

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ
КАФЕДРА КОМП'ЮТЕРНИХ НАУК

КВАЛІФІКАЦІЙНА МАГІСТЕРСЬКА РОБОТА

на тему:

**«Інформаційна технологія 3D-реконструкції
обличчя людини»**

**Завідувач
випускаючої кафедри**

Довбиш А.С.

Керівник роботи

Шелехов І.В.

Студента групи ІН.м – 81н

Шумило Д.О.

СУМИ 2020

Сумський державний університет

(назва вузу)

Факультет ЕЛІП Кафедра Комп'ютерних наук
Спеціальність «Інформатика»

Затверджую:

зав.кафедрою _____

“ _____ ” _____ 20__ р.

ЗАВДАННЯ НА ДИПЛОМНИЙ ПРОЕКТ (РОБОТУ) СТУДЕНТОВІ

Шумилу Дмитру Олександровичу

(прізвище, ім'я, по батькові)

1. Тема проекту (роботи) Інформаційна технологія 3D-реконструкції обличчя людини

затверджую наказом по інституту від “ _____ ” _____ 20__ р. № _____

2. Термін здачі студентом закінченого проекту (роботи) _____

3. Вхідні данні до проекту (роботи) фотографії облич у вільному доступі в мережі Інтернет.

4. Зміст розрахунково-пояснювальної записки (перелік питань, що їх належить розробити) Огляд параметрів сцени, реконструкції поверхні, методів реконструкції обличчя, застосування. Постановка задачі дослідження. Вибір методів розв'язання задачі. Інформаційне та програмне забезпечення системи 3D-реконструкції обличчя: система збору статистичних даних, система генерації тестів та проведення експериментів, реалізація запропонованого підходу, тестування системи.

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

12 рисунків для опису роботи алгоритмів, методів, та опису результатів, серед них схематичні; 12 скріншотів результатів виконання програмного коду (графіки), 1 таблиця.

6. Консультанти до проекту (роботи), із значенням розділів проекту, що стосується їх

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв
<i>1.3 Постановка задачі</i>	<i>Шелехов І.В.</i>		

7. Дата видачі завдання 02.03.2020

Керівник

_____ (підпис)

Завдання прийняв до виконання

_____ (підпис)

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Назва етапів дипломного проекту (роботи)	Термін виконання проекту (роботи)	Примітка
1.	<i>Пошук та вибір літератури для виконання дипломного проекту.</i>	<i>02.03.2020 – 13.03.2020</i>	
2.	<i>Написання першого розділу дипломного проекту: Огляд існуючих рішень.</i>	<i>16.03.2020 – 23.03.2020</i>	
3.	<i>Написання другого розділу дипломного проекту: Вибір методу розв'язання задачі.</i>	<i>24.03.2020 – 03.04.2020</i>	
4.	<i>Розробка та програмна реалізація інформаційної технології.</i>	<i>06.04.2020 – 15.04.2020</i>	
5.	<i>Написання третього розділу дипломного проекту: Інформаційне та програмне забезпечення системи.</i>	<i>16.04.2020 – 01.05.2020</i>	
	<i>Написання розділів Вступ, Висновки, Додатки.</i>	<i>04.05.2020 – 08.05.2020</i>	

Студент – дипломник

_____ (підпис)

Керівник проекту

_____ (підпис)

РЕФЕРАТ

Записка: 64 стор., 24 рис., 1 табл., 2 додатка, 60 джерел.

Об'єкт дослідження – процес реконструкції 3D-поверхні з 2D-зображення.

Мета роботи – інформаційний аналіз та синтез системи 3D-реконструкції обличчя людини.

Методи дослідження – метод Левенберга-Марквардта.

Результати – виконано огляд існуючих способів реконструкції поверхні, розроблено, програмно реалізовано, та протестовано інформаційну технологію 3D-реконструкції обличчя людини за 2D-зображенням. Мовою C++ з використанням бібліотек OpenGL та OpenCV розроблено систему збору статистичних даних, необхідних для розрахунку коваріаційної матриці, що лежить в основі запропонованого підходу. Підхід разом із системою проведення експериментів розроблено мовою Matlab.

3D-РЕКОНСТРУКЦІЯ ОБЛИЧЧЯ, МОРФОВАНА МОДЕЛЬ,
ЗВОРОТНА КОВАРІАЦІЙНА МАТРИЦЯ, КЛЮЧОВІ ТОЧКИ,
ФОТОГРАФІЯ.

ЗМІСТ

ВСТУП	6
1 ОГЛЯД ІСНУЮЧИХ РІШЕНЬ	9
1.1 Параметри сцени	9
1.2 Реконструкція поверхні	16
1.3 Методології реконструкції обличчя	21
1.4 Застосування	31
1.5 Постановка задачі	32
2 ВИБІР МЕТОДУ РОЗВ'ЯЗАННЯ ЗАДАЧІ	33
2.1 Початкове наближення камери	33
2.2 Алгоритм Левенберга-Марквардта	34
2.3 Морфована модель	35
2.4 Алгоритм 3D-реконструкції	36
3 ІНФОРМАЦІЙНЕ ТА ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ СИСТЕМИ	37
3.1 Система збору статистичних даних	37
3.2 Система генерації тестів та проведення експериментів	43
3.3 Реалізація запропонованого підходу	44
3.4 Тестування системи	44
ВИСНОВКИ	52
СПИСОК ЛІТЕРАТУРИ	53
ДОДАТОК А	59
ДОДАТОК Б	61

ВСТУП

3D-реконструкція обличчя - це процес створення 3D-моделі людського обличчя з 2D або 3D (2D зображення + зображення глибини(лазерні дані)) зображення, або пари зображень. 3D-реконструкція обличчя застосовується у багатьох галузях, таких як моделювання пластичної хірургії, розпізнавання обличчя та його виразів, симуляція віртуальної реальності, 3D-ігри та анімація. Таким чином, точна реконструкція 3D-моделі поверхні обличчя з 2D-зображень все ще є популярною темою для дослідження.

У сучасному світі графіка в комп'ютерних іграх і кіноіндустрії продовжує вдосконалюватися з кожним роком. Вже тривалий час простежується тенденція зйомки справжніх акторів для відтворення їх у 3D-моделях комп'ютерних персонажів. Для того, щоб створити повнометражну комп'ютерну анімацію, потрібна участь високо кваліфікованих художників, а також велика кількість часових і обчислювальних ресурсів. Тому розвиток технологій захоплення рухів актора, побудови тривимірних моделей сцени і об'єктів на ній, розпізнавання облич і основних рис обличчя, поступово призводить до автоматизації процесу створення тривимірних моделей акторів і підвищенню їх реалістичності.

Отримання тривимірної моделі обличчя зазвичай вимагає наявності спеціального пристрою, такого як Kinect або лазерний сканер. Але такі прилади складно використовувати поза спеціальними лабораторіями, через складнощі в облаштуванні необхідного оточення, та їхню високу ціну. У більш простому випадку реконструювати 3D-модель обличчя намагаються за однією або декількома фотографіями. При такому підході, як правило, використовують бази шаблонних моделей для пошуку максимально близьких до вихідного зображення ключових областей.

Більшість обмежень в автоматичних методах синтезу облич існують щоб уникнути нереалістичних, несхожих на натуральне обличчя результатів.

Наприклад, деякі методи вимагають внесення ручних правок у ході процесу генерації тривимірної моделі, або працюють тільки для фронтального зображення обличчя. З появою морфованої моделі [40] (3D morphable model, 3DMM) почали активно розвиватися нові напрямки автоматичної генерації, реконструкції та анімації облич. Основна ідея використання морфованої моделі полягає в розкладанні довільної поверхні обличчя в лінійну комбінацію базисних облич. За рахунок апроксимації довільної моделі базисними, зменшується необхідний розмір задалегідь підготовленої бази, а висока деталізація вихідних моделей дозволяє синтезувати реалістичні обличчя. Підхід лінійної апроксимації не вимагає високої обчислювальної потужності, тому застосовується навіть для генерації моделей в реальному часі [47] і дає реалістичні результати при побудові анімації облич.

Реконструкція тривимірної моделі обличчя за двовимірним зображенням складається з двох завдань: побудови тривимірної поверхні обличчя, та знаходження його положення у просторі. Відповідно до визначення параметричної моделі Парке [54] поверхня обличчя задається набором координат тривимірних вершин та множиною площин, що проходять через них. Положення тривимірної моделі в просторі описується матрицею повороту і вектором перенесення камери.

Задача визначення орієнтації і позиції відкаліброваної перспективної камери по n відомим 3D точкам в координатах моделі, і їхнім 2D проєкціям на зображенні (perspective- n -point problem, PnP) має розв'язок при $n \geq 3$ [4, 7]. За наявності інформації про більше, ніж 3 точки на зображенні та їхнє розташування в тривимірному просторі, глобально оптимальний розв'язок цієї задачі може бути знайдено алгоритмом OPnP [60] за обчислювальну складність, що дорівнює $O(n)$.

Мінімізація функції вартості у вигляді нелінійних найменших квадратів методом Левенберга-Марквардта, дозволяє зменшити помилку знайденої орієнтації і позиції камери OPnP алгоритмом. Для цього в якості вагової матриці

в методі найменших квадратів слід взяти зворотню коваріаційну матрицю розподілу 2D точок в 3D. Алгоритм Левенберга-Марквардта крім оптимізації параметрів камери водночас можна використовувати для наближення параметрів морфованої моделі, щоб отримати тривимірну поверхню обличчя.

Описані положення про обчислення початкового наближення параметрів перспективної камери за допомогою OPnP алгоритму і використання зворотної коваріаційної матриці в якості матриці ваг при мінімізації функції вартості у вигляді нелінійних найменших квадратів, лягли в основу інформаційної технології, розробленої у цій роботі.

1 ОГЛЯД ІСНУЮЧИХ РІШЕНЬ

Метою цього розділу є огляд існуючих способів реконструкції обличчя.

1.1 Параметри сцени

Зворотний процес реконструкції називається візуалізацією. Починаючи з 3D-моделі, що складається з поверхні та її текстури, та використовує набір параметрів сцени, що описують положення об'єкта, освітлення та проекцію камери, 2D-зображення може бути візуалізовано. Багато припущень зроблено щодо або обмеженого розуміння властивостей відбиття різних поверхонь, або частіше, щодо обчислювальної ефективності. Розглянемо деякі з них, необхідні для візуалізації зображення.

1.1.1 Модель об'єкта

Модель об'єкта складається як із зображення форми поверхні, так і текстури. Поверхні об'єкта суцільні, але зазвичай наближені до дискретних структур. Обмежена модель об'єкта - це поверхня виражена як карта глибини. Карти глибини представлені як зображення, що складаються з одного каналу, що містить відстань поверхні від конкретного ракурсу. Це спрощує деякі задачі, за рахунок відсутності потреби в проектуванні камери, та зберігання лише видимої інформації. Однак така модель обмежена, оскільки об'єкт містить лише інформацію з одного ракурсу або боку обличчя, а також обмежена відсутністю знань про сторони або підборіддя обличчя, які, як правило, закриті. Ці карти глибини іноді називають 2.5D-моделями, оскільки вони тільки містять часткову інформацію в третьому вимірі. Карти глибини часто використовуються спільно

із звичайним зображенням, де вони називаються, RGB-D для стандартних червоного, зеленого, та синього каналів з додаванням каналу глибини.

Справжня 3D-модель краще представлена у вигляді трикутної сітки, та здатна захоплювати деталі по всій поверхні. Сітка трикутника складається з вершин, ребер та граней трикутника. Вершини - це положення у тривимірному просторі, що можуть містити іншу інформацію, таку як текстура або вектор нормалі. Ребро - це з'єднання між двома вершинами. Грань - закритий набір з трьох ребер. Зазвичай грані ділять кожне ребро з однією іншою гранню, щоб утворити суцільну поверхню, однак можуть бути й межі, що утворюють отвори у поверхні. На рисунку 1.1 показано відмінності між 2.5D та 3D-моделями. Карта глибини має фіксовану орієнтацію, оскільки це просте зображення, однак під трикутним представленням сітки, об'єкт містить інформацію про всі частини обличчя, та може бути повернений та візуалізований під будь-яким ракурсом.

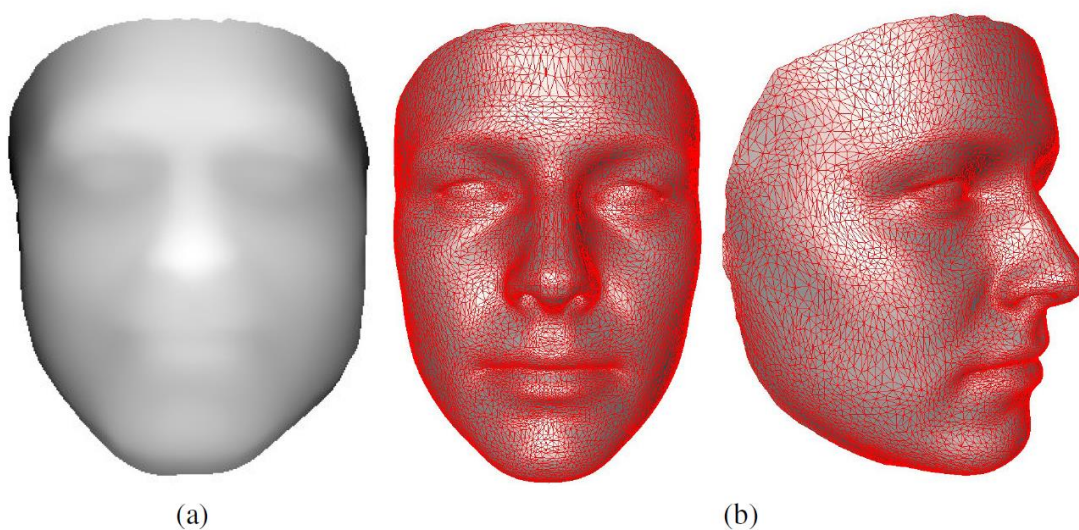


Рисунок 1.1 – Порівняння моделей об'єктів, карти глибини (a) та трикутної сітки (b)

1.1.2 Освітлення

Без освітлення всі сцени виглядали б суцільно чорними. Фотони світла випромінюються від джерел світла та взаємодіють з поверхнями в сцені, поки нарешті не дістаються або до людського ока, або до датчика камери, котрий формує кольори зображення залежно від кількості фотонів, що досягають датчика.

Взаємодія з поверхнями складна і не повністю зрозуміла. На деяких поверхнях світло просто відштовхується, на інших, таких як скло, світло прогинається під час проходження, туман, наприклад, може частково закривати світло, дозволяючи тільки невеликій його кількості проходити крізь нього. Шкіра людини це дуже складна поверхня, оскільки вона частково напівпрозора. Частина світла відбивається від поверхні, тоді як інша частина проходить крізь неї, що дозволяє нам бачити кров'яні вени під шкірою. Багато роботи існує для створення реалістичного відображення шкіри [10].

Оскільки освітлення складне, ми часто робимо спрощення припущень. Одна поширена модель для освітлення, це модель Ламбертіана, яка моделює розсіяне світло і виражається як,

$$I_d = k_d \Gamma \mathbf{n}. \quad (1.1)$$

Де I_d - дифузна інтенсивність, k_d - інтенсивність джерела світла, I - напрямок джерела світла від поверхні до джерела, \mathbf{n} - нормаль поверхні. Це припущення щодо освітлення не залежить від ракурсу, та добре застосовне до матових поверхонь. Однак поліровані поверхні демонструють змінну освітленість, що залежить від розташування спостерігачів. Поміркуємо про яскраву пляму на лисині людини, яка рухається при зміні позицій. Вона моделюється за допомогою моделі підсвічування Фонга, що включає компонент дзеркального відбиття,

$$I_s = k_s (\mathbf{r}^T \mathbf{v})^\alpha.$$

Де I_s - це дзеркальне освітлення, k_s - це дзеркальна константа, α - сила косинусного кута, яка визначає наскільки глянцевою є поверхня, \mathbf{v} - вектор виду від поверхні до точки виду, \mathbf{r} - відображення джерела світла, $\mathbf{r} = 2(\mathbf{l}^T \mathbf{n})\mathbf{n} - \mathbf{l}$. На Рисунку 1.2 показана візуалізація різних векторів, що використовуються в моделях освітлення, де \mathbf{l} - напрямок джерела світла, \mathbf{n} - поверхня нормалі, \mathbf{v} - напрямок точки зору, \mathbf{r} - відбиття джерела світла.

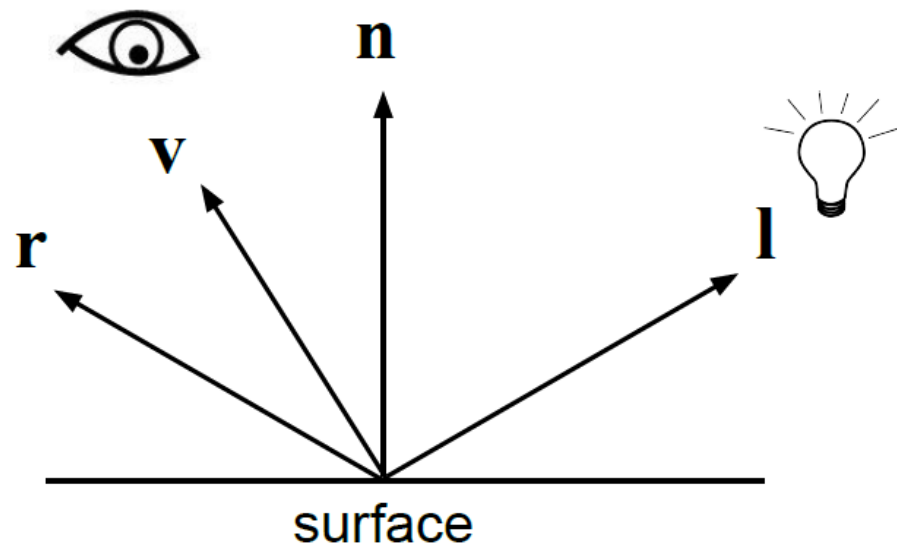


Рисунок 1.2 – Одиничні вектори, що використовуються в моделях освітлення

1.1.3 Модель камери

Щоб зрозуміти образ, нам не тільки потрібно моделювати як світло взаємодіє з поверхнею, але й також – як камера знімає світло в сцені. Модель камери описує як спроектувати 3D-точки в реальному світі на 2D-піксельне розташування на зображенні. З точки зору камери – як фотон світла, що

рухається з поверхні, досягає певного елемента на датчику CCD. Найпоширенішою моделлю є модель пінхол-камери, яка передбачає, що площина зондування розташована позаду бар'єра з єдиною точкою, що дозволяє пропускати світло. З цією моделлю проекція пояснюється матрицею проекції 3x4 [16] P , до масштабу, яка може бути розкладена на внутрішню і зовнішню матриці.

$$P = \underbrace{\begin{bmatrix} f_x & s & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}}_K \times \left[\begin{array}{ccc|c} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{array} \right]$$

Внутрішня матриця K описує властивості пінхола та площини зображення: фокусна відстань (f_x, f_y) , головна точка (c_x, c_y) та скос s . Зовнішня матриця описує положення камери у системі координат: обертання R та переклад T . Зазвичай подані додаткові припущення, $f_x = f_y$, $s = 0$ та (c_x, c_y) є центром зображення, щоб залишити лише 8 параметрів, включаючи масштаб. Це рівняння моделі проекції також відоме як перспективна проекція.

Модель перспективної проекції - це спрощення типових камер, які мають лінзи, що спотворюють світло з більш широкого діапазону, ніж пінхол. У цих випадках радіальне викривлення має розглядатися особливо для фотографій високої роздільної здатності або ширококутних об'єктивів. Однак як правило, спотворення буде усунене до використання зображень у будь-якому пайплайні шляхом перекомпонування зображення з відомими параметрами.

Простіша модель камери використовує слабку перспективну проекцію. У моделі слабкої перспективної проекції, всі промені від площини датчика паралельні, миттєво видаляють перекос, фокусну відстань та головну точку від розгляду. Отже, слабка перспективна модель це лише перші 2 рядки зовнішньої матриці, та має вона 6 параметрів, включаючи шкалу. Порівняння двох вище описаних моделей наведено на Рисунку 1.3. Слабка перспективна проекція

працює, коли об'єкт є відносно віддаленим від камери, порівняно з фокусною відстанню та розміром об'єкта. Щодо облич, зображення, зроблені на відстані більшій за довжину руки, демонструють незначні спотворення в перспективі.

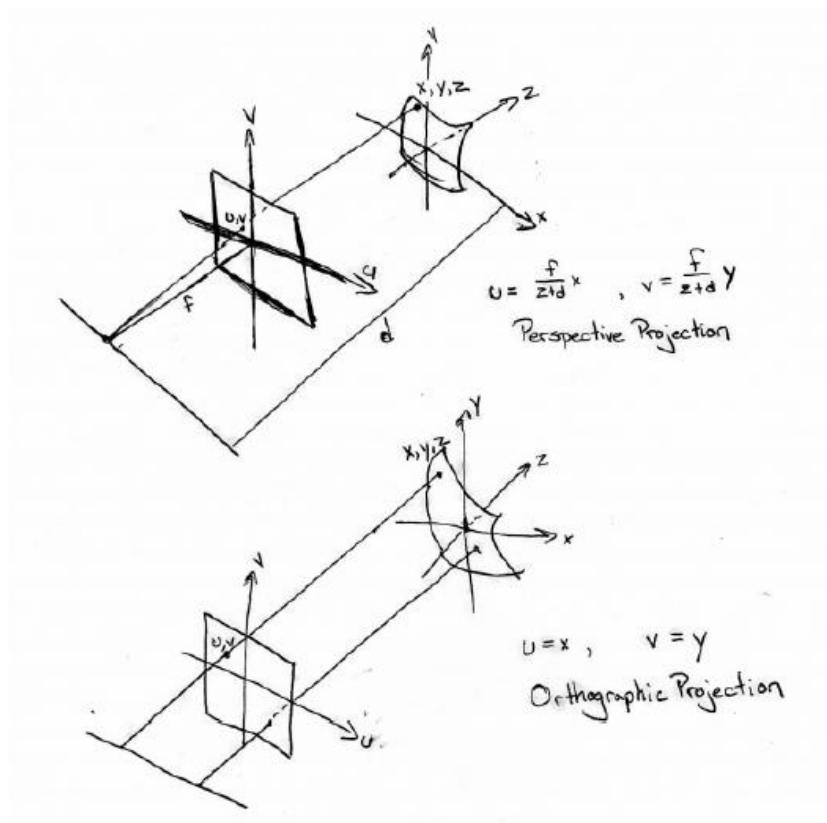


Рисунок 1.3 – Порівняння перспективних та слабких перспективних проєкцій

У перспективній проєкції внутрішні параметри камери та відстань до об'єкта впливають на розташування точки на площині зображення. При слабкій перспективній проєкції точки уздовж ортогональних ліній до зображення проявляються в одному місці незалежно від глибини.

1.1.4 Знаходження перспективної камери

Більшість алгоритмів для знаходження орієнтації і положення камери засновані на відповідності точок або ліній. Стандартний підхід до калібрування

камери формулюється для 2D-3D відповідності точок [46]. Для вирішення цієї задачі необхідні дані щонайменше про 3 точки [43]. Спочатку алгоритми розв'язували задачу для фіксованої кількості точок від 3 до 5, але зі збільшенням їхньої кількості зростала точність відновлення камери. Першим $O(n)$ алгоритмом, що міг розв'язати задачу для 5 і більше точок, був EPnP [49]. Основна ідея полягала в тому, щоб представити 3D координати як лінійну комбінацію 4 контрольних точок, що приймалися за невідомі незалежно від загальної кількості вихідних точок. Ці контрольні точки були отримані за допомогою простих технік лінеаризації. Лінеаризація була послідовно замінена поліноміальними рішеннями в Robust PnP [50], та далі в алгоритмі OPnP (Optimal Pose from n Points) [59], який дає в загальному випадку найбільш точні результати. У OPnP для повороту використовується параметризація кватерніонами, що дозволяє оцінювати позицію камери за допомогою рішення базису Грьобнера.

Ітеративні підходи формулюють знаходження параметрів камери як задачу мінімізації в формі нелінійних найменших квадратів і мінімізують геометричну або алгебраїчну помилку. Одним з перших таких методів був алгоритм ортогональної ітерації (Orthogonal Iteration Algorithm, OI) [52]. Він на відміну від багатьох інших алгоритмів, мінімізує помилку, описану в 3D координатах моделі, а не намагається стандартно мінімізувати 2D помилку перспективного перетворення [45]. На практиці алгоритм сходиться швидко, але може зупинитися в невідповідному локальному мінімумі, якщо був некоректно ініціалізований. Основний недолік алгоритму ортогональної ітерації в тому, що через ітеративність він працює набагато повільніше за аналоги та показує менш стабільні результати.

1.2 Реконструкція поверхні

Реконструкція поверхні - це складний процес, який суттєво відрізняється залежно від типу інформації на вході (зображення, хмара точок тощо), якості даних (шум, зовнішній вигляд тощо), бажаної інформації на виході (сітка, скелет, об'єм тощо) та типу форми (рукотворна, органічна тощо). Для графічної спільноти, реконструкція поверхні майже завжди стосується пристосування форми до хмари точок, потенційно з додатковою сторонньою інформацією, як, наприклад, поверхневі нормалі. Більшість алгоритмів реконструкції приймають хмари точок як вхідну інформацію, включаючи методи з гладкістю поверхні (наприклад, дотичні площини [20], такі що рухаються найменшими квадратами [2], функція радіальної основи [12], реконструкція поверхні Пуассона [25]), замість методів, що базуються на видимості [13], керовані даними [29] тощо.

Одним з найбільш використовуваних методів є метод реконструкції поверхні Пуассона [25], що значною мірою пояснюється його ефективністю та надійністю. Цей метод оцінює об'ємну нормальну область на основі хмари точок і будує 3D-рівняння Пуассона, подібне 2D-рівнянню Пуассона, отриманого у результаті фотометричного стерео. Такий підхід є надійний проти шуму і дозволяє отримати глобальне рішення на відміну від багатьох евристичних підходів.

Однак у комп'ютерному баченні реконструкція поверхні зазвичай стосується отримання тривимірного розуміння об'єкта з двовимірних зображень. Деякі методи розглядатимуть RGB-D вхід, який може розглядатися як хмара точок з інформацією про текстуру. У конкретному випадку, коли RGB-D дані це відео, задача реконструкції називається динамічним синтезом [28]. Ціль в цьому сценарії може бути краще описана як розрахування найбільш вірогідної 3D-форми, що описує зображення набором передбачуваних матеріалів, точок огляду та умов освітлення. Як правило, з різними наборами припущень можливі дуже деталізовані та точні реконструкції. Багато різних підказок, таких як фокус

зображення, текстура, затінення та стерео співвідношення дають можливість попередньо зрозуміти форму об'єкта. Ці дві останні підказки продемонстрували надійність для реконструкції і, як правило, є найбільш поширеними підходами.

1.2.1 Багаторакурсне стерео

Багаторакурсне стерео бере колекцію фотографій, отриману під різними точками огляду, і намагається реконструювати правдоподібну 3D-геометрію, яка пояснює зображення набором обґрунтованих припущень, передусім - жорсткість об'єкта. Огляд процедури виглядає наступним чином. По-перше, оцінюються параметри камери, як зовнішні (положення і орієнтація), так і внутрішні (фокусна відстань і масштаб датчика). Зазвичай це робиться за допомогою структури «від руху» [16] для обчислення моделі камери з колекції фотографій офлайн. Як правило, для повного опису того як 3D крапка на об'єкті розташована на 2D-піксельній локації на зображенні, використовується модель пінхол-камери. По-друге, визначаються спільні точки між різними зображеннями. З відомими моделями камер будь-яка пара точок, знайдена в обох камерах однозначно визначає точку об'єкта в тривимірному просторі (Рисунок 1.4). Нарешті, тривимірні розташування всіх спільних точок утворюють хмару точок, яку потім можна розв'язати за допомогою традиційних методів реконструкції графічної поверхні.

Якщо ліва камера O_L спостерігає за точкою X_L , та існує інша камера O_R з відомою відносною орієнтацією, тоді точка повинна падати по епіполлярній лінії у правий вид. Якщо відповідна точка визначена у правому виді, X_R , тоді положення цієї точки однозначно визначено у тривимірному просторі, X .

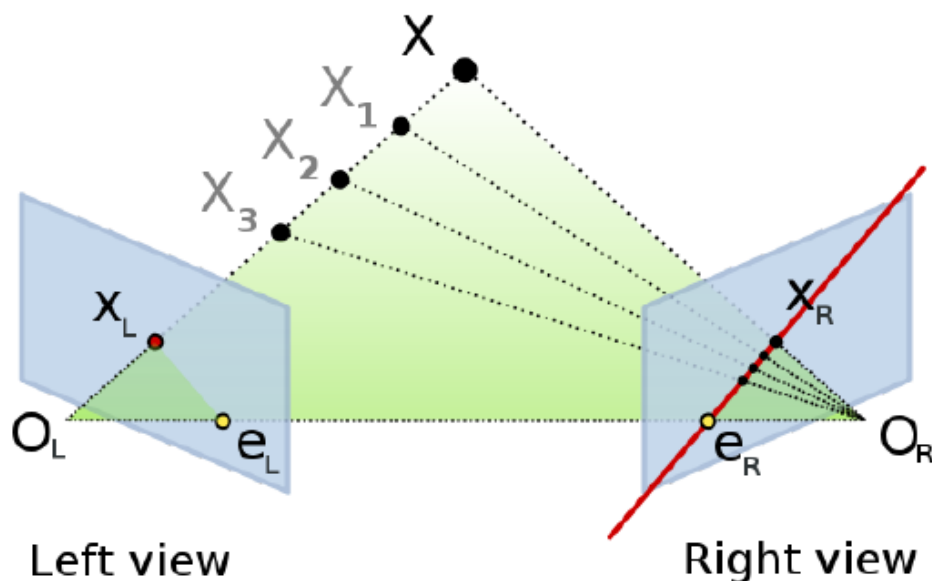


Рисунок 1.4 – Епіполярна геометрія, основа для багатовидової стереореко́нструкції

1.2.2 Фотометричне стерео

У той час як багаторакурсне стерео використовує одночасні зображення, зроблені в різних точках огляду, фотометричне стерео використовує послідовні зображення, зняті з однієї точки зору за різних умов освітлення (Рисунок 1.5). Здійснюючи обґрунтовані припущення щодо властивостей матеріалу та світла, можуть бути визначені нормалі поверхні. Інтуїтивно, частина об'єкта, повернута до світла, буде здаватися яскравішою, ніж частина, повернута подалі від світла, тому можна визначити кут між нормальною поверхнею та напрямком джерела світла. Необхідно кілька різних напрямків освітлення щоб вирішити неоднозначність щодо справжньої орієнтації поверхні, та переконатися, що кожна частина об'єкта повернута до світла в одному зображенні. Нормальна область будується за допомогою піксельної сітки зображення, а форму визначають шляхом інтеграції нормальної області. Форма-від-Затінення створює зображення карти глибини об'єкта, оскільки всі зображення були зроблені з однакового ракурсу.

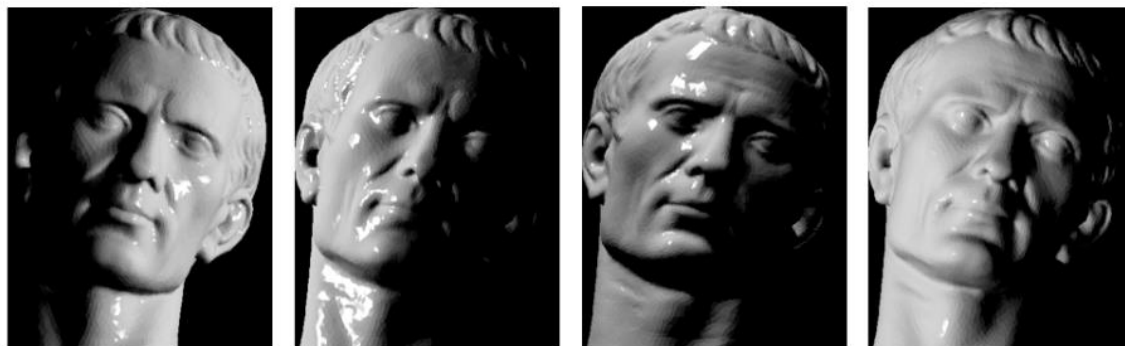


Рисунок 1.5 – Пробні зображення, використані для фотометричного стерео

Найпростіші припущення, це слабка перспективна модель проекції камери з моделлю освітлення Ламбертіана. У цих умовах плідна робота з фотометричним стерео [35] демонструє можливість реконструювати поверхню з трьох зображень із відомими умовами освітлення. Навіть поточні методи все ще використовують відомі умови освітлення для кооперативних предметів [15].

Пізніше було виявлено, що навіть без знань про джерела світла фотометрична стереосистема може скористатися природою низького рангу припущення про освітлення [17, 38, 27, 4, 5, 36] .. Сферичні гармоніки - це повний набір ортогональних функцій на поверхні кулі (Рисунок 1.6), запропоновані щоб допомогти вирішити фотометричне стерео. Замість використання функції двонаправленого розподілу відбиття щоб змоделювати світло, можуть бути використані сферичні гармоніки. Для простої поверхні Ламбертіана (Рівняння 1.1), сферичні гармоніки 1-го порядку прямо відповідають значеннями 1 , n_x , n_y та n_z . Модель підсвічування Фонга прямо не переміщується, але сферичні гармоніки 2-го порядку можуть моделювати 99.2 % енергії освітлення.

Червоний – додатний, синій - від'ємний з інтенсивністю, що вказує на величину функції.

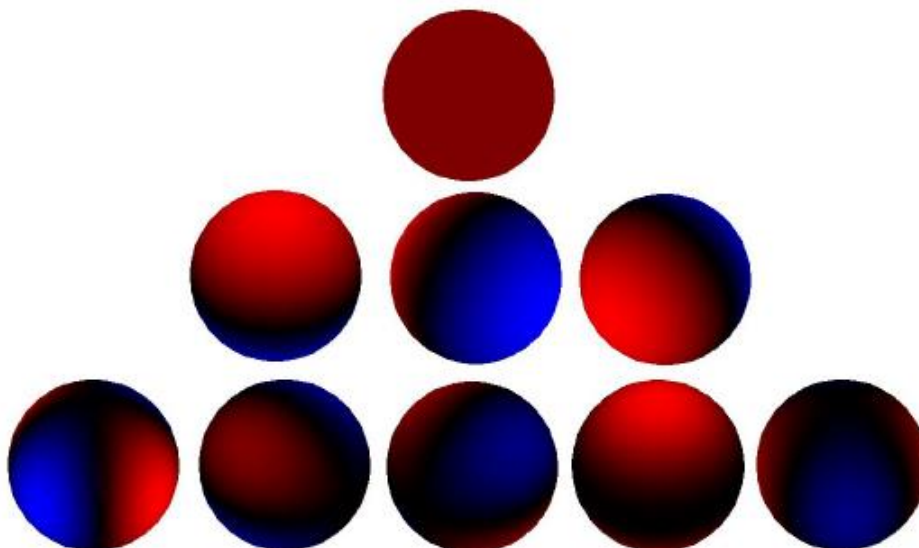


Рисунок 1.6 – Візуальне зображення сферичних гармонік нульового, першого та другого порядків

Перегляд проблеми зі сферичними гармоніками призводить до творчого вирішення через сингулярне розкладання значення (Singular value decomposition, SVD). Візьмемо матрицю, де кожен рядок є зображенням, а кожен стовпець – відповідний піксель серед усіх зображень, тоді SVD-рейтинг 4, це надійний спосіб одночасної оцінки нормалей поверхні та умов освітлення з припущенням Ламбертіана. Розкладання 9 рангу може впоратися зі складнішими ситуаціями освітлення. Після SVD, обмеження інтегральності або попереднє знання об'єкта необхідні для вирішення неоднозначності розкладання. Такі підходи вимагають достатньої кількості зображень для отримання точної реконструкції, особливо для нетвердих об'єктів, таких як обличчя, позаяк зміна емоцій може порушити припущення про низький ранг.

Важливо також зазначити, що ці фотометричні стерео-методи реконструюють із спільних ракурсів, і таким чином створюють 2.5D-поверхню замість справжньої 3D-поверхні, такої як багаторакурсне стерео. Існує декілька робіт, що поєднують багаторакурсне стерео та фотометричне, щоб створювати точніші 3D-поверхні [19, 31], навіть оцінюючи довільні нелінійні карти відгуку

камери. Ці методи реконструкції є загальними і працюють для будь-якого довільного об'єкта, що задовольняє припущенням щодо матеріалу та освітлення.

1.3 Методології реконструкції обличчя

Існують численні методи реконструкції обличчя, які залежать від рівня контролю над захоплюючим середовищем, та можуть створювати реконструкції з різними рівнями якості, придатними для різних завдань. У таблиці 1.1 наведено огляд найпоширеніших сценаріїв та підходів. Рисунок 1.7 наводить приклади рівнів деталізації, що різні підходи до реконструкції можуть створювати. У решті цього підрозділу наводиться короткий огляд різних сценаріїв, їхні найбільш поширені підходи до реалізації, та деякі їхні шляхи застосування.

Таблиця 1.1 Огляд підходів для реконструкції обличчя.

	Вхід	Підхід	Деталь
Обмежений	Хмара точок	3D-сканер	$\pm 0,03\text{мм}$
	Синхронізовані зображення	Багаторакурсне стерео [9]	0,088мм, пори
	Time-multiplexed	Фотометричне стерео [32]	зморшки
Необмежений	Одне зображення	3DMM [13]	гладкість
		CNN [48]	гладкість
	Відео	Оптичне відстеження потоку [51]	зморшки
	RGB-D Video	Динамічний синтез [61]	зморшки
	Колекція фото	Фотометричне стерео [53]	зморшки

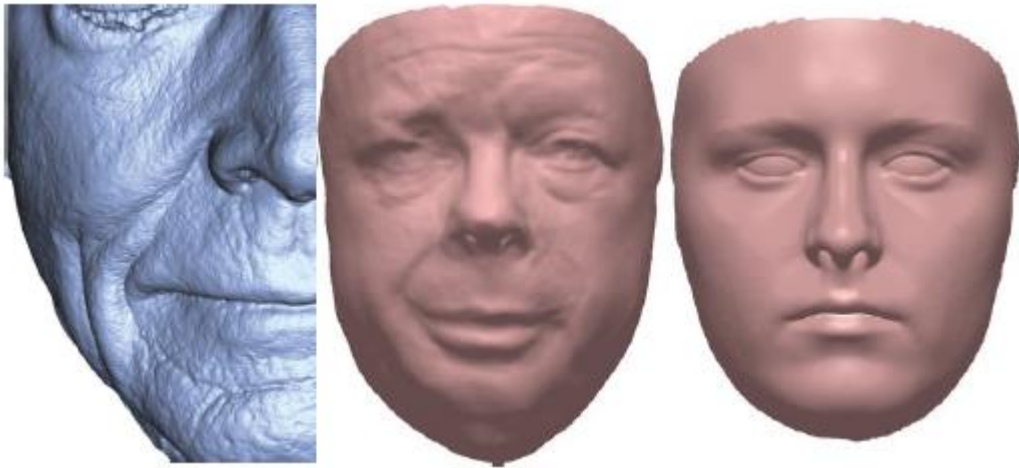


Рисунок 1.7 – Приклад реконструкцій на різних рівнях деталізації.

Зліва направо: пори, зморшки, гладкість.

1.3.1 Обмежена

Найбільш точний сценарій реконструкції обличчя знаходиться в обмеженому середовищі. Обмежений означає, що існує спільне середовище для зйомки зображень. Іншими словами, предмет, що цікавить, може бути розміщений у відомій позі та виразі, умови освітлення можуть бути повністю відомими, або розміщення та калібрування камери можуть бути відомі. У обмеженому сценарії, припущення можуть бути зроблені про процес реконструкції, який дозволяє реконструювати високо деталізовані та точні моделі.

1.3.1.1 3D-сканер

3D-сканер (Range scanner) - це спеціальне обладнання, призначене для прямого захоплення хмари точок. Існують різноманітні підходи, включаючи проектування структурованого інфрачервоного малюнка світла на поверхню і перегляд його деформації, або використання лазера для або вимірювання часу

польоту, або триангуляції. Незалежно від підходу, 3D-сканер виконує обробку всередині обладнання, та повертає або хмару точок, або карту глибини сцени.

За наявності великої суми грошей, Konica Minolta Vivid 9i може захопити деталі з точністю $\pm 0,03$ мм на широкому діапазоні глибин. Система придатна для виготовлення процесів та перевертає інженерні частини. Якщо деталь занадто велика, сканер має програмне забезпечення для автоматичного зшивання разом кількох сканувань, щоб створити один об'єкт. Щоб зафіксувати всю цілісність об'єкта, він включає в себе обертову платформу. Часто використовується для оцінки підходів реконструкції або створення ймовірнісних моделей обличчя. Недоліком цього рівня точності є те, що час сканування займає 2 секунди на одиночне захоплення (Рисунок 1.8).

За більш розумною ціною, можна придбати пристрій PrimeSense, що працює під управлінням розважальної системи Microsoft Kinect. Цей пристрій фіксує з точністю $\pm 0,5$ мм, що може не працювати для виготовлення, але все одно захопить зморшки на обличчі людини. Йому бракує точності, однак цей пристрій може фіксувати 30 кадрів в секунду, що робить його застосовним для динамічного синтезу.

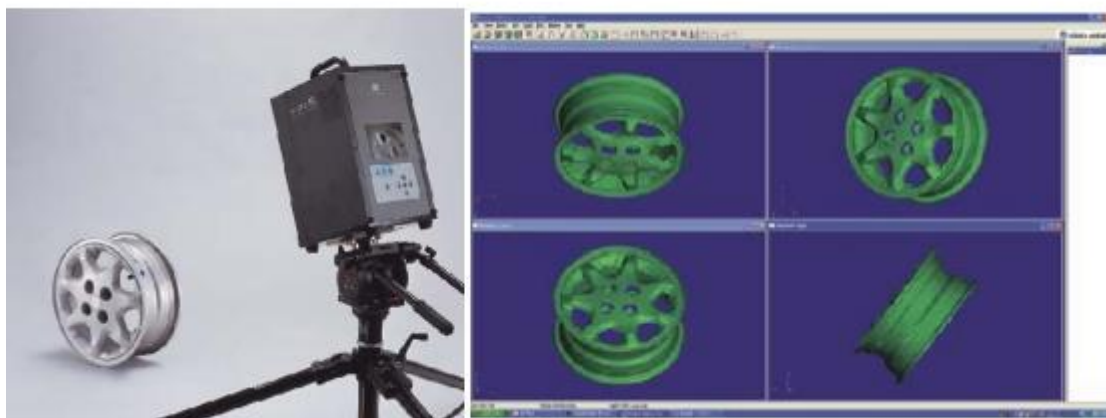


Рисунок 1.8 – Konica Minolta Vivid 9i - безконтактний 3D-сканер

1.3.1.2 Багаторакурсне стерео

Люди сприймають глибину частково, маючи два ока, що можуть триангулювати предмети у простір. Багаторакурсне стерео використовує цей самий принцип, розміщуючи дві або більше камер з відомими позиціями та внутрішніми матрицями камери. Коли в обох ракурсах визначена спільна точка, тривимірне положення однозначно визначається на основі перетину променів, що виходять з кожної камері та проходять через спільну точку. Білер та ін. презентували сучасні методи для реконструкція облич, на основі споживчих налаштувань камер [6] або стерео відео [7]. Їхня робота спершу калібрує камери на основі калібрувальної сфери, щоб визначити матриці камери та відносні положення в просторі. Після калібрування кадри фіксуються протягом 0,1 секунди один від одного з усіх камер, та багаторакурсний стерео підхід реконструює обличчя з деталями рівня пор і зморшок. Працюючи над пірамідою зображень від низької до високої роздільної здатності, здійснюється пошук епіполярних ліній між парами камер на основі нормованої поперечної кореляції, щоб виявити спільні точки між камерами. Далі спільні точки вдосконалюються на основі фотометричної та поверхневої консистенції. Із спільних точок може бути отримана хмара 3D-точок з використанням відомих параметрів камери. Реконструкція поверхні Пуассона пристосовує поверхню до хмари точок, яка далі вдосконалюється, щоб додати деталі рівня пор.

Ці реконструкції відомі як надзвичайно точні, та часто використовуються як основні істини замість лазерного сканування, оскільки обладнання відносно дешево. Використання 3D-друкованої маски замість обличчя [6] демонструє середню помилку 0,088мм використовуючи набір з семи камер DSLR. Справжній людський предмет матиме більш високу помилку, оскільки маска мала послідовне альбедо та не мала дзеркальності або підповерхневого відбиття, що є присутнім в шкірі людини. Однак, вони обходяться витраченим часом на

обчислення, позаяк для реконструкції одного зображення потрібно 20 хвилин, а об'єкт повинен бути нерухомим, та у спеціалізованій кімнаті.

1.3.1.3 Фотометричне стерео

Принципи, що лежать в основі загальної фотометричної стереосистеми, були детально описані у Розділі 1.2.2. Одним з викликів фотометричної стереосистеми є затримка часу між зображеннями, необхідна для зміни освітлення. Для твердих об'єктів це не проблема, але для нетвердих – бажано одночасно захоплювати всі зображення. Для того щоб одночасно фіксувати різні умови освітлення [18, 34], треба використовувати три різні кольорові вогні (червоний, зелений та синій) для освітлення сцени з різних ракурсів. Оскільки це мінімальна кількість варіацій освітленості для визначення нормалі поверхні, слід особливо обережно ставитись до областей тіней на відміну від підходів, що використовують чотири джерела світла [3]. Інший підхід, призначений для людини, є [15], де для освітлення використовується ближнє інфрачервоне світло, оскільки воно не таке нав'язливе, як видиме світло, і не буде турбувати об'єкт під час освітлення.

Поєднанням багаторакурсного стерео з фотометричним можна отримати надзвичайно детальні результати.

1.3.2 Необмежена

Якщо суб'єкт не кооперативний, та відсутні попередні знання про освітлення або камери, реконструкція обличчя стає необмеженою. За цими сценаріями попередні знання про форми обличчя стають надзвичайно важливими для створення переконливої реконструкції.

1.3.2.1 3D морфована модель

Ця плідна робота з необмеженої реконструкції обличчя працює на одному зображенні, та була запропонована Бланзом та Веттером [8]. Вони розробили 3D морфовану модель (3DMM), яка розширила класичну модель з активною зовнішністю від двовимірного вирівнювання обличчя задля розміщення 3D-моделі на зображенні на основі текстурної інформації. 3DMM створена за допомогою сканування 200 осіб та введення їх у зв'язок. Припущення полягає в тому, що довільне обличчя можна виразити як лінійну комбінацію відсканованих облич.

$$S_{\text{mod}} = \sum_{i=1}^{200} \alpha_i S_i, \quad T_{\text{mod}} = \sum_{i=1}^{200} \beta_i T_i, \quad \sum_{i=1}^{200} \alpha_i = \sum_{i=1}^{200} \beta_i = 1.$$

Простір обличчя можна потім стиснути і виразити у вигляді статистичного розподілу за допомогою Головного Аналізу компонента (PCA) [23],

$$S_{\text{mod}} = \bar{S} + \sum_{i=1}^{199} \alpha_i S_i, \quad T_{\text{mod}} = \bar{T} + \sum_{i=1}^{199} \beta_i T_i,$$

де ймовірність для коефіцієнтів α стає,

$$p(\vec{\alpha}) \sim \exp\left[-\frac{1}{2} \sum_{i=1}^{199} \left(\frac{\alpha_i}{\sigma_i}\right)^2\right],$$

де σ_i^2 - власне значення коваріації форми.

Маючи модель як попередні знання про очікувану форму обличчя та текстуру, мета знову полягає у виведенні параметрів пози, освітлення, форми та текстури сцени на основі одного зображення. Початкове формулювання - це аналіз методом синтезу, де похибка, це відстань між реальним зображенням та зображенням візуалізованих моделей, разом із ймовірністю спостереження коефіцієнтів. Це формулювання не випукле і зазвичай відповідає локальним

мінімумам, тому вони мають ручну анотацію освітлення та початкові параметри пози, щоб допомогти моделі прилягати до обличчя.

Щоб спробувати виправити невипуклу енергетичну функцію, [30] розширив процедуру пристосування, щоб включити краї, яскраві виділення та обмеження текстури. Морфовані моделі далі були розкладені на ідентичність та виразну основу замість єдиної комбінованої форми основи [33, 11]. Із плином часу 2D-вирівнювання орієнтира, процес пошуку ключових точок, таких як очі, ніс та рот у зображеннях покращилися. Останні роботи на основі похибки проєкції 2D-вирівнювання орієнтира, є дуже ефективними, та надають обґрунтовані результати [37, 1].

3DMM повсюдний у підходах до реконструкції обличчя і є основною складовою великого відсотка різних підходів. Однією з головних скарг на 3DMM є невеликий обсяг тренування предмета. Нещодавня стаття [9] будує нову модель з 10000 людей, та має допомогти підходам краще узагальнити широкий спектр форм обличчя.

1.3.2.2 Детекція ключових точок на обличчі

Алгоритми детектування ключових точок на обличчі (Facial feature point detection, FFPD) відносяться до класу методів навчання з учителем або часткового навчання на вручну розмічених зображеннях. Відповідно до класифікації, запропонованої в роботі Cootes [41], ключові точки на обличчі можуть бути розділені на 3 типи: точки, стійкі до повороту, такі як центр ока, межі очей або губ; точки, нестійкі до повороту, які розраховуються як екстремум на кривій, що з'єднує стійкі точки; і точки, інтерпольовані з твох попередніх типів - по краях обличчя і на підборідді. Останній тип в силу інтерполяції зберігає найменше інформації про поворот і положення обличчя.

Основний підхід до знаходження ключових точок на обличчі полягає в локалізації моделі каркаса відносно появи обличчя на зображенні. Як правило,

знаходження ключових точок складається з двох фаз: тренування, коли відбувається навчання на позиціях точок і підлаштування моделі каркаса під них, і тестування, коли навчена модель застосовується до вихідного зображення, щоб знайти положення ключових точок. Зазвичай процес пошуку каркаса починається з грубого початкового значення всього зображення цілком, але в деяких методах використовується пошук положення обличчя [51, 56], який забезпечує більш близьке початкове значення.

У більшості існуючих підходів виділяється 4 напрямки методів детектування ключових точок: Constrained Local Model (CLM), Active Appearance Model (AAM), регресивні методи та інші.

CLM-методи розглядають області появи кожної ключової точки незалежно від інших [55]. Для цього положення кожної точки всередині можливої області визначається відповідним до неї локальним експертом. Фінальне розташування точки визначається після зіставлення отриманого розташування точок одна відносно іншої із заздалегідь навченою моделлю каркаса. На відміну від них ААМ-методи використовують цілісний підхід для всіх точок [58]. Модель каркаса точок та їхні області появи будуються як лінійна комбінація базису, навченого на тренувальних зображеннях. Регресивні методи визначають загальне розташування точок без навчання будь-яких моделей.

Детальний аналіз існуючих підходів [57] показує, що каскадний набір простих лінійних регресорів дозволяє досягати найбільш точних результатів [59, 44]. Такі підходи доволі стійкі до зміни положення точок відносно одна одної, що дозволяє їм забезпечувати стабільність на повернутих зображеннях облич.

1.3.2.3 Одне зображення

Окрім 3DMM, існує ряд інших методів виконання реконструкції з єдиного зображення. [26] бере єдину референсну модель, вручну вирівнює її до

зображення та перетворює її спираючись на форму від підходу затінення. Рішенням є 2.5D відновлення фронтальної частини обличчя.

Пара недавніх робіт виконує реконструкцію за допомогою CNN для встановлення 3DMM замість пристосування, базованого на моделі [39, 24]. Ці роботи зосереджуються на проблемі щільного вирівнювання обличчя, де кожному пікселю на обличчі призначається місце на канонічному обличчі, але в процесі вони пристосовують каскад регресорів для вирішення параметрів проєкції та коефіцієнтів форми 3DMM. Таким чином, вони насправді виконують реконструкцію для зображення.

1.3.2.4 Реконструкція на основі відео

Існує великий інтерес до реконструкції на основі відео. Тут стисло висвітлюються кілька ключових робіт. У [14] відслідковуються розсіяні 2D-орієнтири у відео та 3DMM підходить до кожного кадру, моделі потім очищають за допомогою оптичного потоку та тимчасової консистенції. Цей підхід працює досить добре в дикій природі, але бореться з дзеркальними відбиттями. У [32], замість підключення 3DMM, поверхня, реконструйована з колекції фотографій, використовується як вхід, позиція для кожного кадру оцінюється і подібна до [14], оптичний потік вдосконалює вирівнювання моделі і деформується до наявних виразів, і тимчасова узгодженість підтримується як вперед і назад у відео. У [11] 3DMM відокремлено на основі ідентичності та вираження, а потім пристосовано до нерухомого зображення, а зрештою, для знаходження коефіцієнтів вираження, та перенесення виразу користувача до ляльководи використовується датчик Kinect. В [22] каскадний регресор навчений на великому наборі сканів обличчя виконувати 3D вирівнювання обличчя для одного зображення чи відео. Отримане вирівнювання має лише 1024 вершини і не фіксує деталей рівня зморшок обличчя. В [21], людина використовує мобільний телефон, щоб записати своє обличчя в нейтральному виразі з усіх поз.

Структура від руху відповідає статичній моделі обличчя з відео. Потім людина записує їх виконуючи задалегідь задані вирази, щоб побудувати модель змішаної форми. Цей інтерактивний підхід дозволяє їм створити аватари рівня зморшок, які можуть контролюватися іншими відео.

Одним з головних застосувань для реконструкції відео є ляльководство. Аватарське ляльководство це процес, коли людина контролює дії віртуальної людини, що називається аватаром, через власні дії, як правило, записані відеокамерою, але також за допомогою інших засобів, таких як датчики деформації натиснуті на обличчя. Відома кінострічка «Бути Іваном Малковичем» приклад ляльководства. Ця програма вимагає детальної моделі аватара, яка може бути повністю комп'ютерною, згенерованою художником, або реконструйованою на основі реальної людини. Модель повинна мати скелет щоб деформуватися, щоб відповідати бажаним виразам. Це відстеження обличчя дещо схоже з реконструкцією обличчя, особливо коли аватар є моделлю ляльководства. Ці системи найбільше зацікавлені у створенні візуально переконливого аватара та ефективності алгоритму щоб працювати в режимі реального часу, але метрична точність реконструкції є вторинною для візуальної згуртованості.

1.3.2.5 Колекції фото

Колекції фотографій складніші за відео, оскільки немає тимчасових обмежень. Теоретично вони мають більше інформації, ніж одне зображення, але наївне використання колекції фотографій насправді може призвести до згладженої реконструкції, оскільки обличчя не жорстке і може змінюватися між зображеннями. Колекції фотографій є дуже актуальними, оскільки більшість людей мають особисту колекцію фотографій. Колекції фото можуть бути вкрай корисними для цілей біометрики. Точна 3D-модель обличчя може покращити розпізнавання обличчя.

1.4 Застосування

Так само, як існує багато різних методів реконструкції, існує чимало застосувань для них. У цьому підрозділі наведено деякі потенційні застосування реконструкції обличчя.

1.4.1 Медичне

3D-моделі особи для особи застосовуються в медичній галузі. Вони застосовуються для візуалізації точної структури пацієнтів та проектування ділянок, що облягають. Наприклад, Мічиганський університет періодично вирівнює 3D-скани з метою відстеження можливих топографічних змін, що можуть бути спричинені різними захворюваннями. Моделі використовуються в хірургічній корекції, ортопедичній корекції, і для виправлення черепно-лицьових аномалій. Для медичних застосувань точна 3D структура є важливою, та для створення субміліметрового рівня точності необхідні 3D-сканери, або обмежені стереоустановки.

1.4.2 Розпізнавання обличчя

Розпізнавання обличчя безпосередньо застосовується від 3D до 3D. Але нещодавні роботи демонструють ефективність використання 3D-моделі для вдосконалення традиційного 2D до 2D розпізнавання обличчя. Моделі використовуються для нормалізації пози, вираження та освітлення зображень задля того, щоб візуалізувати покращене зображення, яке матиме більш високу точність розпізнавання обличчя. Незрозуміло який рівень деталізації необхідний у моделі для оптимального розпізнавання обличчя. Навіть використання гіпереліпсоїда як моделі обличчя покращує розпізнавання обличчя.

1.4.3 Комерційне редагування відео

Для редагування відео можна використовувати особисті моделі обличчя. Вони можуть бути використані для обробки відео, шляхом додавання віртуального макіяжу та зміни освітлення, або вони можуть бути використані для створення повністю CGI сцени для фільму. Для редагування відео – blend share-модель бажана, бо обличчя може бути деформоване, щоб відповідати різному набору виразів. Blend share може бути загальним набором виразів, отриманим від 3DMM, або набором виразів справжнього актора.

1.4.4 Віртуальна комунікація

Це може розглядатися як низько деталізована версія редагування відео. Реконструюючи або відстежуючи обличчя в режимі реального часу, штучна модель може бути візуалізована на іншій екран для забезпечення зв'язку. Подумайте про відеочат, але замість того, щоб надсилати відео, ви надсилаєте параметри реконструкції обличчя. Snapchat надає фільтри для зміни зовнішнього вигляду вашого власного обличчя, Face2Face показує, як керувати обличчям інших осіб, або ви можете контролювати віртуальний аватар.

1.5 Постановка задачі

Метою роботи є інформаційний аналіз та синтез системи 3D-реконструкції обличчя людини.

Для досягнення поставленої мети необхідно:

- 1) сформулювати вхідний математичний опис системи;
- 2) створити математичну модель 3D-реконструкції обличчя людини;
- 3) розробити та програмно реалізувати алгоритм 3D-реконструкції обличчя людини;
- 4) протестувати систему.

2 ВИБІР МЕТОДУ РОЗВ'ЯЗАННЯ ЗАДАЧІ

Реконструкція обличчя для перспективної камери полягає у вирішенні двох завдань комп'ютерного зору. Перше - визначення положення і орієнтації обличчя, що є рівносильним знаходженню матриці повороту і вектора перенесення камери. Задача формулюється для набору двовимірних точок на зображенні, і відповідних їм тривимірних координатах. Друга - реконструкція тривимірної поверхні обличчя. Для цього необхідно знайти тривимірні координати точок моделі та множину площин, що проходять через них.

Статистичний підхід застосовується як для однієї задачі знаходження параметрів камери, так і для першої і другої задач одночасно. При реконструкції тривимірної поверхні і знаходженні параметрів камери, для отримання тривимірної моделі в пропонованому підході використовується морфологічна модель.

2.1 Початкове наближення камери

Запропонований підхід складається з декількох підготовчих дій, які виконуються до запуску алгоритму. В першу чергу, знаходяться ключові точки на зображенні, і отриманий набір нормалізується за допомогою оберненої матриці калібрування камери. Потім співставляються із ключовими точками відповідні їм тривимірні координати за індексами в середній морфологічній моделі. Індеси отримані заздалегідь з неповернутого зображення середньої моделі для використовуваного алгоритму детекції особливих точок.

Після цього для набору двовимірних точок та їхніх тривимірних координат застосовується алгоритм OPnP і отримується початкове наближення матриці повороту R_0 та вектора перенесення t_0 камери. Це початкове наближення

використовується в якості початкового значення в алгоритмі Левенберга-Марквардта.

2.2 Алгоритм Левенберга-Марквардта

У загальному випадку для наближення параметрів камери і тривимірної поверхні обличчя можна сформулювати задачу мінімізації суми квадратів залишків (2.1), де e - вектор залишків, W – симетрична позитивно визначена вагова матриця.

$$\sum e^T W e \rightarrow \min \quad (2.1)$$

У запропонованому підході за вектор залишків приймається ортогональна помилка, що складає основну ідею ітеративного алгоритму ортогональної ітерації [52]. Згідно з ним вектор залишків розраховується як $e = M X_{cam}$, де $X_{cam} = R X(\alpha) + t$ - тривимірні точки в координатах морфованої моделі. Для цього щодо кожної здетектованої точки в однорідних координатах $x_i = (u \ v \ 1)^T$ розраховується оператор (2.2), де $x_i^- = \frac{x_i}{|x_i|}$.

$$L_i = \bar{x}_i \bar{x}_i^T \quad (2.2)$$

Будуємо блочно-діагональну матрицю M (2.3) для розрахунку тривимірної помилки із множини на діагоналі L_i , розрахованих для кожної ключової точки.

$$M = \begin{pmatrix} I - L_1 & 0 & \dots & 0 \\ 0 & I - L_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & I - L_n \end{pmatrix} \quad (2.3)$$

Основна ідея запропонованого підходу полягає у використанні зворотної коваріаційної матриці у якості вагової матриці W в алгоритмі Левенберга-Марквардта [53]. Відомо, що якщо у якості вагової матриці W використовується зворотна коваріаційна матриця V випадкових помилок (тобто $W = V^{-1}$), то

оцінки параметрів, знайдені за допомогою методу найменших квадратів (МНК), будуть найбільш ефективними. Коваріаційна матриця описує розподіл ключових точок в координатах моделі.

Отримане початкове наближення матриці повороту R_0 використовується для переведення матриці коваріації з координат моделі в координати камери. Перевод матриці в координати камери розраховується покомпонентно (2.4) для кожної здетектованої точки $i, j \in 1..n$.

$$\Sigma_{ij}^{cam} = R_0 \Sigma_{ij} R_0^T \quad (2.4)$$

У запропонованому підході коваріаційна матриця в координатах камери $V = M \Sigma^{cam} M^T$ є виродженою (сингулярною або особливою), тому в алгоритмі замість зворотної матриці використовується псевдообернена матриця V^+ , що розраховується за допомогою SVD розкладання.

2.3 Морфована модель

Морфована модель - це параметрично задана тривимірна модель, представлена лінійною комбінацією тривимірних поверхонь облич з підготовленої бази. Вона математично описується середньою моделлю обличчя $\mu \in R^{3m}$ і вектором випадкових величин $\alpha_i \in N(0, \sigma_i^2)$ із відомою дисперсією $\sigma \in R^{n-1}$ для відповідних ортонормованих базисів $U = [u_1, \dots, u_n] \in R^{3m \times n-1}$, що описують компоненти морфованої моделі (2.5).

$$shape = \mu + U \text{diag}(\sigma) \alpha \quad (2.5)$$

У запропонованому підході ключовим точкам зіставляються координати у середній моделі μ для пошуку початкового рішення ОРnP. У разі реконструкції тривимірної поверхні, середня модель приймається за початкове значення алгоритму. На кожній ітерації генерується тривимірна модель за знайденим значенням α_i та підбираються нові значення випадкових величин.

Отримані алгоритмом значення компонент морфованої моделі застосовуються для генерації тривимірної моделі обличчя, а знайдені матриця повороту і вектор переносу – для відтворення обличчя у відновленому положенні.

2.4 Алгоритм 3D-реконструкції

Далі наведено алгоритм дій для виконання 3D-реконструкції обличчя людини.

1. Розробити систему збору статистичних даних;
2. Реалізувати підхід з реконструкції 3D-поверхні обличчя та знаходження параметрів перспективної камери;
3. Розробити систему генерації тестових даних і проведення порівняльних експериментів;
4. Провести порівняння з існуючими алгоритмами знаходження перспективної камери, та алгоритмами реконструкції 3D-поверхні обличчя;
5. Дослідити залежність результатів запропонованого підходу від зміни параметрів тестових даних.

3 ІНФОРМАЦІЙНЕ ТА ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ СИСТЕМИ

Для реалізації запропонованого підходу мовою C++ з використанням бібліотек OpenGL та OpenCV була розроблена система збору статистичних даних, необхідних для розрахунку коваріаційної матриці, що лежить в основі запропонованого підходу. Підхід разом із системою проведення експериментів розроблені мовою Matlab, аналогічно алгоритмам, з якими проводилося порівняння. Для проведення експериментів на основі модулів, розроблених в рамках системи збору статистики, була створена система генерації тестових даних. З метою проведення порівняння пропонуваного підходу з аналогами було реалізовано алгоритм ортогональної ітерації [52] та базовий алгоритм розрахунку помилки перспективного перетворення [45].

3.1 Система збору статистичних даних

Основна мета створення системи збору статистики полягає в отриманні даних про розподіл ключових точок в тривимірному просторі в координатах морфованої моделі.



Рисунок 3.1 – Модулі системи збору статистики

Цикл отримання статистики по ключових точках на зображенні однієї моделі складається з проходження декількох послідовних кроків, представлених на Рисунку 3.1.

Генерація тривимірної поверхні обличчя відбувається за допомогою морфованої моделі з інструментарію 3D Basel Face Model (BFM) мовою Matlab. Для цього створюється вектор α з 199 компонент, де $\alpha_1 \in N(0, 1)$ є випадковою величиною з нормальним розподілом, а всі інші значення, це нулі. Перший коефіцієнт відповідає за форму обличчя, тому його зміна дозволяє генерувати однакові обличчя з різною формою. Отриманий вектор використовується для генерації обличчя стандартною функцією інструментарію BFM.

Отримана тривимірна модель обличчя зберігається у файл відповідно до .PLY формату опису тривимірних моделей. Щодо кожної з 53490 вершин моделі зберігається інформація про її координату та колір, після чого записується інформація про 106466 площин, що проходять через ці вершини.

Збережена тривимірна модель подається на вхід візуалізатору, котрий візуалізує обличчя у сцені за допомогою примітивів OpenGL у випадковому положенні з поворотом обличчя в діапазоні від -30° до 30° . Далі на отриманому зображенні обличчя детектуються ключові точки. Реалізація візуалізації та знаходження ключових точок описані далі у підрозділах 3.1.1 та 3.1.2 відповідно.

Для збору статистики розподілу ключових точок у просторі необхідно зіставити 3D координату вершини 2D ключовій точці у моделі. За відомими координатами x та y на зображенні сцени відновлюється глибина точки у z -буфері за допомогою функції `glReadPixels`. Далі за допомогою `Modelview`, `Projection`, `Viewport` матриць, та стандартної функції `gluUnProject` відбувається зворотне перепроєктування 2D точки у 3D координати моделі. У разі, коли знайдена ключова точка не потрапила на обличчя, 3D координату відновити неможливо, тому було додано пропуск таких винятків.

В отриманих назад даних про 3D-точки існує похибка, яка пояснюється недостатньо високою роздільною здатністю зображення, помилкою збереження

даних у 32 бітному z -буфері або просто втратою точності при перемножуванні матриць. Тому знайдена 3D-точка не збігається з координатами вершин на моделі. З метою виправлення погрішності було реалізовано пошук найближчої вершини, з мінімальною відстанню від знайденої. Для перевірки коректності роботи був створений тест, в якому знайдена вершина моделі знову візуалізується на зображенні. В середньому помилка візуалізованої вершини становить 1,23 пікселя на роздільній здатності зображень 800 x 600.

В результаті зберігається статистика про розташування ключових точок в тривимірному просторі. Інформація про знайдені 3D-координати для кожної ключової точки зберігається у файл щодо кожної моделі. Ці дії повторюються в циклі на безлічі моделей. Була зібрана статистика про 3000 моделей з малою, середньою та великою варіацією першого коефіцієнта α_1 .

Для пропонованого підходу був реалізований розрахунок коваріаційної матриці за зібраною статистикою. Матриця розраховується покомпонентно для кожної здетектованої точки за формулою (3.1), де X_i, X_j - тривимірні координати моделі, відповідні двомірним проєкціям.

$$\Sigma_{ij} = E[(X_i - EX_i)(X_j - EX_j)^T] \quad (3.1)$$

У результаті роботи системи збору статистики було розраховано 4 матриці коваріації по 1000 моделей з малою варіацією першого коефіцієнта, по 1000 із середньою, по 1000 з великою, і по 1000 з різною варіацією від малої до великої. Матриці були розраховані для порівняння результатів пропонованого підходу від варіації статистичних даних.

3.1.1 Візуалізація тривимірної моделі обличчя

Модуль візуалізації тривимірного обличчя складається з декількох послідовних етапів: завантаження об'єктів на сцену, ініціалізація параметрів візуалізації, ініціалізація і створення вікна сцени, візуалізація об'єктів у сцені,

збереження зображення сцени, очищення графічного буферу і буферів об'єктів у сцені. Також візуалізатор має набір методів, які дозволяють отримувати параметри створеного зображення, такі як фокусна відстань камери, матриця повороту, вектор перенесення та інші.

Тривимірний модель завантажується з .PLY файлу за допомогою бібліотеки Assimp, яка автоматично розраховує для кожної вершини напрямок її нормалі, що необхідно для реалістичного відтворення моделі без текстури. Далі відбувається ініціалізація параметрів візуалізації: візуалізація з текстурою або без (Рисунок 3.2), роздільна здатність вікна, кут повороту і зсув моделі, рівень Гаусовського шуму на зображенні.

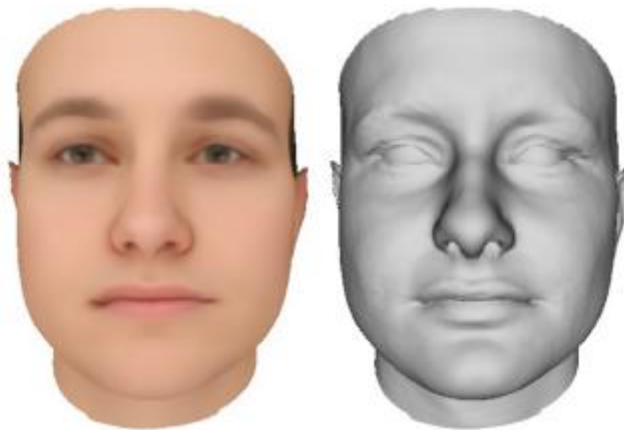


Рисунок 3.2 – Візуалізація обличчя із текстурою, та без неї

Найбільш стандартний підхід до візуалізації сцени за допомогою бібліотеки OpenGL, це використання нескінченного циклу `glutMainLoop()`. Однак, цей спосіб не дозволяє змінювати параметри створеного вікна сцени, що є критичним для проведення експерименту залежності точності підходу від роздільної здатності зображень. Більш того, такий спосіб постійно перемальовує сцену і є окремим процесом у системі, а у запропонованому підході потрібно тільки один раз візуалізувати кожне обличчя. Для вирішення проблеми використовується штучний виклик події візуалізації `glutMainLoopEvent`, який підтримується в бібліотеці OpenGL тільки для мови C++. Після візуалізації і

збереження зображення сцени, вікно сцени може бути успішно закрито, чого не можна зробити у нескінченному циклі. Цей спосіб дозволяє одноразово проініціалізувати візуалізатор функцією `glutInit` і багаторазово повністю міняти параметри візуалізації.

Візуалізація тривимірної моделі відбувається за допомогою примітиву `GL_TRIANGLES`, що на порядок повільніше, ніж при використанні вершинних та фрагментних шейдерів, однак він простіший у реалізації, а у запропонованому підході для збору статистики не потрібна швидкість, позаяк розрахунок здійснюється заздалегідь. Тому для кожної вершини передається її координата та колір, а у разі відсутності текстури ще й нормаль в точці. У разі візуалізації без текстури для створення освітленості на сцені використовується стандартне освітлення `GL_LIGHT0`.

3.1.2 Детектування ключових точок на обличчі

Для знаходження ключових точок на обличчі був обраний SDM алгоритм каскадної регресії [59], який у порівнянні з іншими дає найбільш стабільні результати при повороті. Алгоритм має готову реалізацію у відкритому доступі мовою C++, та застосовується для детектування 68 ключових точок. З цього набору були обрані 22 ключові точки, що є найбільш стійкими до повороту обличчя: по краях очей, губ та на носі (Рисунок 3.3).

У зв'язку з припущенням використання тільки стійких ключових точок, в SDM алгоритм була додана оптимізація пошуку обличчя Наар класифікатором [51], який дає більш близьке початкове значення для каскадної регресії. Для цього використовувався класифікатор `haarcascade_frontalface_alt.xml` з бібліотеки OpenCV, та вбудована функція `detectMultiScale`.

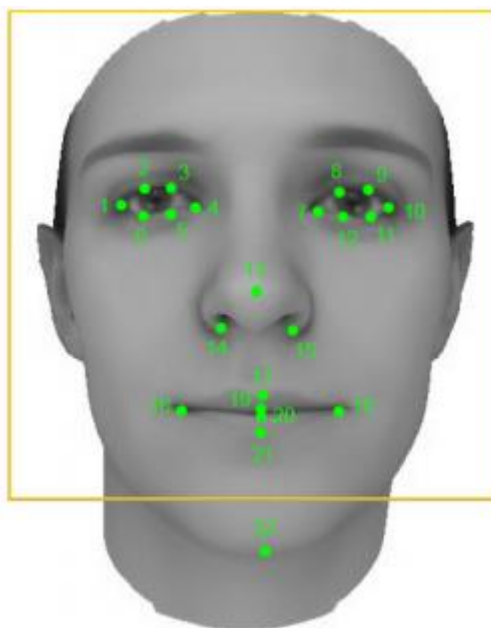


Рисунок 3.3 – Ключові точки, стійкі до повороту обличчя

З урахуванням доданої оптимізації алгоритм був навчений для дескриптора особливих точок HOG (Histogram of Oriented Gradients) [42] на відкритій базі зображень, і ключових точок IBUG з відібраних 20 зображень, що містять тільки одне обличчя. HOG дескриптор обчислюється для 4 квадратних комірок навколо кожної ключової точки, де висота однієї комірки 12 пікселів. Гістограма орієнтації кожної комірки розбивається на 4 інтервали по 90° .

Детектор був навчений для 3 послідовних лінійних регресорів. Параметри кожного регресора обчислюються як рішення лінійної системи $A \cdot x = b$ (де A - матриця тренувальних даних, x - коефіцієнти, що навчаються), знайдене методом найменших квадратів з відкритої математичної бібліотеки Eigen для мови C++. Слід зазначити, що коефіцієнти для кожної ключової точки навчаються незалежно, а навчання наступного регресора відбувається після отримання прогнозів вже навченого регресора на рівень вище.

Підібрані після навчання коефіцієнти лінійної моделі для 22 ключових точок були збережені в бінарний файл, який далі завантажувався щоразу при ініціалізації детектора, та використовувався для передбачення положення ключових точок на зображенні обличчя.

3.2 Система генерації тестів та проведення експериментів

Для проведення порівняльних експериментів, на основі розроблених модулів із системи збору статистики була розроблена система генерації тестів та проведення експериментів. Для цього були описані вимоги до тестових даних, необхідні для порівняння алгоритмів один з одним. Для проведення порівняння необхідно мати інформацію про фокусну відстань камери, про ключові точки на зображенні, про дозвіл та рівень шуму на зображенні, про матрицю повороту та вектор перенесення, а також інформацію про тривимірну поверхню обличчя. З огляду на складність формування бази тестових тривимірних поверхонь реальних облич, для порівняння результатів реконструкції використовувалися параметри облич морфованої моделі.

Для кожного тесту для проведення експериментів інформація зберігається у файл відповідно до специфікації:

- Фокусна відстань камери - float
- Кількість ключових точок - int
- Ключові точки - <int, int> []
- Роздільна здатність зображення - int [41]
- Гаусовський шум - float
- Матриця повороту - float [48]
- Вектор перенесення - float [42]
- Розмірність морфованої моделі - int
- Компоненти морфованої моделі - float []

Система проведення експериментів мовою Matlab зчитує інформацію з файлу і запускає тестування набору алгоритмів на тестових даних. У результаті роботи алгоритмів отримуються значення R_{real} та t_{real} , а у випадку реконструкції тривимірної поверхні ще й вектор компонент морфованої моделі α_{real} . Для отриманих результатів та істинних значень параметрів з тестового

файлу розраховується помилка повороту камери в градусах (3.2), помилка вектора перенесення в процентах (3.3), та сумарна помилка компонент морфованої моделі з урахуванням розподілу (3.4), де $\alpha_i \in N(0, \sigma_i^2)$.

$$E_{rot} = \max_{k=1..3} \text{acos}(\text{dot}(R_{truth}^k, R_{real}^k)) * 180/\pi \quad (3.2)$$

$$E_{trans} = \|t_{truth} - t_{real}\| / \|t\| * 100 \quad (3.3)$$

$$E_{alpha} = \sum_{i=1}^n \sigma(i)^2 * (\alpha_{truth}(i) - \alpha_{real}(i))^2 \quad (3.4)$$

3.3 Реалізація запропонованого підходу

Статистичний підхід до реконструкції тривимірної поверхні обличчя був реалізований мовою Matlab та використовує готову реалізацію алгоритму Левенберга-Марквардта з функції lsqnonlin в базовому пакеті Optimization toolbox. Робота алгоритму обмежується 20 ітераціями, з метою зменшення часу роботи. Підхід реалізований для задачі знаходження параметрів перспективної камери, а також для задачі реконструкції поверхні та камери одночасно. З метою зменшення параметрів в алгоритмі та оптимізації його роботи, матриця повороту з 9 значень параметризується 3 кутами за допомогою функції rodrigues.

3.4 Тестування системи

Для порівняння результатів знаходження параметрів перспективної камери взято OPnP алгоритм [60], який серед аналогів показує найбільш точні результати. З метою проведення експериментів з реконструкції поверхні обличчя та камери було реалізовано ітеративний алгоритм розрахунку ортогональної помилки [52] (на графіках OPnP + OI), який на відміну від запропонованого

підходу використовує в якості вагової матриці одиничну. Також був реалізований базовий алгоритм розрахунку помилки перспективного перетворення [45] (на графіках OPnP + Proj), помилка якого розраховується для $X_{proj} = K * (R * X(\alpha) - t)$ за формулою (3.5), де K - матриця калібрування камери, а $X(\alpha)$ - вершина в морфованій моделі.

$$e_{proj} = \begin{pmatrix} \frac{X_{proj}(1)}{X_{proj}(3)} - x \\ \frac{X_{proj}(2)}{X_{proj}(3)} - y \end{pmatrix} \quad (3.5)$$

Обидва реалізованих ітеративних алгоритми за початкове значення приймають рішення, знайдене OPnP алгоритмом. Помилка пропонованого підходу на усіх графіках виділена зеленим кольором.

3.4.1 Залежність від коваріаційної матриці

В результаті роботи системи збору статистики була отримана інформація про розподіл ключових точок в тривимірному просторі по 3000 моделей із випадковим поворотом від -30° до 30° . Було розраховано 4 коваріаційних матриці, що відрізняються варіацією випадкових величин в компонентах α морфованої моделі, за якими збиралася статистика. Перша матриця була розрахована по 1000 моделей з малою варіацією першого коефіцієнта (від -2 до 2), друга по 1000 із середньою (від -4 до -2 та від 2 до 4), третя по 1000 з великою (від -6 до -4 та від 4 до 6), а четверта по 1000 моделям з різною варіацією від малої до великої. Дослідження підходу для розв'язання задачі знаходження камери проводилися на 100 моделях з випадковим першим коефіцієнтом $\alpha_1 \in N(0, 1)$ та випадковим поворотом від -30° до 30° . Результати середньої помилки пропонованого підходу представлені на Рисунку 3.4, помилка матриці повороту в градусах (3.2), та вектор перенесення в процентах (3.3). Графіки медіани помилки представлені в Додатку А на Рисунку А.1.

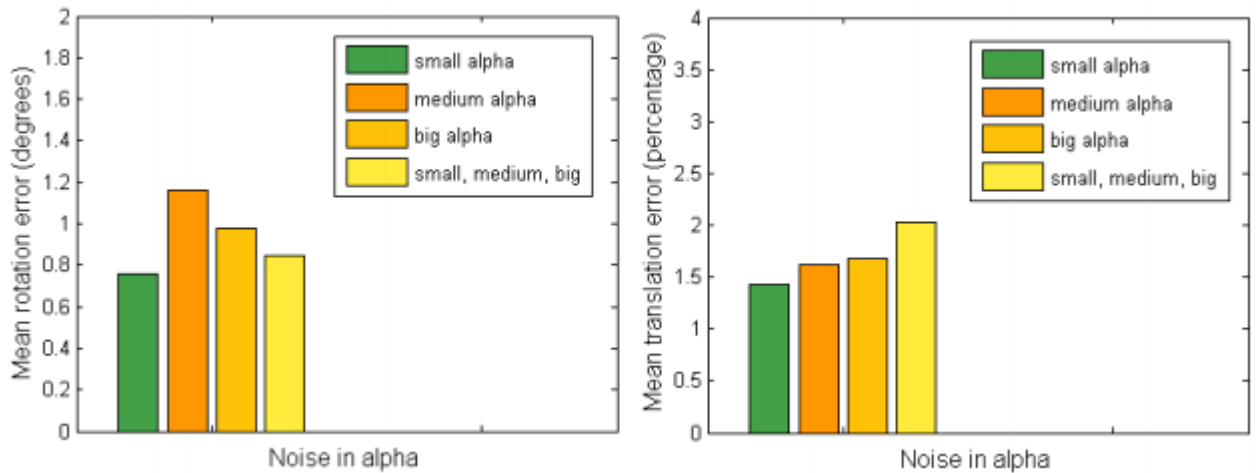


Рисунок 3.4 – Середня помилка камери від різної матриці коваріацій

Також була досліджена залежність від коваріаційної матриці результатів пропонованого підходу для задачі реконструкції тривимірної поверхні та параметрів камери. Дослідження проводилися на 50 моделях з усіма випадковими компонентами морфологічної моделі $\alpha_i \in N(0, 1)$, $i = 1..199$ та випадковим поворотом від -30° до 30° . Результати середньої помилки реконструкції поверхні обличчя (3.4) та камери представлені на Рисунку 3.5. Графіки медіани помилки представлені в Додатку А на Рисунку А.2.

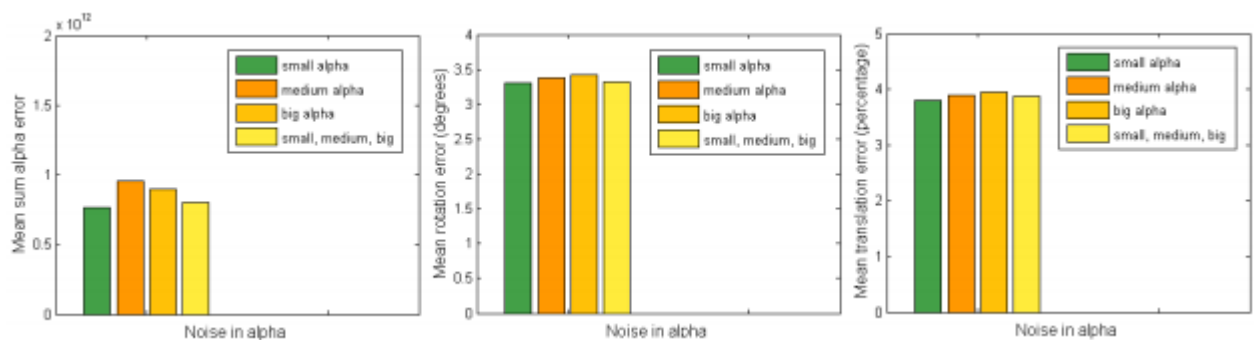


Рисунок 3.5 – Середня помилка реконструкції поверхні та камери від різної матриці коваріацій

З Рисунків 3.4 та 3.5 можна зробити загальний висновок, що найбільш точні та стабільні результати були отримані з використанням матриці коваріації

з малою варіацією поверхні обличчя. Це може бути наслідком того, що обличчя з середньою та великою варіацією виглядають менш реалістично.

Порівняння пропонованого підходу, представлене в наступних підрозділах, проводилося з використанням коваріаційної матриці з малою варіацією.

3.4.2 Задача знаходження перспективної камери

Для порівняння алгоритмів було проведено дослідження залежності результатів від кута повороту обличчя. Для цього взята одна модель з випадковим першим коефіцієнтом морфованої моделі $\alpha_1 \in N(0, 1)$ та іншими компонентами рівними нулю. Модель рівномірно обертається з кроком в 10° у діапазоні від -30° до 30° . Результати для задачі знаходження перспективної камери представлені на Рисунку 3.6, помилки матриці повороту та вектора перенесення.

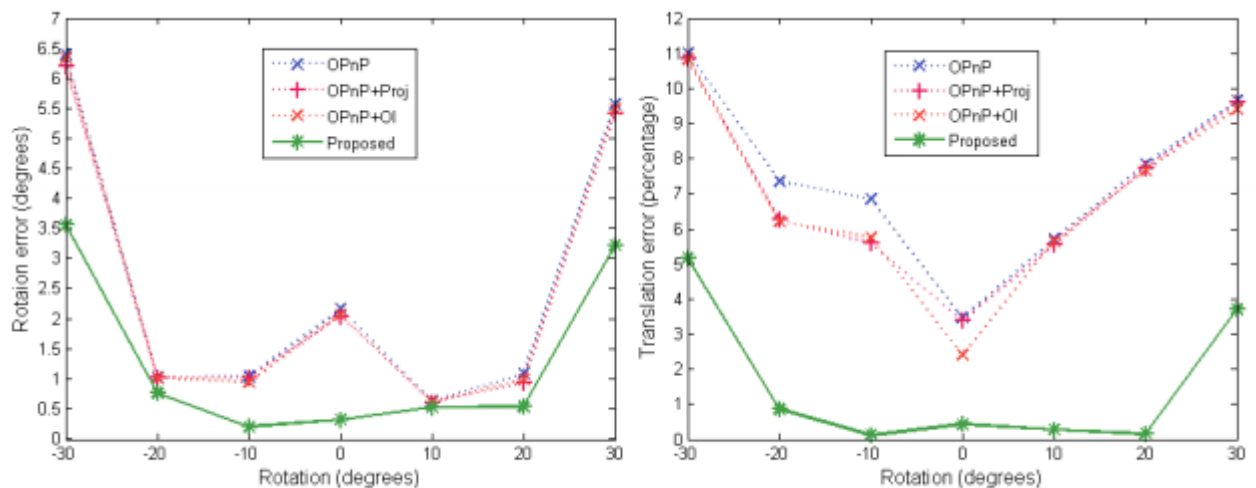


Рисунок 3.6 – Залежність помилки повороту та переносу від повороту обличчя

На графіках видно, що пропонований статистичний підхід показує найбільш точні результати. Особливо варто відзначити стабільність результатів при повороті обличчя до 20° та стійкість до сильного повороту на 30° .

Для розрахунку середньої помилки та медіани було підготовлено 100 тестів на основі тривимірних облич з випадковою першою компонентою морфованої моделі $\alpha_1 \in N(0, 1)$ на випадковому повороті від -30° до 30° . Середня помилка матриці повороту та вектора перенесення представлена на Рисунку 3.7. Графіки медіани помилки представлені у Додатку А на Рисунку А.3.

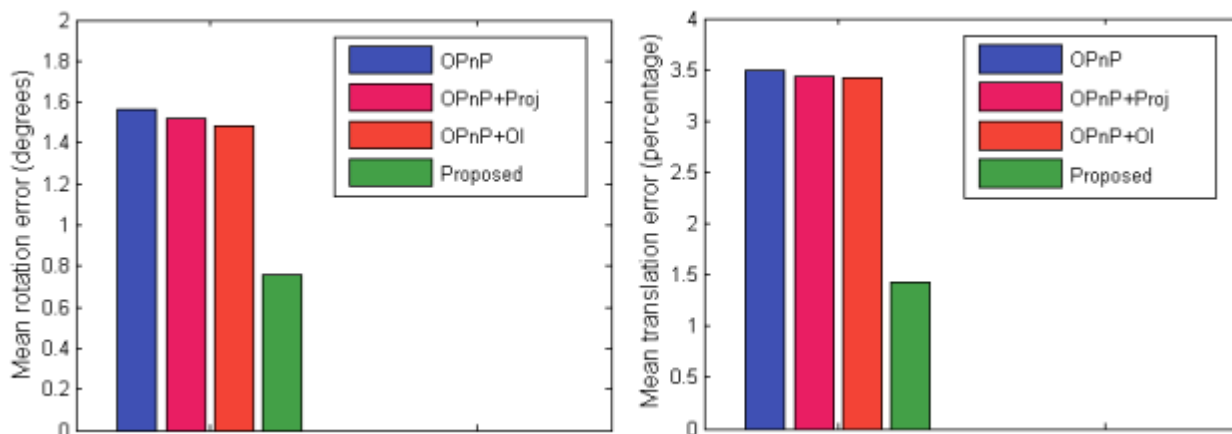


Рисунок 3.7 – Середня помилка повороту та переносу на 100 тестів

На рисунку видно, що помилка пропонованого підходу для визначення повороту менша за 1° і складає близько 1.5% для перенесення. Отриманий результат майже в 2 рази точніший всіх алгоритмів, з якими проводилося порівняння.

3.4.3 Реконструкція поверхні обличчя та камери

Для задачі реконструкції тривимірної поверхні обличчя і знаходження параметрів перспективної камери також були проведені аналогічні експерименти з дослідження залежності помилки від кута повороту обличчя і порівнянню середньої помилки з аналогами. Експерименти проводилися на повністю випадкових обличчях морфованої моделі з усіма випадковими коефіцієнтами $\alpha_i \in N(0, 1)$, $i = 1..199$.

Дослідження залежності від кута проводилося при рівномірному повороті одного обличчя з кроком в 10° у діапазоні від -30° до 30° . Помилка відновлення

параметрів морфованої моделі α_i , $i = 1..199$, помилка повороту і перенесення представлені на Рисунку 3.8.

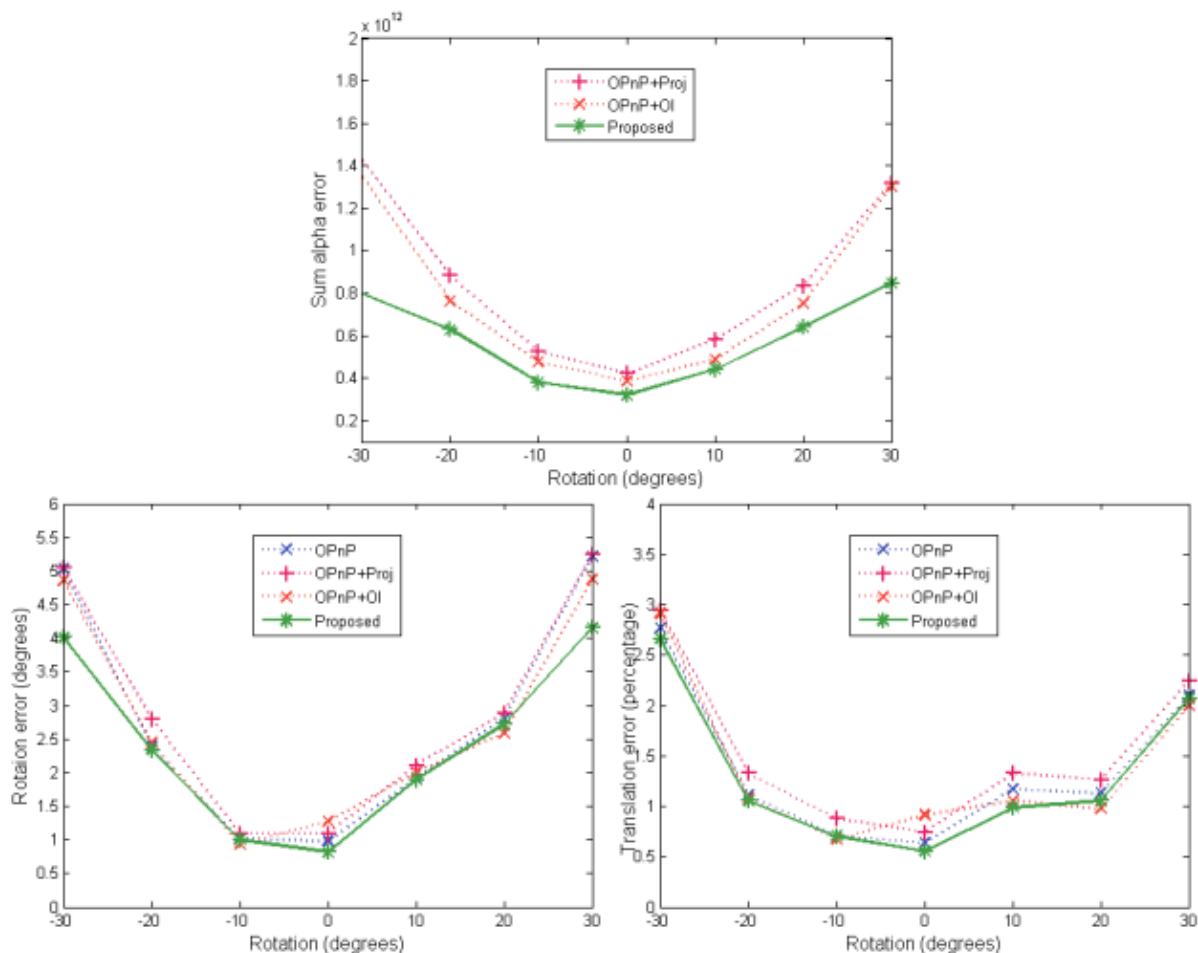


Рисунок 3.8 – Залежність від повороту обличчя помилки параметрів морфованої моделі, повороту та перенесення

На графіку помилки повороту та перенесення видно, що пропонуванний підхід дає помилку близьку до аналогів, однак стабільно показує один з найкращих результатів. Варто зазначити, що алгоритм ортогональної помилки, та алгоритм розрахунку помилки перспективного перетворення погіршують початкові значення повороту, знайдені OPnP алгоритмом, за рахунок оптимізації параметрів морфованої моделі. На відміну від них, пропонуванний підхід, як і для задачі знаходження камери, тільки покращує початкове значення. При цьому він точніший за аналоги при реконструкції поверхні обличчя.

Для розрахунку середнього та медіани було проведено 50 тестів з випадковим поворотом моделі обличчя від -30° до 30° . Помилки параметрів морфованої моделі, помилка матриці повороту і вектора перенесення представлені на Рисунку 3.9. Медіана помилок представлена у Додатку А на Рисунку А.4.

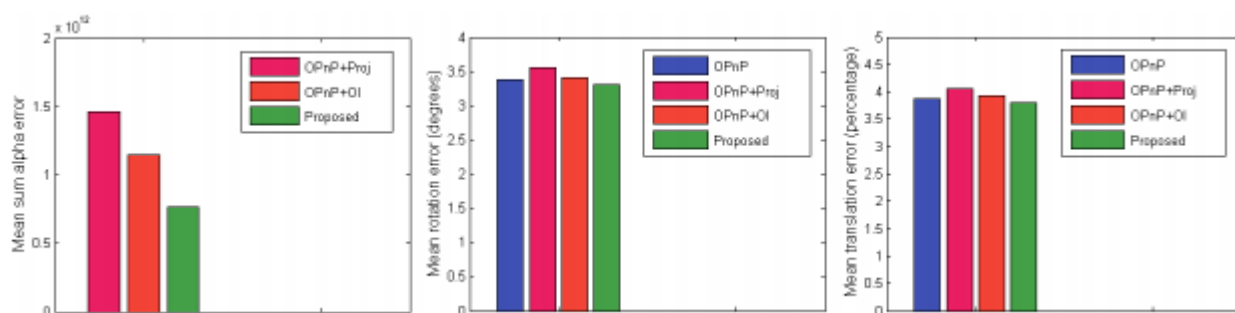


Рисунок 3.9 – Середня помилка тривимірної поверхні та камери на 50 тестах

В середньому пропонуванний підхід дає найбільш точний результат реконструкції поверхні обличчя. При цьому початкове значення знайдене OPnP алгоритмом не погіршується, а навпаки поліпшується на невелике значення. На Рисунку 3.10 показані приклади реконструйованих пропонуванним підходом поверхонь облич.

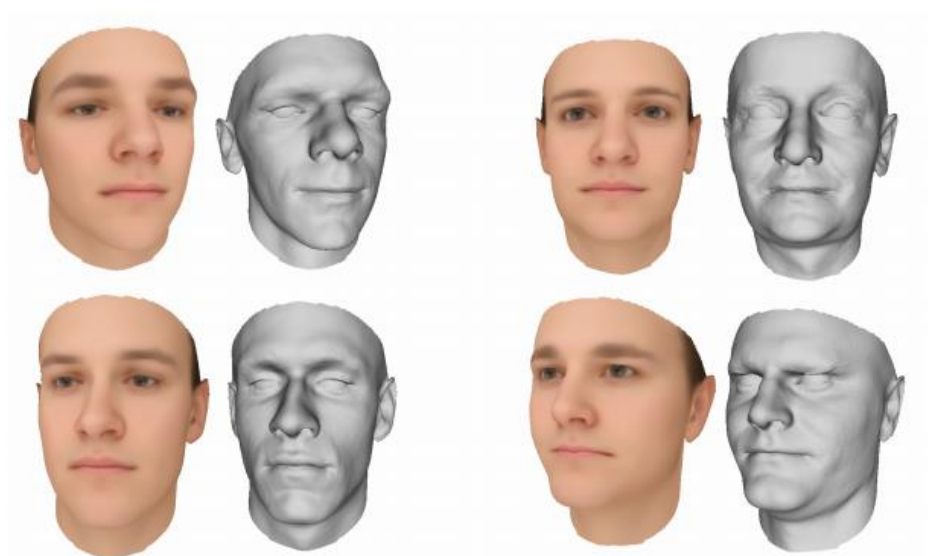


Рисунок 3.10 – Реконструйовані обличчя з урахуванням знайденого розташування

Зліва показано вихідне зображення обличчя, а праворуч отримана тривимірна модель з урахуванням знайденого повороту та перенесення.

3.4.4 Залежність від параметрів тестових даних

З метою перевірки стабільності роботи запропонованого підходу було проведено дослідження залежності результатів від рівня шуму на вихідних зображеннях. При додаванні випадкового Гаусовського шуму SDM детектор дає зміщення ключової точки в середньому на 4,12 пікселі на зашумленому зображенні щодо зображення без шуму. Дослідження проводилося на зображеннях із роздільною здатністю 800 x 600. Дослідження помилки повороту та перенесення для задачі знаходження параметрів камери представлено на Рисунку 3.11.

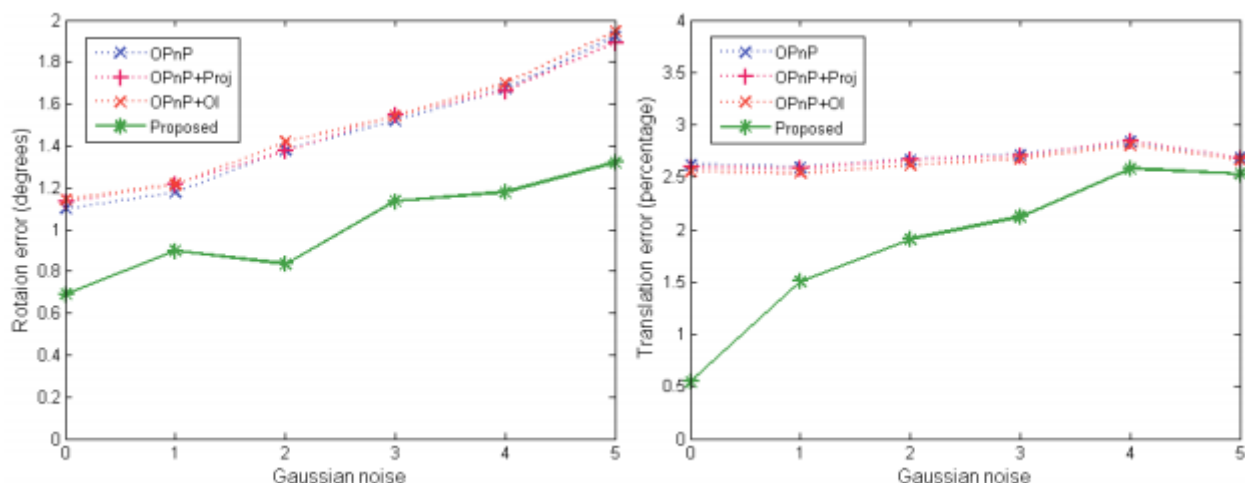


Рисунок 3.11 – Помилка камери від додавання шуму на зображення

З графіків можна зробити висновок, що помилка запропонованого підходу збільшується з додаванням шуму. При цьому помилка повороту аналогів також збільшується, через що зберігається різниця з запропонованим підходом. Для знаходження перенесення, аналоги є більш стабільними. На Рисунку А.5 у Додатку А представлений графік залежності помилки для задачі реконструювання тривимірної поверхні та камери.

ВИСНОВКИ

У кваліфікаційній роботі було виконано огляд існуючих способів реконструкції поверхні, розроблено, програмно реалізовано, та протестовано інформаційну технологію 3D реконструкції обличчя людини за 2D зображенням.

А саме було виконано наступні задачі:

- розроблено систему збору статистики розподілу ключових точок у тривимірному просторі;
- реалізовано статистичний підхід до вирішення задачі реконструкції тривимірного обличчя за зображенням;
- розроблено систему генерації тестових даних та проведення експериментів;
- досліджено залежність результатів пропонованого підходу від коваріаційної матриці і додавання Гаусовського шуму на зображення облич.

Порівняння з існуючими підходами показало, що для задачі знаходження камери запропонований підхід точніший за аналоги у 2 рази, а для задачі реконструкції поверхні обличчя та камери, дає помилку по камері близьку до аналогів, та в 2 рази меншу помилку поверхні обличчя;

Таким чином, було отримано, що запропонований підхід на базі морфованої моделі дає в порівнянні з аналогами найбільш стабільні результати на різних тестових даних, як для задачі знаходження перспективної камери, так і для задачі реконструкції поверхні обличчя та камери одночасно.

Подальша робота з метою поліпшення отриманих результатів може бути спрямована у бік застосування інших алгоритмів детектування ключових точок, а також збільшення кількості використовуваних в алгоритмі ключових точок за рахунок додавання нестійких особливих точок по краях обличчя та підборіддя.

СПИСОК ЛІТЕРАТУРИ

1. O. Aldrian. Inverse rendering of faces with a 3D morphable model. *IEEE Trans. Pattern Anal. Mach. Intell.*, 35(5):1080–1093, May 2013.
2. M. Alexa, J. Behr, D. Cohen-Or, S. Fleishman, D. Levin, and C. T. Silva. Computing and rendering point set surfaces. *Visualization and Computer Graphics, IEEE Transactions on*, 9(1):3–15, 2003.
3. S. Barsky and M. Petrou. The 4-source photometric stereo technique for three-dimensional surfaces in the presence of highlights and shadows. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(10):1239–1252, 2003.
4. R. Basri and D. Jacobs. Lambertian reflectance and linear subspaces. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(2):218–233, 2003.
5. R. Basri, D. Jacobs, and I. Kemelmacher. Photometric stereo with general, unknown lighting. *Int. J. Comput. Vision*, 72(3):239–257, 2007.
6. T. Beeler, B. Bickel, P. Beardsley, R. Sumner, and M. Gross. High-quality single-shot capture of facial geometry. *ACM Trans. Graph.*, 29(3), 2010.
7. T. Beeler, F. Hahn, D. Bradley, B. Bickel, P. Beardsley, C. Gotsman, and M. Gross. Highquality passive facial performance capture using anchor frames. *ACM Trans. Graph.*, 30(4):75:1–75:10, 2011.
8. V. Blanz and T. Vetter. Face recognition based on fitting a 3D morphable model. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(9):1063–1074, 2003.
9. J. Booth, A. Roussos, S. Zafeirious, A. Ponniah, and D. Dunaway. A 3D morphable model learnt from 10,000 faces. In *CVPR*, pages 5543–5552, 2016.
10. G. Borshukov and L. J. P. Realistic human face rendering for “the matrix reloaded”. In *ACM Siggraph 2005 Courses*, page 13, 2005.
11. C. Cao, Y. Weng, S. Zhou, Y. Tong, and K. Zhou. Facewarehouse: a 3D facial expression database for visual computing. *IEEE Trans. Vis. Comput. Graphics*, 20(3):413–425, 2014.

12. J. C. Carr, R. K. Beatson, J. B. Cherrie, T. J. Mitchell, W. R. Fright, B. C. McCallum, and T. R. Evans. Reconstruction and representation of 3D objects with radial basis functions. In Proc. of the 28th annual conf. on Computer graphics and interactive techniques, pages 67–76. ACM, 2001.
13. B. Curless and M. Levoy. A volumetric method for building complex models from range images. In Proc. of the 23rd annual conf. on Computer graphics and interactive techniques, pages 303–312. ACM, 1996.
14. P. Garrido, L. Valgaerts, C. Wu, and C. Theobalt. Reconstructing detailed dynamic face geometry from monocular video. *ACM Trans. Graph.*, 33(6):158, 2013.
15. M. F. Hansen, G. A. Atkinson, L. N. Smith, and M. L. Smith. 3D face reconstructions from photometric stereo using near infrared and visible light. *CVIU*, 114(8):942–951, 2010.
16. R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2004.
17. H. Hayakawa. Photometric stereo under a light source with arbitrary motion. *J. Optical Soc. America A.*, 11(11):3079–3089, 1994.
18. C. Hernández, G. Vogiatzis, G. Brostow, B. Stenger, and R. Cipolla. Non-rigid photometric stereo with colored lights. In *ICCV*, 2007.
19. C. Hernández, G. Vogiatzis, and R. Cipolla. Multiview photometric stereo. *IEEE Trans. Pattern Anal. Mach. Intell.*, 30(3):548–554, 2008.
20. H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle. Surface reconstruction from unorganized points. *SIGGRAPH Comput. Graph.*, 26(2):71–78, July 1992.
21. A. E. Ichim, S. Bouaziz, and M. Pauly. Dynamic 3D avatar creation from hand-held video input. *ACM Trans. Graph.*, 34(4):45, 2015.
22. L. A. Jeni, J. F. Cohn, and T. Kanade. Dense 3D face alignment from 2D videos in real-time. In *FG*, 2015.
23. I. Jolliffe. *Principal Component Analysis*. Springer-Verlag, 1986.

24. A. Jourabloo and X. Liu. Large-pose face alignment via CNN-based dense 3D model fitting. In CVPR, 2016.
25. M. Kazhdan, M. Bolitho, and H. Hoppe. Poisson surface reconstruction. In Proc. of the 4th Eurographics symposium on Geometry processing, 2006.
26. I. Kemelmacher-Shlizerman and R. Basri. 3D face reconstruction from a single image using a single reference face shape. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(2):394–405, 2010.
27. K. Lee, J. Ho, and D. Kriegman. Nine points of light: Acquiring subspaces for face recognition under variable lighting. In CVPR, pages 129–139, 2001.
28. R. Newcombe, D. Fox, and S. Seitz. Dynamicfusion: Reconstruction and tracking on nonrigid scenes in real-time. In CVPR, pages 343–352, 2015.
29. M. Pauly, N. J. Mitra, J. Giesen, M. H. Gross, and L. J. Guibas. Example-based 3d scan completion. In Symposium on Geometry Processing, number EPFL-CONF-149337, pages 23–32, 2005.
30. S. Romdhani and T. Vetter. Estimating 3D shape and texture using pixel intensities, edges, specular highlights, texture constraints and a prior. In CVPR, 2005.
31. B. Shi, K. Inose, Y. Matsushita, and P. Tan. Photometric stereo using internet images. In 3DV, pages 361–368, 2014.
32. S. Suwajanakorn, I. Kemelmacher-Shlizerman, and S. M. Seitz. Total moving face reconstruction. In ECCV, pages 796–812. Springer, 2014.
33. D. Vlastic, M. Brand, H. Pfister, and J. Popovi'c. Face transfer with multilinear models. *ACM Trans. Graph.*, 24(3):426–433, July 2005.
34. G. Vogiatzis and C. Hernández. Practical 3d reconstruction based on photometric stereo. In *Computer Vision*, pages 313–345. Springer, 2010.
35. R. J. Woodham. Photometric method for determining surface orientation from multiple images. *Optical Engineering*, 19(1):139–144, 1980.

36. L. Wu, A. Ganesh, B. Shi, Y. Matsushita, Y. Wang, and Y. Ma. Robust photometric stereo via low-rank matrix completion and recovery. In ACCV, pages 703–717, 2010.
37. F. Yang, J. Wang, E. Shechtman, L. Bourdev, and D. Metaxas. Expression flow for 3D-aware face component transfer. *ACM Trans. Graph.*, 30(4), July 2001.
38. A. L. Yuille, D. Snow, R. Epstein, and P. N. Belhumeur. Determining generative models of objects under varying illumination: Shape and albedo from multiple images using SVD and integrability. *Int. J. Comput. Vision*, 35:203–222, 1999.
39. X. Zhu, Z. Lei, X. Liu, H. Shi, and S. Z. Li. Face alignment across large poses: A 3D solution. In CVPR, 2016.
40. Blanz Volker, Vetter Thomas. A morphable model for the synthesis of 3D faces // *Proceedings of the 26th annual conference on Computer graphics and interactive techniques.* / ACM Press/Addison-Wesley Publishing Co. — 1999. — P. 187–194.
41. Cootes T. F., et al. Active shape models-their training and application // *Computer vision and image understanding.* — 1995. — P. 38–59.
42. Dalal N., Triggs B. Histograms of oriented gradients for human detection // *Computer Vision and Pattern Recognition.* — 2005. — P. 886–893.
43. DeMenthon D., Davis L. S. Face alignment through subspace constrained mean-shifts // *IEEE Transactions on Pattern Analysis & Machine Intelligence.* — 1992. — P. 1100–1105.
44. Face alignment by explicit shape regression / X. Cao, Y. Wei, F. Wen, J Sun // *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition.* — 2012. — P. 2887–2894.
45. Haralick R. M., et al. Pose estimation from corresponding point data // *Systems, Man and Cybernetics, IEEE Transactions on.* — 1989. — P. 1426–1446.
46. Hartley R., Zisserman A. Multiple view geometry in computer vision. — Cambridge university press, 2003.

47. Huber P., et al. Fitting 3D Morphable Models using local features // arXiv preprint arXiv:1503.02330. — 2015.
48. Izadi S., et al. KinectFusion: real-time 3D reconstruction and interaction using a moving depth camera // Proceedings of the 24th annual ACM symposium on User interface software and technology. / ACM. — 2011. — P. 559–568.
49. [13] Lepetit V., Moreno-Noguer F., Fua P. Epnp: An accurate $O(n)$ solution to the pnp problem // International journal of computer vision. — 2009. — P. 155–166.
50. Li S., Xu C., Xie M. A robust $O(n)$ solution to the perspective-npoint problem // Pattern Analysis and Machine Intelligence, IEEE Transactions on. — 2012. — P. 1444–1450.
51. Lienhart R., Maydt J. An extended set of haar-like features for rapid object detection // Image Processing. 2002. Proceedings. 2002 International Conference on. / IEEE. — 2002. — P. 900–903.
52. Lu C. P., Hager G. D., Mjolsness E. Fast and globally convergent pose estimation from video images // Pattern Analysis and Machine Intelligence, IEEE Transactions on. — 2000. — P. 610–622.
53. Moré J. J. The Levenberg-Marquardt algorithm: implementation and theory. — Numerical analysis : Springer Berlin Heidelberg, 1978.
54. Parke F.I. A parametric model for human faces. №. UTEC-CSc-75- 047. — UTAH UNIV SALT LAKE CITY DEPT OF COMPUTER SCIENCE, 1974.
55. Saragih J. M., Lucey S., Cohn J. F. Deformable model fitting by regularized landmark mean-shift // International journal of computer vision. — 2011. — P. 200–215.
56. Viola P., Jones M. J. Robust real-time face detection // International journal of computer vision. — 2004. — P. 137–154.
57. Wang N., et al. Facial feature point detection: A comprehensive survey // arXiv preprint arXiv:1410.1037. — 2014.

58. X. Gao, et al. A review of active appearance models // Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on. — 2010. — P. 145–158.
59. Xiong X., Torre F. Supervised descent method and its applications to face alignment // Proceedings of the IEEE conference on computer vision and pattern recognition / IEEE. — 2013. — P. 532–539.
60. Zheng Y. Revisiting the pnp problem: A fast, general and optimal solution // Proceedings of the IEEE International Conference on Computer Vision. — 2013. — P. 2344–2351.

ДОДАТОК А

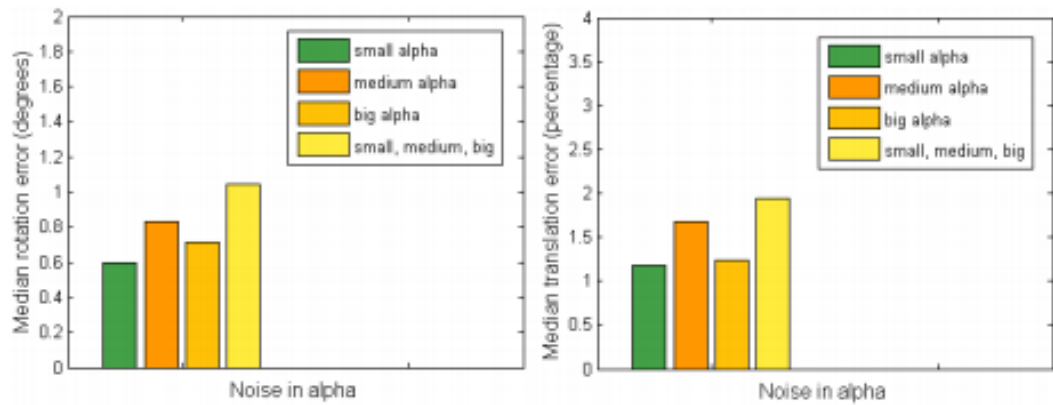


Рисунок А.1 – Медіана помилки камери від коваріаційної матриці

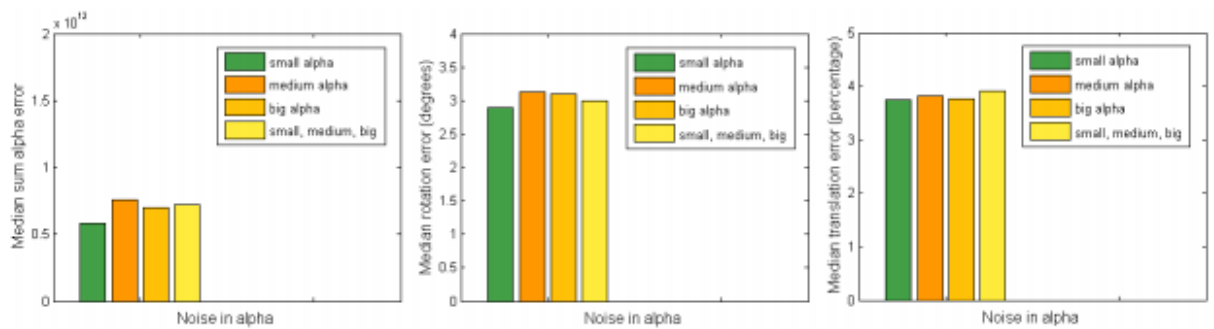


Рисунок А.2 – Медіана помилки реконструкції поверхні та камери у залежності від коваріаційної матриці

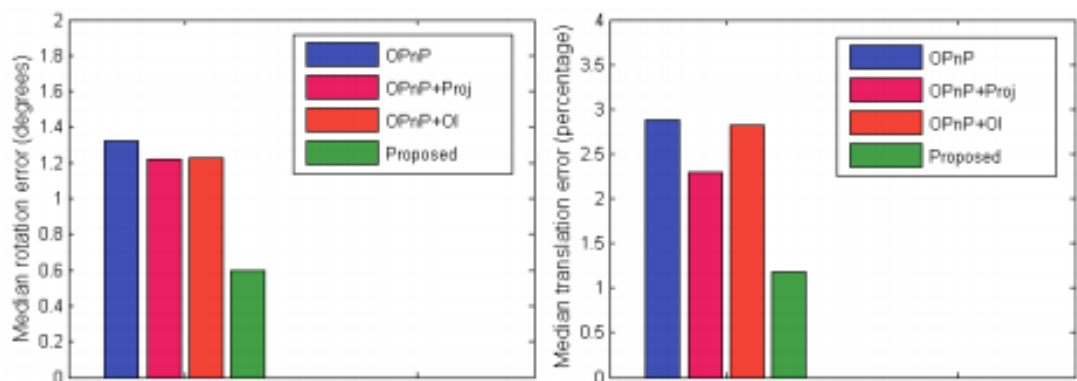


Рисунок А.3 – Медіана помилки для задачі знаходження камери

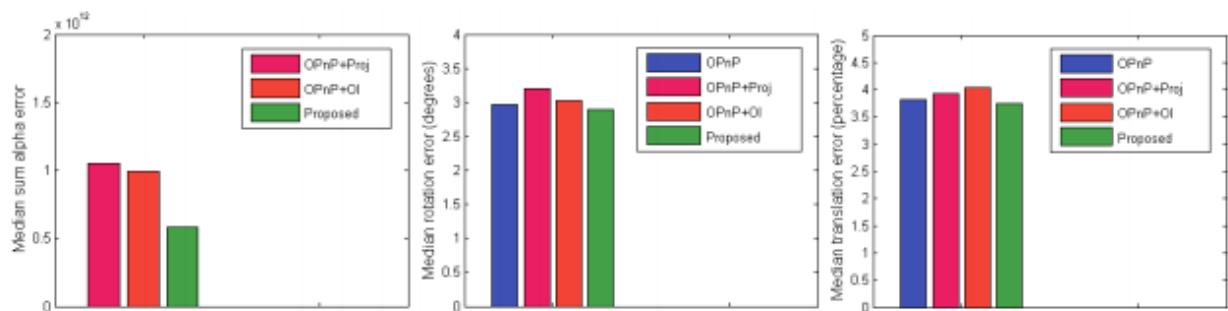


Рисунок А.4 – Медіана помилки для задачі реконструкції тривимірної поверхні та знаходження камери

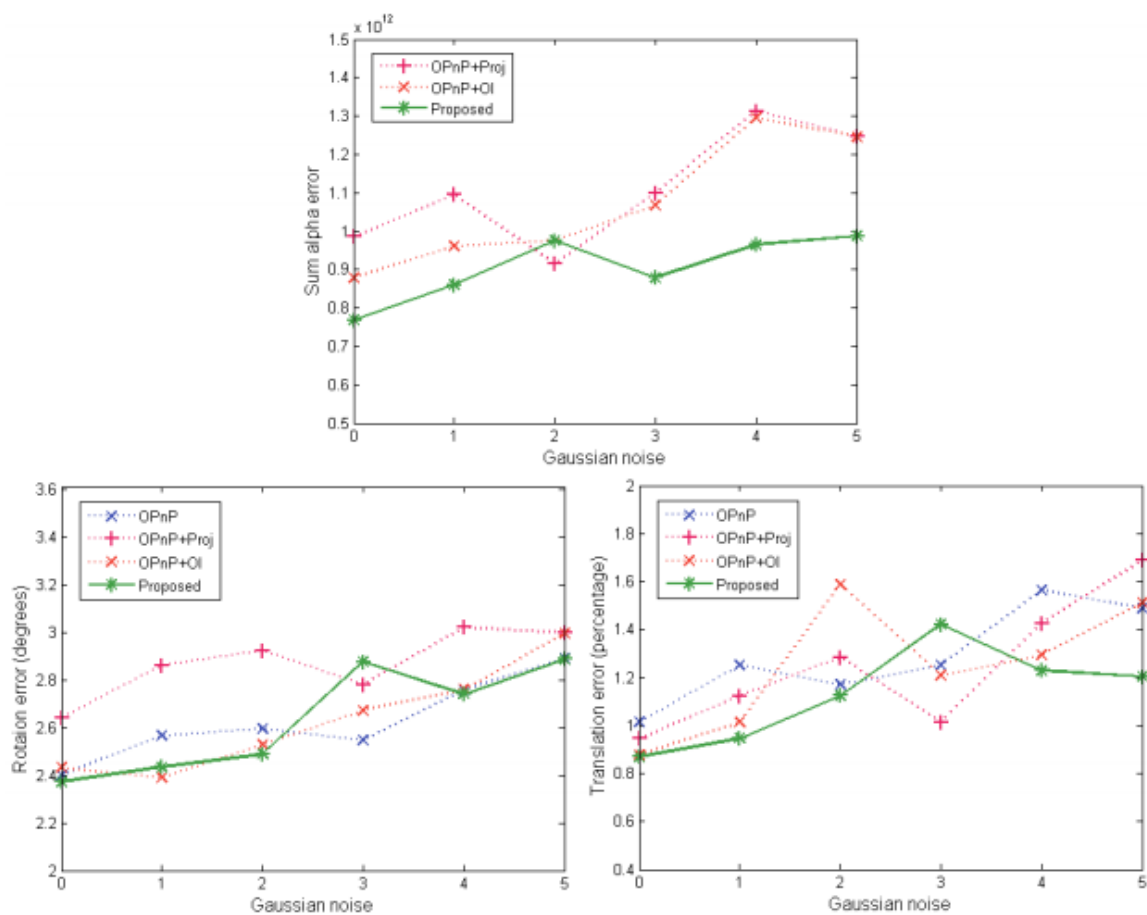


Рисунок А.5 – Помилка в залежності від шуму на зображеннях для задачі реконструкції поверхні та камери

ДОДАТОК Б

Face.m

```

function [F] = face(
sigma,M,N,win_size,features_m,disparity_feature_points,dispartity1_ref,ma
sk_middle)

w=(win_size-1)/2;

gaussians = zeros(M,N,74);
for i=1:68
    x=uint32(features_m(i,1));
    y=uint32(features_m(i,2));
    gaussians(y-w:y+w,x-w:x+w,i) = fspecial('gauss',win_size,20);
end

c1 = uint32(features_m(2,:) + (features_m(30,)-features_m(2,))/3);
c2 = uint32(features_m(4,:) + (features_m(32,)-features_m(4,))/2);
c3 = uint32(features_m(36,:) + (features_m(14,)-features_m(36,))/2);
c4 = uint32(features_m(30,:) + (features_m(16,)-features_m(30,))/3);
c5 = uint32(features_m(2,:) + 2*(features_m(30,)-features_m(2,))/3);
c6 = uint32(features_m(30,:) + 2*(features_m(16,)-features_m(30,))/3);

gaussians(c1(2)-w:c1(2)+w,c1(1)-w:c1(1)+w,69) =
fspecial('gauss',win_size,sigma(1,1));
gaussians(c2(2)-w:c2(2)+w,c2(1)-w:c2(1)+w,70) =
fspecial('gauss',win_size,sigma(1,1));
gaussians(c3(2)-w:c3(2)+w,c3(1)-w:c3(1)+w,71) =
fspecial('gauss',win_size,sigma(1,1));
gaussians(c4(2)-w:c4(2)+w,c4(1)-w:c4(1)+w,72) =
fspecial('gauss',win_size,sigma(1,1));
gaussians(c5(2)-w:c5(2)+w,c5(1)-w:c5(1)+w,73) =
fspecial('gauss',win_size,sigma(1,1));
gaussians(c6(2)-w:c6(2)+w,c6(1)-w:c6(1)+w,74) =
fspecial('gauss',win_size,sigma(1,1));

disp=zeros(74,1);
disp(1:68,1) = disparity_feature_points;
w1=dispartity1_ref(c1(2)-w:c1(2)+w,c1(1)-w:c1(1)+w);
w2=dispartity1_ref(c2(2)-w:c2(2)+w,c2(1)-w:c2(1)+w);
w3=dispartity1_ref(c3(2)-w:c3(2)+w,c3(1)-w:c3(1)+w);
w4=dispartity1_ref(c4(2)-w:c4(2)+w,c4(1)-w:c4(1)+w);
w5=dispartity1_ref(c5(2)-w:c5(2)+w,c5(1)-w:c5(1)+w);
w6=dispartity1_ref(c6(2)-w:c6(2)+w,c6(1)-w:c6(1)+w);

disp(69,1) = sum(sum(w1))/sum(sum(w1~=0));
disp(70,1) = sum(sum(w2))/sum(sum(w2~=0));
disp(71,1) = sum(sum(w3))/sum(sum(w3~=0));
disp(72,1) = sum(sum(w4))/sum(sum(w4~=0));
disp(73,1) = sum(sum(w5))/sum(sum(w5~=0));
disp(74,1) = sum(sum(w6))/sum(sum(w6~=0));

```

```

tot=sum(gaussians,3);
for i=1:74
    gaussians(:,:,i)=gaussians(:,:,i)./tot;
end

F = zeros(M,N);
fo=zeros(74,1);
for k=1:M
    for l=1:N
        if mask_middle(k,l)==1
            fo(:,l)=gaussians(k,l,:);
            F(k,l) = sum(fo.*disp);
        end
    end
end
end

%Smooth
% h=fspecial('gauss',100,20);
% F=imfilter(F,h);

%Cost function
%J = sum(sum((dispartityl_ref_out - F).^2));

end

```

face_mask_extraction.m

```

function [ foreground ] = face_mask_extraction( I , which_image )
img = imcrop(I,[1 1 size(I,1) size(I,2)/2-1]);
%img = imresize(img,0.5);
[m,n,x] = size(img);

% k-means for half part of the input image
pixel_labels = k_means(img,3);
%figure();imshow(pixel_labels,[]);

%Expand the image
pixel_labels_extended = zeros(2*m,n);
pixel_labels_extended(1:m,:) = pixel_labels(:,:);

%Labels
if which_image=='r'
    label_back = round(mean(mean((pixel_labels(1:end,1:round((n/5)))))));
elseif which_image=='l'
    label_back =
round(mean(mean((pixel_labels(1:end,round(4*n/5):end)))));
else
    label_back = round(mean(mean((pixel_labels(1:end,1:round((n/5)))))));
end

labeled_pixels = pixel_labels_extended ~= label_back;
labeled_pixels(m+1:2*m,:) = 0;
%figure();imshow(labeled_pixels,[]);

% k-means for entire image

```

```

pixel_labels2 = k_means(I,3);

%Labels
if which_image=='r'
    label_face =
round(mean(mean((pixel_labels2(round(2*m/5):round(3*m/5),round(4*n/5):end
)))));
elseif which_image=='l'
    label_face =
round(mean(mean((pixel_labels2(round(2*m/5):round(3*m/5),1:round(1*n/5))
)))));
else
    label_face =
round(mean(mean((pixel_labels2(round(2*m/5):round(3*m/5),round(2*n/5):rou
nd(3*n/5))))));
end

labeled_pixels2 = pixel_labels2 == label_face;

foreground = (labeled_pixels | labeled_pixels2);
%figure();imshow(foreground,[]);title('foreground');

%Deleting small details
SE = strel('disk', 12);
seed = imerode(foreground,SE);
foreground = imreconstruct(seed,foreground);
%figure();imshow(foreground,[]);title('delete small details');

%% Region Growing
% foreground = region_growing(I,foreground,30);
% figure();imshow(foreground,[]);title('region growing');

% %Deleting small details
% SE = strel('disk', 12);
% seed = imerode(foreground,SE);
% foreground = imreconstruct(seed,foreground);
% figure();imshow(foreground,[]);title('delete small details 2');

%Region-filling
SE2 = strel('disk',8);
SE3 = strel('disk',30);
foreground = imopen(foreground,SE2);
foreground = imclose(foreground,SE3);
%figure();imshow(foreground,[]);title('region filling');

%Deleting small details
SE = strel('disk', 24);
seed = imerode(foreground,SE);
foreground = imreconstruct(seed,foreground);
%figure();imshow(foreground,[]);title('delete small details 3');
End

mesh_create_func.m

function [] = mesh_create_func( img, disparityMap, xyzPoints, unreliable)
%% create a connectivity structure

```

```

[M, N] = size(disparityMap);           %get image size
res = 2;                               %resolution of mesh
[nI,mI] = meshgrid(1:res:N,1:res:M); %create a 2D meshgrid of pixels,
thus defining a resolution grid
TRI = delaunay(nI(:),mI(:));          %create a triangle connectivity list
indI = sub2ind([M,N],mI(:),nI(:));    %cast grid points to linear indices

%% linearize the arrays and adapt to chosen resolution
pcl = reshape(xyzPoints,N*M,3);
% reshape to (N*M)x3
J1l = reshape(img,N*M,3);
% reshape to (N*M)x3
pcl = pcl(indI,:);
% select 3D points that are on resolution grid
J1l = J1l(indI,:);
% select pixels that are on the resolution grid
%% remove the unreliable points and the associated triangles
ind_unreliable = find(unreliable(indI));% get the linear indices of
unreliable 3D points
imem = ismember(TRI(:),ind_unreliable); % find indices of references to
unreliable points
[ir,~] = ind2sub(size(TRI),find(imem)); % get the indices of rows with
refs to unreliable points.
TRI(ir,:) = [];
% dispose them
iused = unique(TRI(:));
% find the ind's of vertices that are in use
used = zeros(length(pcl),1);
% pre-allocate
used(iused) = 1;
% create a map of used vertices
map2used = cumsum(used);
% conversion table from indices of old vertices to the new one
pcl = pcl(iused,:);
% remove the unused vertices
J1l = J1l(iused,:);
TRI = map2used(TRI);
% update the ind's of vertices
%% create the 3D mesh
TR = triangulation(TRI,double(pcl));
%% visualize
figure;
TM = trisurf(TR);
set(TM,'FaceVertexCData',J1l);
set(TM,'Facecolor','interp');
% set(TM,'FaceColor','red');
set(TM,'EdgeColor','none');
xlabel('x (mm)')
ylabel('y (mm)')
zlabel('z (mm)')
axis([-250 250 -250 250 400 900])
set(gca,'xdir','reverse')
set(gca,'zdir','reverse')
daspect([1,1,1])
axis tight % create the object % plot the mesh % set colors to input
image % if you want a colored surface % suppress the edges
end

```