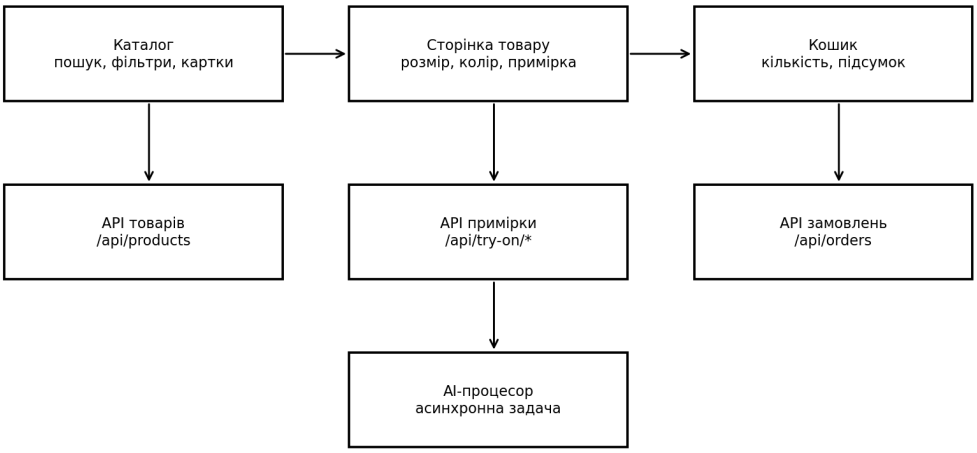


Рисунок 3.2 – Компоненти користувацького застосунку та API

3.3 Реалізація backend, API та асинхронної обробки

Backend-рівень виконує роль координатора між користувацьким інтерфейсом, файловим сховищем, базою даних і AI-модулем. Він приймає файли, створює запис завдання, повертає його статус, обробляє помилки та надає клієнтській частині шлях до готового результату. Завдяки цьому логіка примірки не змішується з логікою відображення сторінок [16], [19], [25].

Для обробки запитів і завантажень backend перевіряє тип файлу, розмір, наявність необхідних параметрів та створює окреме AI-завдання. Асинхронний підхід дозволяє не чекати завершення генерації в межах одного HTTP-запиту, а періодично отримувати статус і результат після завершення обробки. Це особливо важливо для AI-примірки, оскільки час генерації може залежати від завантаженості зовнішнього сервісу, розміру зображення та складності сцени. У випадку помилки backend повинен повернути зрозумілий стан, а не лише технічний збій.

Серверна частина побудована за принципом поділу маршрутів (лістинг 3.1). Окремі маршрутизатори відповідають за товари, замовлення, завантаження файлів, користувацький сценарій примірки та внутрішній AI-рівень. Такий поділ знижує зв’язність коду та полегшує підтримку.

Лістинг 3.1 – Реєстрація основних маршрутів FastAPI

```
app = FastAPI(
    title="AI Virtual Try-On Service",
    description="AI-powered virtual clothing try-on",
    version="2.0.0",
    lifespan=lifespan,
)

app.include_router(products.router)
app.include_router(orders.router)
app.include_router(tryon.router)
app.include_router(uploads.router)
app.include_router(ai_engine.router)
```

Для обробки примірки використовується механізм завдань. Після завантаження фото створюється запис TryOnJob зі статусом QUEUED. Надалі завдання переходить у статус PROCESSING, DONE або

FAILED (лістинг 3.2).. Це дозволяє не блокувати HTTP-запит на весь час генерації зображення.

Лістинг 3.2 – Модель статусів завдання примірки

```
 class JobStatus(str, Enum):
    QUEUED = "QUEUED"
    PROCESSING = "PROCESSING"
    DONE = "DONE"
    FAILED = "FAILED"

class Job(BaseModel):
    job_id: str
    status: JobStatus = JobStatus.QUEUED
    user_image_path: Optional[str] = None
    product_image_path: Optional[str] = None
    result_image_url: Optional[str] = None
    quality_score: Optional[float] = None
```

Основні API-ендпоінти системи наведені в таблиці 3.4.

Таблиця 3.4 – Основні API-ендпоінти системи

Метод	Шлях	Призначення
GET	/api/products	Отримання каталогу товарів
POST	/api/orders	Створення замовлення
POST	/api/try-on/upload	Завантаження фото та запуск примірки
GET	/api/try-on/{id}	Отримання статусу завдання
POST	/ai/process	Внутрішній запуск AI-обробки
GET	/health	Перевірка стану сервісу

3.4 Реалізація AI-модуля та інтеграція з IDM-VTON

AI-модуль відповідає за підготовку даних до IDM-VTON, запуск генерації та отримання сформованого зображення. Його робота залежить від коректності вхідних файлів, параметрів запиту та доступності зовнішнього

або локального середовища виконання. Саме тому в системі доцільно ізолювати AI-обробку від основних маршрутів backend-частини [4], [20], [25], [26].

Під час інтеграції з генеративною моделлю важливо передавати їй підготовлені вхідні дані у стабільному форматі: фото користувача, зображення обраного товару та параметри режиму примірки. Якщо зовнішній сервіс повертає помилку або недоступний, система повинна зберегти це як стан failed і не втрачати інформацію про початковий запит. Для практичного використання також важливо обмежити кількість повторних спроб і логувати основні події, щоб можна було відрізнити помилку мережі від проблеми з вхідним зображенням або параметрами моделі.

У практичній реалізації генерація результату виконується через Gradio API для простору IDM-VTON (лістинг 3.3). Такий підхід дозволяє використати сучасну модель віртуальної примірки без обов'язкового локального розгортання великих ваг моделі на кожному робочому комп'ютері. При цьому архітектура системи зберігає можливість заміни зовнішнього провайдера на локальний inference-сервер.

Лістинг 3.3 – Виклик генеративної моделі через Gradio API

```
 result = client.predict(
dict={
    "background": handle_file(person_image_path),
    "layers": [],
    "composite": None,
},
garm_img=handle_file(garment_image_path),
garment_des=garment_description,
denoise_steps=denoise_steps,
seed=seed,
api_name="/tryon",
```

)

Після отримання результату тимчасовий файл копіюється у каталог результатів і стає доступним через статичний маршрут. Завдання позначається як успішно завершене, а користувач отримує URL готового зображення. Якщо API повертає помилку або результат не має очікуваного формату, завдання переводиться у статус FAILED із відповідним кодом помилки (рисунок 3.3).

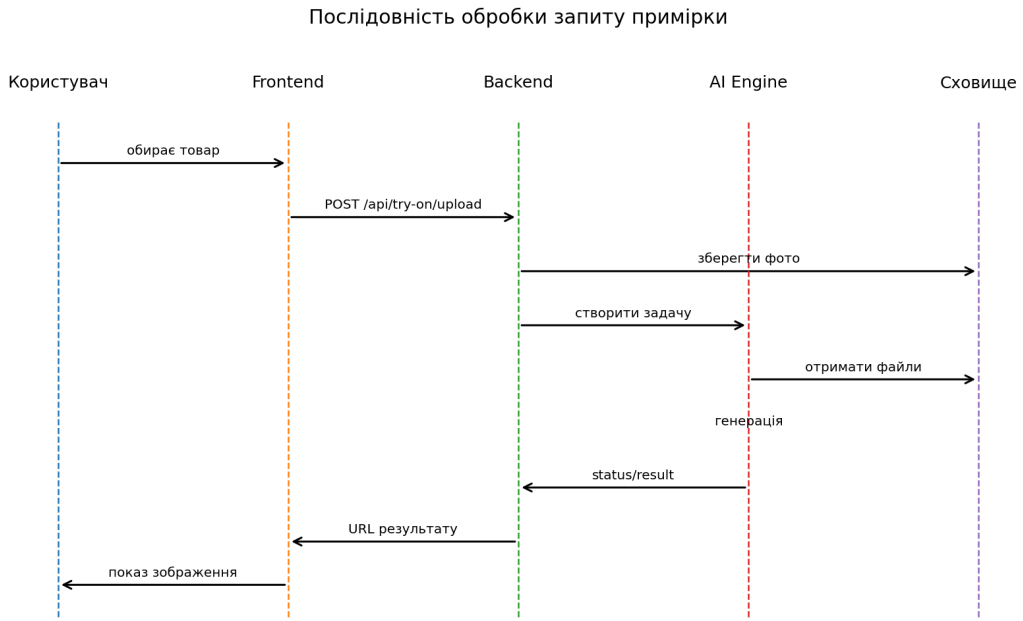


Рисунок 3.3 – Послідовність обробки запиту примірки

3.5 Реалізація frontend та користувацького інтерфейсу

Frontend-рівень подає складний процес примірки у вигляді звичного сценарію електронної комерції. Користувач взаємодіє з головною сторінкою, каталогом, карткою товару, кошиком і кнопкою запуску AI-примірки, тоді як технічні етапи обробки приховані за станами інтерфейсу та повідомленнями про результат [17], [18], [25].

Взаємодія користувача з каталогом, товаром і кошиком має залишатися зрозумілою незалежно від того, чи запускається AI-примірка. Тому кнопки, повідомлення, індикатори завантаження та блок результату повинні пояснювати, що саме відбувається: файл прийнято, завдання створено, результат готується або сталася помилка. Такий підхід зменшує невизначеність для користувача і не порушує звичний сценарій покупки. AI-примірка виступає додатковою функцією, а не замінює основні елементи каталогу та оформлення замовлення.

Клієнтська частина реалізована як web-застосунок із темним візуальним стилем, акцентним фіолетовим кольором, каталогом товарів, фільтрами, сторінкою детального перегляду та кошиком. Інтерфейс має демонструвати не лише технічну можливість примірки, а й повний сценарій електронної комерції, у якому AI-модуль стає природною частиною вибору товару.

Основні сторінки web-застосунку STYLEAI демонструють логіку взаємодії користувача із системою. Головну сторінку web-застосунку STYLEAI наведено на рисунку 3.4. Сторінку каталогу товарів із фільтрами подано на рисунку 3.5. Сторінку товару з кнопкою віртуальної примірки наведено на рисунку 3.6. Сторінку кошика з підсумком замовлення подано на рисунку 3.7.

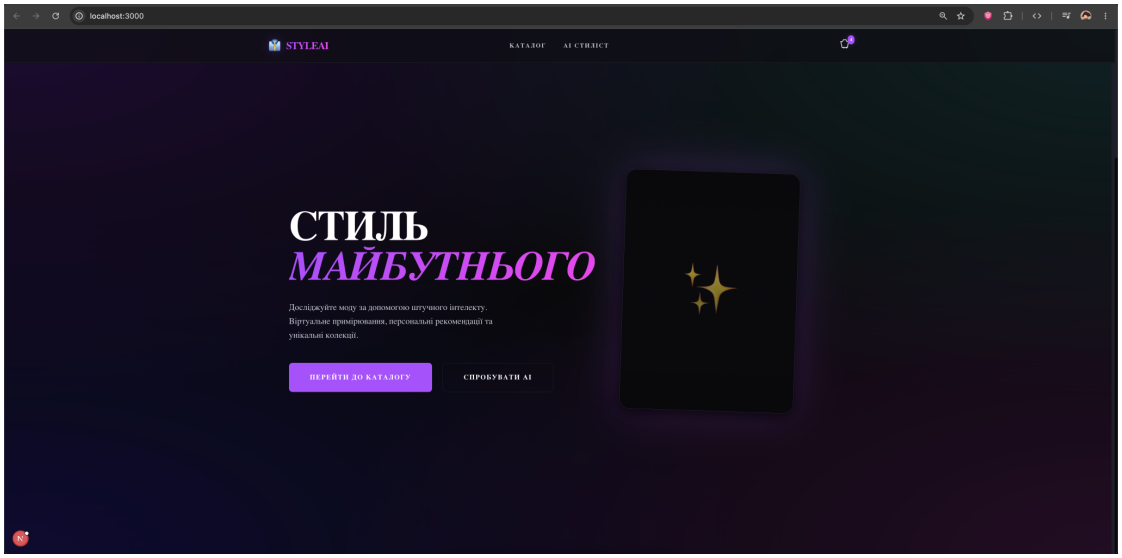


Рисунок 3.4 – Головна сторінка web-застосунку STYLEAI

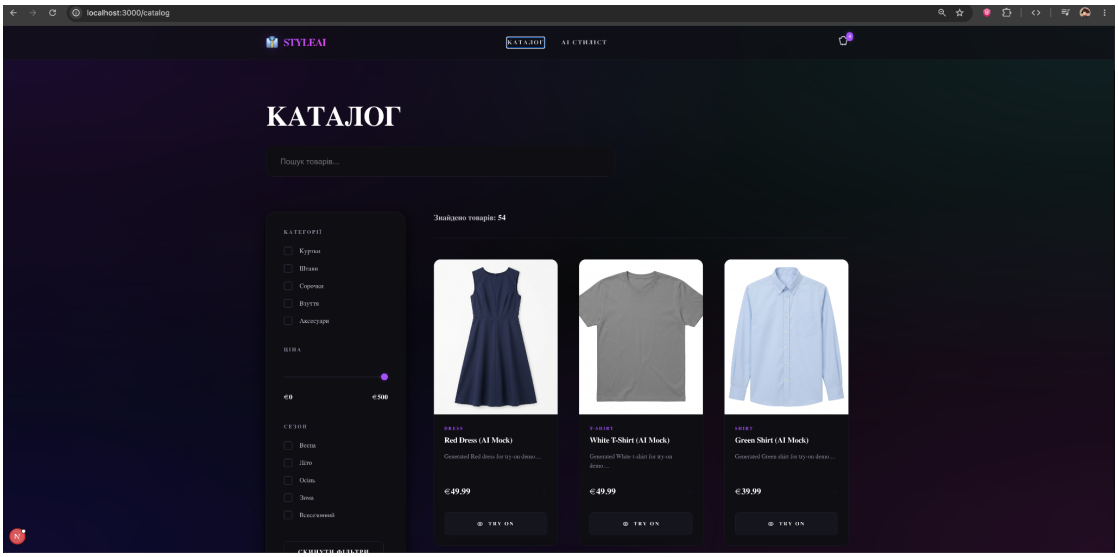


Рисунок 3.5 – Сторінка каталогу товарів із фільтрами

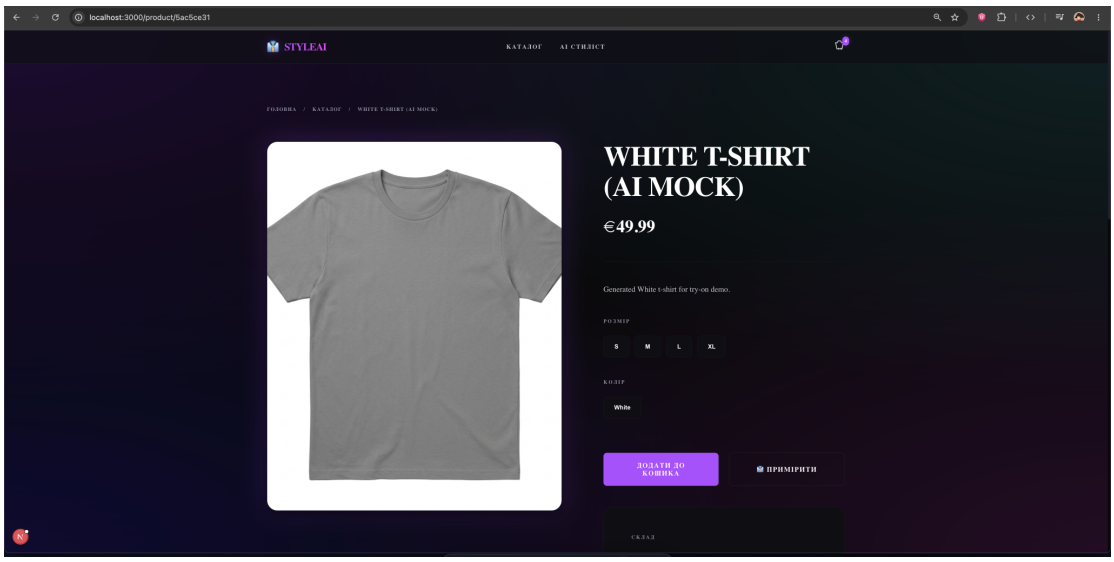


Рисунок 3.6 – Сторінка товару з кнопкою віртуальної примірки

Ключовими поняттями для цього підрозділу є Next.js, каталог, картка товару, кнопка примірки, кошик, локальний сервер. Вони визначають спосіб

організації користувацької взаємодії та передачі даних між інтерфейсом і серверною частиною системи. Якщо на етапі побудови цих представлень не врахувати зв’язок між товаром, фотографією користувача та результатом примірки, система може сформувати некоректний сценарій обробки або повернути результат, що не відповідає вибраному предмету одягу.

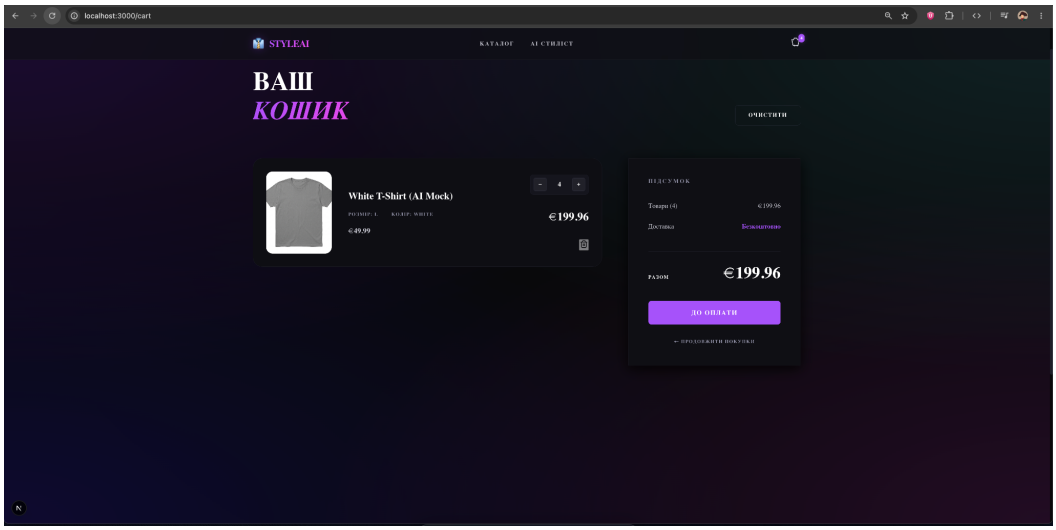


Рисунок 3.7 – Сторінка кошика з підсумком замовлення

3.6 Модель даних, збереження файлів та розгортання

Модель даних забезпечує зв’язок між товарами, замовленнями, завантаженими файлами, завданнями примірки та результатами генерації. Такий зв’язок потрібний для відстеження історії обробки, повторного відкриття результату, діагностики помилок і подальшого розширення системи, наприклад додавання профілів користувачів або історії примірок [19], [25].

Під час збереження товарів, завантажень і результатів важливо підтримувати логічні зв’язки між сутностями. Наприклад, завдання примірки має містити посилання на конкретний товар, фото користувача, статус обробки та файл результату. Це дозволяє відтворити сценарій обробки й спростити перевірку помилок. У майбутньому така модель може

бути доповнена історією примірок, повторним запуском генерації або порівнянням кількох товарів для одного користувача. Тому структура даних повинна бути достатньо простою для прототипу, але не закривати можливість розвитку системи.

Система оперує кількома основними сутностями: Product, Order, OrderItem, Upload, TryOnJob та Result. Товар містить комерційні дані та посилання на зображення; Upload описує файл, завантажений користувачем; TryOnJob зберігає статус обробки, якість, шлях до результату та повідомлення про помилки. Така модель забезпечує трасованість обробки та дозволяє аналізувати проблемні випадки. Спрощену модель даних системи наведено на рисунку 3.8.

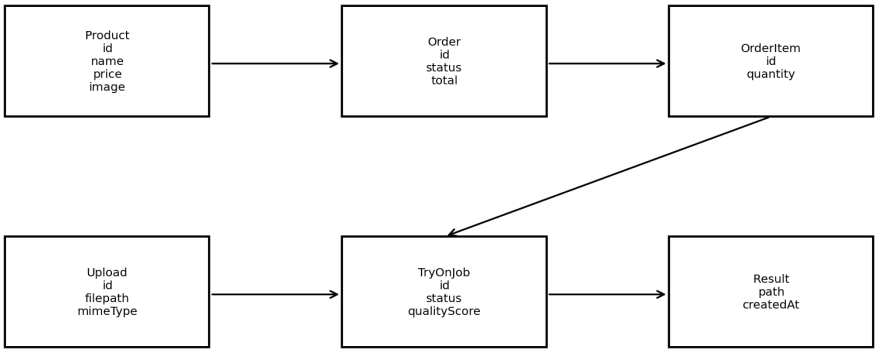


Рисунок 3.8 – Спрощена модель даних системи

Основні стани завдання віртуальної примірки подано в таблиці 3.5.

Таблиця 3.5 – Стан завдання віртуальної примірки

Статус	Значення	Дія системи
QUEUED	Завдання створено	Очікування запуску AI-обробки
PROCESSING	Обробка триває	Файли передано до AI-модуля
DONE	Генерацію завершено	Результат доступний за URL
FAILED	Помилка	Користувач отримує повідомлення, а лог містить причину

3.7 Обробка помилок, захист файлів і журналювання запитів

Під час проєктування програмної системи важливо враховувати, що AI-примірка є ресурсомісткою операцією і не повинна виконуватися так само, як звичайне відкриття сторінки товару. Запит користувача може містити великий файл, некоректне зображення, дубльовану спробу або ситуацію, коли зовнішній AI-сервіс тимчасово недоступний. Тому backend повинен не тільки приймати запит, а й керувати його життєвим циклом: від валідації до збереження результату або фіксації помилки.

Для цього у системі доцільно використовувати чіткі статуси завдання: created, validating, processing, completed та failed. Такий підхід спрощує взаємодію з frontend-частиною, оскільки інтерфейс може періодично запитувати стан завдання й показувати користувачеві зрозуміле повідомлення. Якщо помилка виникла на етапі валідації, користувачеві можна запропонувати завантажити інше фото. Якщо помилка пов’язана із зовнішнім AI-модулем, система може запропонувати повторити спробу пізніше.

Захист файлів у такій системі має практичне значення, оскільки користувач передає власне зображення. На рівні backend потрібно обмежити допустимі формати, перевіряти MIME-тип і розширення, контролювати максимальний розмір файлу та не використовувати початкову назву файлу як частину системного шляху. Для зберігання доцільно створювати службові імена на основі унікального ідентифікатора завдання, щоб уникнути конфліктів і випадкового перезапису даних [15], [25].

Журналювання запитів потрібне не для накопичення персональних даних, а для технічного аналізу роботи системи. У журналі достатньо зберігати час створення завдання, статус, тривалість обробки, тип помилки та службовий ідентифікатор. Самі зображення не повинні без потреби потрапляти до логів. Такий підхід дозволяє знаходити слабкі місця конвеєра, не перетворюючи журнал на сховище чутливої інформації.

У розробленому прототипі обробка помилок пов’язана з асинхронною природою AI-запиту. Frontend не очікує блокуючої відповіді з готовим зображенням, а працює зі статусом і URL результату. Це робить систему стійкішою до тривалих операцій генерації та дозволяє в майбутньому додати чергу завдань, повторні спроби, обмеження частоти запитів і очищення тимчасових файлів після завершення обробки.

З погляду подальшого розвитку системи корисним є збереження не лише фінального результату, а й мінімального набору технічних параметрів обробки. До таких параметрів можна віднести розмір вхідного зображення, тип обраного товару, тривалість генерації, результат валідації та код помилки у разі невдалого завершення. Ці дані можуть використовуватися для аналізу продуктивності, пошуку типових проблем і покращення алгоритму без необхідності переглядати самі зображення користувачів.

Таким чином, обробка помилок, захист файлів і журналювання запитів є не допоміжними, а повноцінними елементами архітектури системи віртуальної примірки. Вони забезпечують стабільність роботи прототипу, зменшують ризик некоректної обробки даних, покращують взаємодію з користувачем і створюють основу для переходу від локальної демонстраційної версії до більш надійного web-сервісу.

4 ЕКСПЕРИМЕНТАЛЬНІ ДОСЛІДЖЕННЯ ТА АНАЛІЗ РЕЗУЛЬТАТІВ

4.1 Методика тестування

Методика тестування орієнтована на перевірку не лише окремих функцій, а повного користувацького маршруту від відкриття каталогу до отримання результату примірки. Окремо оцінюються завантаження файлів, створення AI-завдання, коректність статусів, стабільність API та візуальна придатність сформованого зображення [23], [24].

Перевірка функціональності та стійкості має охоплювати типові й проблемні сценарії: коректне відкриття сторінок, роботу каталогу, додавання товару до кошика, завантаження фото, запуск примірки, отримання результату, а також реакцію системи на неправильний формат файлу або збій AI-обробки. Для кожного сценарію фіксується очікувана поведінка системи, фактичний результат і можливі відхилення. Такий підхід дозволяє оцінити не тільки демонстраційну працездатність, а й готовність прототипу до подальшого вдосконалення.

Тестування системи виконувалося за трьома групами критеріїв: функціональна коректність, візуальна коректність і технічна стабільність. До функціональної коректності належать сценарії перегляду каталогу, відкриття сторінки товару, додавання товару до кошика, запуск примірки й отримання результату. Візуальна коректність оцінюється за відповідністю сторінок макету, читабельністю елементів і відсутністю критичних помилок відображення. Технічна стабільність визначається коректністю API-відповідей, обробкою помилок і збереженням статусів завдання. Тестові сценарії системи подано в таблиці 4.1.

Таблиця 4.1 – Тестові сценарії системи

№	Сценарій	Очікуваний результат	Статус
1	Відкриття головної сторінки	Відображено назву, навігацію і кнопки переходу	успішно
2	Перехід до каталогу	Відображено товари, фільтри та пошук	успішно
3	Відкриття сторінки товару	Відображено фото, ціну, розмір, колір і кнопку примірки	успішно
4	Додавання товару до кошика	Товар з’являється у кошику, сума перераховується	Успішно
5	Завантаження фото для примірки	Створюється завдання та повертається ідентифікатор	перевірено
6	Отримання результату AI	Система повертає URL результату або статус помилки	перевірено

4.2 Перевірка інтерфейсу та основних функцій

Перевірка інтерфейсу зосереджена на тому, наскільки зрозуміло користувач проходить основні етапи роботи з web-застосунком. Для цього аналізуються головна сторінка, каталог, сторінка товару, кошик і блок AI-примірки, а також наявність зворотного зв’язку під час завантаження, очікування та завершення обробки [25].

Під час аналізу сторінок інтерфейсу окремо перевірялися логіка навігації, доступність основних кнопок, зрозумілість візуальних акцентів і послідовність дій користувача. Такий підхід дозволяє оцінити не лише наявність сторінок, а й те, чи формують вони єдиний завершений сценарій онлайн-покупки з AI-приміркою. Важливо, щоб користувач міг без додаткових інструкцій перейти від перегляду товару до завантаження фото, запуску обробки та перегляду результату. Це підвищує практичну цінність прототипу для e-commerce.

Головна сторінка виконує роль візуального входу до системи. Вона має демонструвати основну ідею продукту – стиль майбутнього та AI-примірку. Важливо, що вже на цьому рівні користувач бачить два основні

сценарії: перехід до каталогу та спробу AI-функції. Це дозволяє зменшити когнітивне навантаження і швидко зрозуміти призначення застосунку.

Каталог реалізує основний комерційний сценарій. Наявність фільтрів за категоріями, ціною та сезоном створює основу для подальшого розширення системи, наприклад додавання рекомендацій, персональних добірок або AI-стиліста. Сторінка товару містить кнопку примірки, що є точкою інтеграції генеративної моделі з користувацьким інтерфейсом.

Кошик підтверджує, що система не обмежується лише демонстрацією AI-можливості, а має повний сценарій електронної комерції. Користувач може змінювати кількість товару, бачити проміжну суму, доставку та загальну суму до оплати. Це важливо для дипломного проєкту, оскільки демонструє практичну придатність розробки.

Результати перевірки сторінок інтерфейсу подано в таблиці 4.2.

Таблиця 4.2 – Перевірка сторінок інтерфейсу

Сторінка	Перевірені елементи	Результат
Головна	Навігація, логотип, CTA-кнопки, hero-блок	Відображається коректно
Каталог	Фільтри, пошук, картки товарів, кнопка Try On	Відображається коректно
Товар	Фото, ціна, розмір, колір, додавання до кошика, примірка	Відображається коректно
Кошик	Позиції, кількість, видалення, підсумок	Відображається коректно

4.3 Аналіз якості, обмежень і напрямів покращення

Оцінювання якості результатів враховує як технічні, так і візуальні критерії. До технічних належать успішність обробки, відсутність помилок API та доступність файлу результату, а до візуальних — відповідність одягу контурам тіла, збереження деталей товару, природність меж і відсутність очевидних артефактів [3], [4], [26].

Оцінювання обмежень системи пов’язане з якістю фото, складністю пози, типом одягу та доступністю обчислювальних ресурсів.

Найскладнішими залишаються випадки сильного повороту тіла, перекриття рук, складного фону, низької роздільної здатності та нестандартних предметів одягу, для яких генеративна модель може створювати помітні артефакти. Також потрібно враховувати час очікування результату, оскільки AI-обробка є значно повільнішою за звичайне відкриття сторінки товару. Тому практичне використання системи потребує чітких повідомлень, контролю помилок і поступового покращення якості моделі.

Якість віртуальної примірки залежить не лише від генеративної моделі, а й від якості вхідних зображень. Якщо користувач завантажує фото з низькою роздільною здатністю, складним фоном, сильним поворотом тіла або перекриттям тулуба, система може сформувати результат із артефактами. Тому важливо не лише запускати модель, а й перевіряти придатність вхідних даних перед обробкою. Основні обмеження системи та способи їх усунення подано в таблиці 4.3.

Таблиця 4.3 – Основні обмеження системи та способи їх усунення

Обмеження	Прояв	Напрям покращення
Низька якість фото	Розмитість або некоректна поза	Додати попередню перевірку фото та підказки користувачу
Складні перекриття	Руки або волосся можуть перекривати одяг	Покращити сегментацію і Z-order правила
Залежність від зовнішнього API	Час очікування або недоступність сервісу	Передбачити локальний inference або резервний провайдер
Неповна підтримка категорій	Краще працює з верхнім одягом	Розширити моделі для нижнього одягу та аксесуарів
Суб’єктивність оцінки якості	Складно автоматично визначити реалістичність	Додати метрики та анкетування користувачів

У подальшому систему доцільно розширити модулем автоматичного аналізу придатності фотографії, локальною підтримкою моделей сегментації, кешуванням результатів, історією примірок користувача та персональним AI-стилістом. Також важливо додати механізми захисту персональних зображень, оскільки фото користувача є чутливими даними і повинні оброблятися з дотриманням принципів мінімізації зберігання.

4.4 Практична придатність прототипу для інтеграції в електронну комерцію

Експериментальна перевірка прототипу має показати не лише факт запуску AI-модуля, а й готовність системи бути частиною комерційного web-сценарію. Для електронної комерції важливо, щоб користувач міг перейти від перегляду товару до примірки без розриву логіки інтерфейсу. Якщо примірка виглядає як окремий технічний експеримент, вона не підсилює продаж, а навпаки відволікає від основного сценарію вибору товару.

Практична придатність оцінювалася за трьома групами ознак: завершеність користувацького маршруту, технічна стабільність обробки та корисність отриманого результату. До першої групи належать відкриття каталогу, сторінки товару, робота кошика та запуск примірки з конкретного товару. До другої групи належать коректні API-відповіді, збереження статусів і відсутність критичних збоїв під час передачі файлів. До третьої групи належить візуальна оцінка того, чи допомагає результат зрозуміти посадку й зовнішній вигляд виробу.

Окремим критерієм є сталість результату для однакових або близьких вхідних умов. Якщо система щоразу суттєво змінює форму товару, вона не може використовуватися як надійний інструмент підтримки купівлі. Тому для подальших експериментів доцільно фіксувати параметри генерації, порівнювати результати для різних типів фото та окремо оцінювати збереження деталей одягу, положення рукавів, нижньої межі виробу й природність областей біля шиї та плечей.

За результатами аналізу можна зробити висновок, що розроблений прототип є придатним як основа для інтеграції в демонстраційний онлайн-магазин. Водночас для промислового використання потрібно посилити модуль перевірки фотографій, додати автоматичне очищення тимчасових файлів, розширити набір тестових зображень і відокремити обробку

персональних фото від загальної логіки каталогу. Це дозволить перейти від дослідницького прототипу до сервісу, який можна безпечно використовувати в реальному середовищі.

Таким чином, експериментальна частина підтверджує, що цінність системи визначається не лише якістю окремої моделі IDM-VTON, а й узгодженістю всього програмного конвеєра: інтерфейсу, backend-обробки, підготовки файлів, генеративного модуля та механізму повернення результату користувачеві. Саме така цілісність є головною умовою подальшого розвитку системи в напрямі повноцінного інтелектуального сервісу для онлайн-продажу одягу.

ВИСНОВКИ

У ході виконання кваліфікаційної роботи було розроблено інтелектуальну систему віртуальної примірки одягу, що поєднує засоби web-розробки, методи комп'ютерного зору, геометричні перетворення та генеративне моделювання. Розроблена система демонструє повний сценарій взаємодії користувача з онлайн-магазином: перегляд головної сторінки, перехід до каталогу, відкриття сторінки товару, додавання товару до кошика та запуск AI-примірки.

У першому розділі було проаналізовано предметну область віртуальної примірки одягу. Показано, що ця задача є комплексною, оскільки потребує не лише накладання зображення товару на фотографію, а й урахування пози тіла, пропорцій фігури, сегментації, перекриттів, освітлення, текстури тканини та фотореалістичності результату.

У другому розділі сформовано математичне та алгоритмічне забезпечення системи. Було описано модель представлення зображень, ключових точок і масок, наведено формули афінного перетворення, TPS-викривлення, alpha blending та інтегральної оцінки якості результату. Це дозволило обґрунтувати вибір гібридного підходу, у якому геометричний контроль поєднується з генеративним покращенням.

У третьому розділі спроєктовано та описано програмну реалізацію системи. Backend реалізовано на FastAPI, frontend – на Next.js і React, а AI-модуль інтегровано через Gradio API з моделлю IDM-VTON. Описано структуру проєкту, API-ендпоінти, модель даних, статуси асинхронних завдань і механізми збереження результатів.

У четвертому розділі проведено експериментальну перевірку. Було підтверджено працездатність основних сторінок інтерфейсу, коректність комерційного сценарію та можливість запуску процесу примірки. Визначено основні обмеження системи: залежність від якості вхідного фото,

складність обробки перекриттів, потреба у стабільному AI-сервісі та необхідність додаткових механізмів захисту персональних зображень.

Практичне значення отриманих результатів полягає у можливості використання системи як основи для інтеграції модуля віртуальної примірки в електронні магазини одягу, fashion-маркетплейси та мобільні застосунки. Подальший розвиток роботи може включати підтримку більшої кількості категорій одягу, локальне розгортання генеративних моделей, автоматичну перевірку придатності фотографії, історію примірок і персоналізовані рекомендації стилю.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Han X., Wu Z., Wu Z., Yu R., Davis L. S. VITON: An Image-based Virtual Try-on Network. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018. URL: https://arxiv.org/abs/1711.08447 (дата звернення: 19.06.2026).

2. Wang B., Zheng H., Liang X., Chen Y., Lin L., Yang M. Toward Characteristic-Preserving Image-based Virtual Try-On Network. Proceedings of the European Conference on Computer Vision. 2018. URL: https://arxiv.org/abs/1807.07688 (дата звернення: 19.06.2026).

3. Choi S., Park S., Lee M., Choo J. VITON-HD: High-Resolution Virtual Try-On via Misalignment-Aware Normalization. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021. URL: https://arxiv.org/abs/2103.16874 (дата звернення: 19.06.2026).

4. Choi Y., Kwak S., Lee K., Choi H., Shin J. Improving Diffusion Models for Authentic Virtual Try-on in the Wild. Proceedings of the European Conference on Computer Vision. 2024. URL: https://arxiv.org/abs/2403.05139 (дата звернення: 19.06.2026).

5. Bazarevsky V., Grishchenko I., Raveendran K., Zhu T., Zhang F., Grundmann M. BlazePose: On-device Real-time Body Pose Tracking. arXiv. 2020. URL: https://arxiv.org/abs/2006.10204 (дата звернення: 19.06.2026).

6. Cao Z., Simon T., Wei S.-E., Sheikh Y. Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017. URL: https://openaccess.thecvf.com/content_cvpr_2017/html/Cao_Realtime_Multi-Person_2D_CVPR_2017_paper.html (дата звернення: 19.06.2026).

7. Güler R. A., Neverova N., Kokkinos I. DensePose: Dense Human Pose Estimation in the Wild. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018. URL: https://arxiv.org/abs/1802.00434 (дата звернення: 19.06.2026).

8. Kirillov A., Mintun E., Ravi N. et al. Segment Anything. Proceedings of the IEEE/CVF International Conference on Computer Vision. 2023. URL: https://arxiv.org/abs/2304.02643 (дата звернення: 19.06.2026).

9. Rombach R., Blattmann A., Lorenz D., Esser P., Ommer B. High-Resolution Image Synthesis with Latent Diffusion Models. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022. URL: https://arxiv.org/abs/2112.10752 (дата звернення: 19.06.2026).

10. Ronneberger O., Fischer P., Brox T. U-Net: Convolutional Networks for Biomedical Image Segmentation. Medical Image Computing and Computer-Assisted Intervention. 2015. URL: https://arxiv.org/abs/1505.04597 (дата звернення: 19.06.2026).

11. Jaderberg M., Simonyan K., Zisserman A., Kavukcuoglu K. Spatial Transformer Networks. Advances in Neural Information Processing Systems. 2015. URL: https://arxiv.org/abs/1506.02025 (дата звернення: 19.06.2026).

12. Bookstein F. L. Principal Warps: Thin-Plate Splines and the Decomposition of Deformations. IEEE Transactions on Pattern Analysis and Machine Intelligence. 1989. Vol. 11, No. 6. P. 567–585.

13. Goodfellow I., Pouget-Abadie J., Mirza M. et al. Generative Adversarial Nets. Advances in Neural Information Processing Systems. 2014. URL: https://arxiv.org/abs/1406.2661 (дата звернення: 19.06.2026).

14. Ho J., Jain A., Abbeel P. Denoising Diffusion Probabilistic Models. Advances in Neural Information Processing Systems. 2020. URL: https://arxiv.org/abs/2006.11239 (дата звернення: 19.06.2026).

15. Vaswani A., Shazeer N., Parmar N. et al. Attention Is All You Need. Advances in Neural Information Processing Systems. 2017. URL: https://arxiv.org/abs/1706.03762 (дата звернення: 19.06.2026).

16. FastAPI Documentation. URL: https://fastapi.tiangolo.com/ (дата звернення: 19.06.2026).

17. Next.js Documentation. URL: https://nextjs.org/docs (дата звернення: 19.06.2026).

18. React Documentation. URL: https://react.dev/ (дата звернення: 19.06.2026).

19. SQLAlchemy Documentation. URL: https://docs.sqlalchemy.org/ (дата звернення: 19.06.2026).

20. Gradio Client Documentation. URL: https://www.gradio.app/docs/python-client/introduction (дата звернення: 19.06.2026).

21. OpenCV Documentation. URL: https://docs.opencv.org/ (дата звернення: 19.06.2026).

22. Pillow Documentation. URL: https://pillow.readthedocs.io/ (дата звернення: 19.06.2026).

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

24. ДСТУ 8302:2015. Бібліографічне посилання. Загальні положення та правила складання. Київ : ДП «УкрНДНЦ», 2016. 20 с.

25. nickaq/ai_virtual_tryon : репозиторій дипломного проєкту. GitHub. URL: https://github.com/nickaq/ai_virtual_tryon (дата звернення: 19.06.2026).

26. yisol/IDM-VTON : Hugging Face Space. Hugging Face. URL: https://huggingface.co/spaces/yisol/IDM-VTON (дата звернення: 19.06.2026).

27. Бодянський Є. В., Руденко О. Г. Штучні нейронні мережі: архітектури, навчання, застосування. Харків : ТЕЛЕТЕХ, 2004. 369 с.

28. Szeliski R. Computer Vision: Algorithms and Applications. 2nd ed. Cham : Springer, 2022. 925 p.

29. Gonzalez R. C., Woods R. E. Digital Image Processing. 4th ed. New York : Pearson, 2018. 1168 p.