УДК 681.518:621.865.8
DOI https://doi.org/10.32782/2663-5941/2025.6.2/07

Біганський Б.М.
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

Ковалюк Д.О.
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

АЛГОРИТМ ВІЗУАЛЬНО-ІНЕРЦІЙНОЇ ОДОМЕТРІЇ З ВИКОРИСТАННЯМ ГЕОМЕТРИЧНО-ОРІЄНТОВАНОЇ НЕЙРОННОЇ МЕРЕЖІ

У статті розв'язано актуальну задачу підвищення стійкості та точності систем візуально-інерційної одометрії (VIO) для навігації автономних роботів. Основний акцент зроблено на роботі в середовищах із низькою текстурною насиченістю (наприклад, однотонні стіни, індустріальні коридори), де традиційні методи на основі розріджених ознак, такі як VINS-Mono, втрачають ефективність через неможливість стабільного відстеження характерних точок.

Запропоновано новітню архітектуру системи Dense VIO, що базується на тісній інтеграції (tightly-coupled) геометрично-орієнтованої нейронної мережі та методів ймовірнісної оптимізації. Ключовою інновацією є використання моделі Microsoft MoGe для генерації щільних метричних карт глибини безпосередньо з монокулярного відеопотоку. Це дозволяє відмовитися від класичного детектування кутових ознак на користь використання повної геометричної інформації про сцену.

Детально описано математичну модель системи, реалізовану в межах єдиного фактор-графа. Візуальний фактор побудовано на основі реєстрації послідовних хмар точок із використанням алгоритму Point-to-Plane ICP, що мінімізує геометричну похибку вирівнювання поверхонь. Фактор інерційної преінтеграції забезпечує об'єднання високочастотних даних IMU з візуальними вимірюваннями, гарантуючи метричну коректність масштабу, точну оцінку зміщень (biases) акселерометра й гіроскопа та спостережуваність вектора гравітації.

Програмну реалізацію виконано із залученням бібліотек SymForce, а візуалізацію – через Rerun. Проведено комплексний порівняльний аналіз із VINS-Mono, BASALT та DM-VIO на наборах даних зі складними сценаріями, включаючи різкі зміни освітлення. Оцінка абсолютної похибки траєкторії (ATE) засвідчила, що запропонований алгоритм забезпечує менший дрейф та стабільніше утримання масштабу в безтекстурних зонах, підтверджуючи переваги щільних методів.

Ключові слова: *мобільні роботи, керування, автономна навігація, візуально-інерційна одометрія, SLAM, нейронні мережі, MoGe, Computer Vision.*

Постановка проблеми. Точна та надійна оцінка власного руху є фундаментальною передумовою для автономної навігації мобільних роботів, особливо в середовищах, де сигнал супутникової навігації недоступний або ненадійний. Протягом останнього десятиліття візуально-інерційна одометрія (Visual-Inertial Odometry – VIO) затвердилася як стандарт для вирішення цієї задачі завдяки поєднання даних сенсорів: камери забезпечують повну інформацію про структуру сцени, тоді як інерційні вимірювальні модулі (IMU) надають високочастотні дані про динаміку руху та метричний масштаб.

Аналіз останніх досліджень і публікацій. Розглянемо існуючі підходи до візуальної та візуально-інерційної одометрії, поділяючи їх на три основні категорії: традиційні геометричні методи, методи на основі глибокого навчання та сучасні підходи до оцінки глибини.

1. *Традиційні геометричні методи.* Ця категорія охоплює класичні алгоритми, які використовують геометрію кількох виглядів. Їх можна розділити на два підкласи: розріджені та щільні. Розріджені Візуально-Інерційні Системи (Sparse Visual-Inertial Systems), що базуються на розріджених ознаках, домінували в галузі навігації робо-

тів завдяки їхній обчислювальній ефективності. Алгоритми VINS-Mono [1, с. 1003-1017] та OKVIS [2, с. 314-334] використовують щільно пов'язану нелінійну оптимізацію, об'єднуючи вимірювання IMU та помилку репроєкції точкових ознак у ковзному вікні. ORB-SLAM3 [3, с. 1874-1890] розширив цей підхід, додавши можливість повторного використання карти (map reuse) та замикання циклів. Головною перевагою цих методів є здатність працювати в реальному часі на обмежених обчислювальних ресурсах (CPU). Однак, їхня залежність від висококонтрастних точок (кутів, країв) є критичним недоліком. У середовищах з низькою текстурою (наприклад, білі стіни, коридори) або при розмитті зображення детектори ознак (такі як FAST або Shi-Tomasi) не можуть знайти достатню кількість стабільних точок, що призводить до розбіжності оцінки стану та дрейфу траєкторії. Крім того, розріджена хмара точок, яку вони генерують, несе мало інформації для задач навігації, таких як уникнення перешкод.

Щільні методи (Dense SLAM), на відміну від розріджених, намагаються використовувати інформацію з усіх пікселів зображення. DTAM [4, с. 2320-2327] був одним із перших, хто запропонував мінімізувати фотометричну помилку для реконструкції щільної карти в реальному часі, але вимагав потужних GPU. LSD-SLAM [5, с. 834-849] запропонував компроміс, оцінюючи глибину лише в зонах високого градієнта інтенсивності (semi-dense), що дозволило роботу на CPU. Основним викликом для монокулярних щільних методів є спостережуваність масштабу. Без інтеграції з інерційними датчиками або стерео-парою, такі системи страждають від значного дрейфу масштабу (scale drift), коли карта помилково "стискається" або "розширюється" з часом. Хоча прямі методи (direct methods) є більш стійкими до низької текстури, вони чутливі до змін освітлення, оскільки порушується припущення про постійність яскравості.

2. *VIO на основі глибокого навчання (Learning-based VIO)*. З розвитком нейронних мереж з'явилися методи такі як DeepVO [6, с. 2043-2050] та TartanVO [7, с. 1761-1772], які визначають позу камери безпосередньо із послідовності зображень, оминаючи явну геометричну модель. Droid-SLAM [8, с. 16558–16569] досяг вражаючих результатів, поєднуючи диференційований рекурентний оновлювач (GRU) з щільним оптичним потоком. Такі методи демонструють високу стійкість до складних умов (динамічні об'єкти, розмиття). Проте, вони часто діють як "чорні скриньки", що ускладнює гарантування фізичної коректності траєкторії (наприклад, узгодженість з гравітацією) та інтеграцію з IMU в імовірнісному контексті. Більшість з них також вимагають величезних датасетів для тренування і можуть погано генералізуватися на нові типи середовищ.

3. *Монокулярна оцінка глибини (Monocular Depth Estimation)*. Ключовим компонентом такого підходу є оцінка глибини з одного кадру. Ранні моделі, такі як MiDaS [9, с. 1623-1637], забезпечували високоякісну відносну глибину, але не мали метричного масштабу, що робило їх непридатними для прямої метричної одометрії. Новітні архітектури, зокрема ZoeDepth [10, с. 19432–19442] та Microsoft MoGe [11, с. 5261-5271], зробили прорив у метричній оцінці глибини (Metric Depth Estimation). Цей підхід на сьогодні є надзвичайно перспективним, оскільки MoGe завдяки його унікальній здатності генерувати геометрично узгоджені карти, які зберігають чіткі межі об'єктів та планарні структури, критично важливі для алгоритмів реєстрації хмар точок (Point-to-Plane ICP). На відміну від MiDaS, MoGe тренується з урахуванням геометричних пріоритетів, що дозволяє отримувати точніші нормалі поверхонь та зменшувати ефект "викривлення простору" на краях зображення.

Таким чином, незважаючи на значний прогрес, досягнутий такими алгоритмами, як VINS-Mono, ORB-SLAM3, Basalt та OpenVINS, більшість з них покладаються на підхід, що базується на розріджених ознаках (sparse feature-based methods). Ці методи відстежують кути та висококонтрастні точки (keypoints) для оцінки геометричних обмежень. Хоча цей підхід є обчислювально ефективним, він демонструє критичну вразливість у середовищах з низькою текстурою (наприклад, коридори з білими стінами, індустріальні приміщення) або в умовах змінного освітлення та розмиття зображення (motion blur). Втрата візуальних ознак у таких сценаріях неминуче призводить до швидкого дрейфу траєкторії або повної втрати трекінгу.

Методи "щільної" одометрії (dense odometry), які використовують усю інформацію пікселів зображення, вимагають значних обчислювальних ресурсів і страждають від невизначеності масштабу при використанні монокулярної камери. З іншого боку, сучасні підходи на основі глибокого навчання часто діють як "чорні скриньки", не гарантуючи фізичної коректності траєкторії та генералізації в нових середовищах.

Постановка завдання. Метою роботи є дослідження гібридної архітектури Dense VIO, яка

поєднує можливості сучасних генеративних нейромереж для оцінки геометрії з надійністю класичної ймовірнісної оптимізації. Такий підхід використовує модель Microsoft MoGe для генерації метрично-узгоджених карт глибини з одного RGB зображення, усуваючи потребу в пошуку розріджених точок. Замість мінімізації помилки репроєкції, формулюємо задачу оцінки руху як щільну реєстрацію хмар точок (Point-to-Plane ICP), яка тісно пов'язана з даними IMU в межах фактор-графа оптимізації. Це дозволяє системі використовувати геометричну структуру сцени (площини стін, підлоги) навіть за відсутності текстури, тоді як IMU-преінтеграція забезпечує спостережуваність гравітації, масштаб та корекцію збурень.

Для досягнення поставленої мети в роботі розв'язані наступні задачі:

– Архітектура системи: розробка пайплайну, що інтегрує вихід нейромережі MoGe безпосередньо у бекенд нелінійної оптимізації.

– Математичне забезпечення: Введення щільного візуального фактору на основі ICP, що дозволяє сумісну оптимізацію поз камери, швидкостей та зміщень IMU.

– Валідація: програмна реалізація алгоритму та експериментальне підтвердження результатів його роботи.

Виклад основного матеріалу. Основні ідея полягає у відмові від розріджених ознак (де рівняння базуються на піксельних помилках) на користь щільної геометрії. У фактор-графі оптимізується стан системи у моменти часу i. Стан x_i у момент t_i:

$$x_i = \left[R_i, p_i, v_i, b_i^a, b_i^g \right] \quad (1)$$

де

– $R_i \in SO(3)$: Орієнтація тіла (IMU) відносно світу.

– $p_i \in \mathbb{R}^3$: Позиція IMU у світовій системі.

– $v_i \in \mathbb{R}^3$: Швидкість IMU у світовій системі.

– $b_i^a,\ b_i^g \in \mathbb{R}^3$: Зміщення акселерометра та гіроскопа.

Архітектура системи наведена на рис. 1.

Загальна функція витрат (Cost Function) задачі мінімізації:

$$J(x) = \sum_{i,j \in K} \left\| r_I\left(z_{ij}, x_i, x_j\right) \right\|_{\Sigma_I}^2 + \sum_{i,j \in V} \left\| r_V\left(T_{ij}^{icp}, T_i, T_j\right) \right\|_{\Sigma_V}^2 + \left\| r_{prior} \right\|_{\Sigma_P}^2 \quad (2)$$

У рівнянні (2) є три фактори, це IMU Factor, Dense Visual Factor та Prior Factor. Розберемо кожний із них детальніше.

Інерційний фактор (IMU Factor). Це сума всіх помилок, які виникають через невідповід-

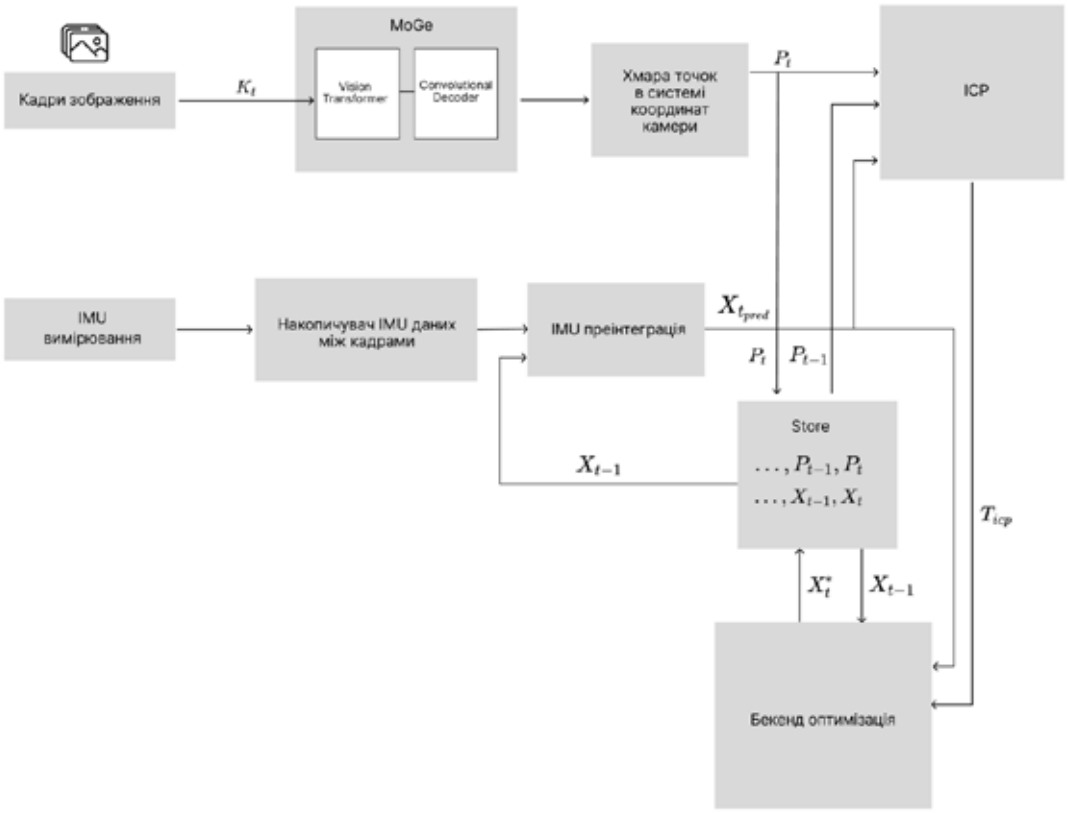

Рис. 1. Блокова архітектура системи. K_t – поточний кадр зображення (в момент часу t); P_t та P_{t-1} – хмари точок з поточного та попереднього кадрів відповідно; X_t та X_{t-1} – стани системи поточного попереднього кадрів відповідно; $X_{t_{pred}}$ – преінтегрований стан системи з даних IMU; T_{icp} – відносна поза між двома кадрами, обчислена за допомогою ICP; X_t^* – стан системи після оптимізації

ність руху робота даним IMU між відповідними кадрами.

$$\sum_{i,j\in K}\left\|r_I\left(z_{ij},x_i,\mathbb{x}_j\right)\right\|^2_{£_I} \quad (3)$$

де,

– x_i, x_j – вектори стану в моменти часу t_i та t_j.

– z_{ij} – преінтегроване вимірювання від IMU. Включає $\Delta R_{ij}, \Delta v_{ij}, \Delta p_{ij}$. ΔR_{ij} – поворот відносно кадру в момент часу i до моменту часу j. Δv_{ij} – відносна зміна швидкості. Δp_{ij} – відносна зміна положення.

– $£_I$ – матриця коваріації, що описує шум IMU.

У сучасних алгоритмах візуально-інерціальної одометрії (VIO) ключовим етапом є формування фактора преінтеграції. Замість того, щоб інтегрувати дані акселерометра та гіроскопа у глобальній системі координат (що вимагало б повторного обчислення при кожній зміні оцінки стану), ми інтегруємо їх у локальній системі координат попереднього кадру.

Наведені нижче рівняння описують модель вимірювань, яка пов'язує попередньо інтегровані дані IMU з оцінюваними станами системи [12, с. 1-20]:

$$\Delta R_{ij} = R_i^T R_j Exp\left(\delta\phi_{ij}\right)$$

$$\Delta v_{ij} = R_i^T\left(v_j - v_i - g\Delta t_{ij}\right) + \delta v_{ij}$$

$$\Delta p_{ij} = R_i^T\left(p_j - p_i - v_i\Delta t_{ij} - \frac{1}{2}g\Delta t_{ij}^2\right) + \delta p_{ij} \quad (4)$$

де:

– ΔR_{ij} – преінтегроване вимірювання обертання (зміни орієнтації).

– Δv_{ij} – преінтегроване вимірювання зміни швидкості.

– Δp_{ij} – преінтегроване вимірювання зміни позиції.

– R_i, R_j – матриці обертання (Rotation matrices) у моменти часу i та j відносно глобальної системи координат.

– p_i, p_j – вектори позиції у глобальній системі координат.

– v_i, v_j – вектори швидкості у глобальній системі координат.

– g – вектор гравітації у глобальній системі координат

– Δt_{ij} – інтервал часу між кадрами i та j.

– $Exp\left(\delta\phi_{ij}\right)$ – Експоненційне відображення для помилки орієнтації. Оскільки обертання лежать на многовиді ($SO(3)$), ми не можемо просто додати шум; ми використовуємо експоненційну карту для представлення малого відхилення.

– $\delta\phi_{ij}, v_{ij}, \delta p_{ij}$ – вектори шуму (random noise vectors) для орієнтації, швидкості та позиції відповідно. Вони моделюються як гаусовський білий шум.

Враховуючи попередньо інтегровану модель вимірювання можна легко представити залишкові помилки для фактору IMU[12]:

$$r_{\Delta R_{ij}} = Log\left(\left(\Delta R_{ij}\left(b_i^g\right)Exp\left(\frac{\partial\Delta R_{ij}}{\partial b^g}\delta b^g\right)\right)^T R_i^T R_j\right)$$

$$r_{\Delta v_{ij}} = R_i^T\left(v_j - v_i - g\Delta t_{ij}\right) - \left[\Delta v_{ij}\left(b_i^g, b_i^a\right) + \frac{\partial\Delta v_{ij}}{\partial b^g}\delta b^g + \frac{\partial\Delta v_{ij}}{\partial b^a}\delta b^a\right]$$

$$r_{\Delta p_{ij}} = R_i^T\left(p_j - p_i - v_i\Delta t_{ij} - \frac{1}{2}g\Delta t_{ij}^2\right) - \left[\Delta p_{ij}\left(b_i^g, b_i^a\right) + \frac{\partial\Delta \boldsymbol{p}_{ij}}{\partial b^g}\delta b^g + \frac{\partial\Delta p_{ij}}{\partial b^a}\delta b^a\right] \quad (5)$$

де:

– $r_{\Delta R_{ij}}, r_{\Delta v_{ij}}, r_{\Delta p_{ij}}$ – залишки обертання, швидкості та позиції відповідно.

– $\frac{\partial\Delta R_{ij}}{\partial b^g}, \frac{\partial\Delta v_{ij}}{\partial b^g}, \frac{\partial\Delta v_{ij}}{\partial b^a}, \frac{\partial\Delta p_{ij}}{\partial b^g}, \frac{\partial\Delta p_{ij}}{\partial b^a}$ – Якобіани преінтегрованих вимірювань по відношенню до зміщень гіроскопа та акселерометра.

– $\delta b^g, \delta b^a$ – відхилення поточного зміщення від номінального.

– $Log(\cdot)$ – Логарифмічне відображення (Log map), яке переводить помилку обертання з простору матриць $SO(3)$ у векторний простір дотичної алгебри $\mathbb{R}^3$. Це дозволяє мінімізувати "кут повороту" помилки.

Візуальний фактор (Dense Visual Factor). Це сума помилок неузгодженості між тим, як робот рухається згідно з графом та преінтегрованими позами, і тим, як зміщуються хмари точок (MoGe + ICP) між кадрами.

$$\sum_{i,j\in V}\left\|r_V\left(T_{ij}^{icp}, T_i, T_j\right)\right\|^2_{\Sigma_V} \quad (6)$$

де:

– T_{ij}^{icp} – відносна поза, що є результатом роботи алгоритму ICP, який наклав хмару точок з кадру i на хмару точок кадру j. Це є вимірювання (Measurement) і це константа для оптимізатора.

– T_i – поза камери в момент часу i в групі Лі $SE(3)$.

– T_j – поза камери в момент часу j в групі Лі $SE(3)$.

Функція нев'язки r_V визначається наступним чином:

$$r_V\left(T_{ij}^{icp}, T_i, T_j\right) = Log\left(\left(T_{ij}^{icp}\right)^{-1}\cdot\left(T_i^{-1}T_j\right)\right)^{\vee} \quad (7)$$

розберемо детальніше:

– $T_i^{-1}T_j$ – беремо поточні оцінки поз T_i та T_j і обчислюємо відносний рух між ними.

– $\left(T_{ij}^{icp}\right)^{-1}\cdot\left(T_i^{-1}T_j\right)$ – множимо обернену матрицю вимірювання на передбачення. Якщо передбачення ідеально збігається з ICP, то результат буде одиничною матрицею.

– $Log(\ldots)^{\vee}$ – Матриця помилки – це незручно, потрібен вектор чисел, щоб піднести його до квадрата. Тому операція Log (логарифмічне відображення) переносить з групи Лі $SE(3)$ (матриці)

в алгебру Лі $\mathfrak{se}(3)$ (дотичний простір). Оператор "vee" (V) перетворює це на вектор з 6 чисел: $\left[rot_x, rot_y, rot_z, trans_x, trans_y, trans_z\right]^T$.

– Σ_V – матриця коваріації. Вона є адаптивною, в залежності від якості хмари точок і накладання ICP.

Апріорний Фактор (Prior Factor). Це різниця між поточним станом x_j робота і фіксованим значенням, що є початковим положенням, звідки відбувся рух x_{start} (зазвичай – це позиція 0, 0, 0).

$$r_{prior} = x_j - x_{start} \tag{8}$$

де:

– x_j – це змінні, які оптимізатор може змінювати (R, p, v, b_a, b_g).

– x_{start} – це жорстко задані початкові умови. В оптимізаторі ці значення як константи.

Запуск системи проводився на платформі Apple Silicon (чип M3). Для інференсу нейромережі MoGe використано бекенд Metal Performance Shaders (MPS). Це дозволяє виконувати обчислення PyTorch безпосередньо на графічних ядрах чипа M3, використовуючи архітектуру об'єднаної пам'яті (Unified Memory Architecture).

Слід зазначити, що архітектура трансформерів (ViT), яка лежить в основі MoGe, є обчислювально вимогливою. На чипі M3 (через обмеження пропускної здатності MPS для певних операцій трансформерів) система досягає частоти оновлення одометрії близько 5–10 Гц. Для застосувань, що вимагають вищої динаміки, рекомендовано використання платформ з підтримкою NVIDIA CUDA (наприклад, NVIDIA Jetson Orin або дискретні GPU серії RTX). Використання бібліотек TensorRT та cuBLAS дозволило б прискорити інференс MoGe та паралельне виконання ICP, потенційно подвоївши частоту кадрів (до 30+ Гц). Тим не менш, результати на M3 підтверджують доцільність запропонованого підходу на енергоефективних Edge-пристроях без дискретних відеокарт.

Параметри системи:

– розмір вокселя (Voxel Grid): 0.05 м. Це компроміс, що зменшує хмару точок з ~300k до ~5k точок, дозволяючи ICP збігатися швидше.

– вікно оптимізації розміром N=20 ключових кадрів із маргіналізацією старіших станів;

– часові мітки камери та IMU програмно синхронізовані, а вхідні дані інтерполюються для точного співставлення моментів часу.

Для оцінки точності запропонованого алгоритму Dense MoGe-VIO, виконаємо порівняння з провідними алгоритмами візуально-інерційної одометрії, що вважаються стандартом (state-of-the-art) у галузі. У якості базових методів було обрано: VINS-Mono, ORB-SLAM3, DM-VIO [13, с. 1408-1415] та BASALT [14, с. 422-429]. Для забезпечення рівних умов проведення експерименту, оцінюємо виключно якість одометрії. Тому в усіх порівнюваних методах (VINS-Mono, ORB-SLAM3, BASALT) були примусово вимкнені модулі замикання циклів (Global Loop Closure) та релокалізації. Це дозволяє ізолювати та виміряти накопичувальний дрейф кожного алгоритму без корекцій від розпізнавання вже відвіданих місць. В якості основної метрики оцінки використовується RMSE ATE (Root Mean Square Error of Absolute Trajectory Error) – середньоквадратична похибка абсолютної траєкторії після вирівнювання за 6 ступенями вільності.

Таблиця 1 демонструє результати на стандартному датасеті EuRoC MAV [15, с. 1157-1163] (послідовності Machine Hall). Таблиця 2 представляє результати на спеціалізованому тестовому наборі (Low-Texture Dataset), записаному в умовах засвітлення з однорідними текстурами [16, с. 5495-5502]. Виходячи з отриманих результатів на цих сценаріях традиційні методи демонструють значне зростання похибки і часткову втрату трекінгу через нестачу візуальних ознак, тоді як запропонований алгоритм зберігає стабільність завдяки використанню щільної геометрії від MoGe. У таблицях найкращі результати виділені жирним шрифтом, підкреслення – другий найкращий результат.

На рис. 2 наведено візуальну траєкторію, побудовану системами BASALT, DM-VIO та запропоновану MoGe-VIO для датасету EuRoC послідовності MH_05_difficult. Запропонований алгоритм (MoGe-VIO, зелена лінія) демонструє траєкторію, узгоджену з системами Basalt (синя лінія) та DM-VIO (червона лінія), у порівнянні з істинною траєкторією (Ground Truth, пунктир).

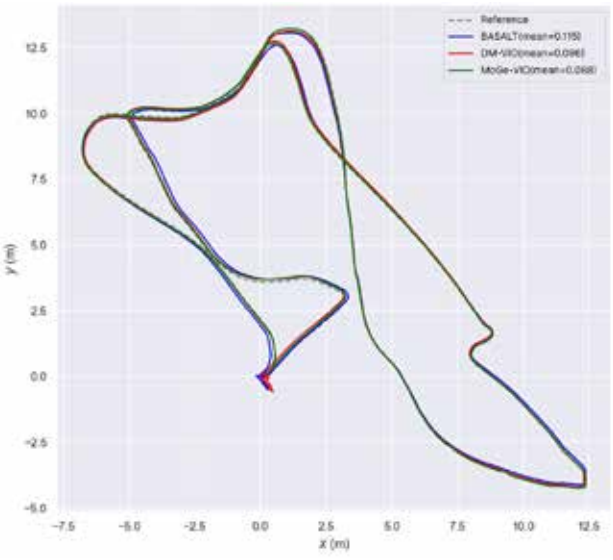

Рис. 2. Порівняння траєкторій на послідовності MH_05_difficult з набору даних EuRoC

Таблиця 1

Порівняння ATE(m) та середнього часу обробки між різними системами на наборі даних EuRoC

	ATE(m)				
	VINS mono	**ORB-SLAM3 mono**	**DM-VIO mono**	**BASALT mono**	**MoGe-VIO mono**
MH1	0.15	0.053	0.065	0.08	**0.012**
MH2	0.16	0.079	0.044	0.07	**0.027**
MH3	0.23	**0.056**	0.097	0.07	0.08
MH4	0.31	0.115	0.104	0.13	**0.053**
MH5	0.31	0.107	0.096	0.115	**0.088**
Час обробки(ms)	16.7	116	46	12	130

Таблиця 2

Порівняння ATE(m) між різними системами на наборі даних TUM-VI

	ATE(m)				
	VINS mono	**ORB-SLAM3 mono**	**DM-VIO mono**	**BASALT mono**	**MoGe-VIO mono**
outdoors1	74.92	111.21	123.22	255.04	**42.45**
outdoors2	133.34	19.079	12.43	64.64	**3.59**
outdoors3	36.99	-	8.897	38.17	**2.16**
outdoors5	130.31	16.87	7.14	7.57	1.91

Висновки. У роботі запропоновано нову архітектуру монокулярної візуально-інерційної одометрії, що використовує щільні карти глибини, згенеровані нейронною мережею MoGe, у поєднанні з імовірнісною оптимізацією на основі фактор-графа. Запропонований підхід дозволяє подолати фундаментальні обмеження традиційних розріджених методів, замінюючи ненадійний пошук кутових ознак на реєстрацію щільної геометрії сцени.

Результати комп'ютерного експерименту підтвердили ефективність запропонованого алгоритму:

– Точність: на стандартному наборі даних EuRoC MAV система продемонструвала здебільшого вищу точність серед порівнюваних методів, досягаючи середньоквадратичної похибки (RMSE ATE) на рівні 0.027–0.088 м у складних динамічних сценаріях (MH_02, MH_05). Це перевершує результати VINS-Mono та є конкурентним, а часто й кращим, ніж показники ORB-SLAM3 та BASALT.

– Стійкість у складних умовах: вирішальна перевага проявилася на датасеті TUM-VI (Outdoors), що характеризується змінним освітленням та однорідними текстурами. У сценаріях, де класичні методи демонстрували дрейф (до 133 м для VINS-Mono та 64 м для BASALT) або повну втрату трекінгу, Dense MoGe-VIO зберіг стабільність із похибкою 1.91–3.59 м. Це доводить, що використання геометричних пріоритетів від нейромережі є критичним для навігації у середовищах бідних на текстури.

Слід зазначити, що підвищена точність та надійність запропонованого підходу досягається ціною обчислювальних витрат: час обробки кадру є вищим порівняно з легкими розрідженими методами (BASALT: ~12 мс vs. MoGe-VIO: ~130 мс).

Напрямки подальших досліджень включають:

– Оптимізацію інференсу нейромережі шляхом перенесення системи на платформи з підтримкою NVIDIA CUDA та використання TensorRT для досягнення частоти 30+ Гц.

– Інтеграцію модуля замикання циклів (Loop Closure) на основі глобальних дескрипторів для усунення залишкового дрейфу на довгих траєкторіях.

– Дослідження адаптивної зміни роздільної здатності воксельної сітки для балансування між швидкістю ICP та точністю в реальному часі.

Таким чином, запропонований підзід Dense MoGe-VIO є перспективним кроком до створення повністю автономних систем навігації, здатних оперувати в неструктурованих та візуально складних середовищах.

Список літератури:

1. Qin T., Li P., Shen S. VINS-Mono: A robust and versatile monocular visual-inertial state estimator. *IEEE Transactions on Robotics*. 2018. Vol. 34, No. 4. Pp. 1003–1017. https://doi.org/10.48550/arXiv.1708.03852

2. Leutenegger S., Lynen S., Bosse M., Siegwart R., Furgale P. OKVIS: Keyframe-based visual-inertial odometry using nonlinear optimization. *The International Journal of Robotics Research*. 2015. Vol. 34, No. 3. Pp. 314–334. https://doi.org/10.1177/0278364914554813

3. Campos C., Elvira R., Rodríguez J. J. G., Montiel J. M. M., Tardós J. D. ORB-SLAM3: An accurate open-source library for visual, visual–inertial, and multimap SLAM. *IEEE Transactions on Robotics.* 2021. Vol. 37, No. 6. Pp. 1874–1890. https://doi.org/10.48550/arXiv.2007.11898

4. Newcombe R. A., Lovegrove S. J., Davison A. J. DTAM: Dense tracking and mapping in real-time. *2011 International Conference on Computer Vision (ICCV)*. 2011. Pp. 2320–2327. https://doi.org/10.1109/ICCV.2011.6126513

5. Engel J., Schöps T., Cremers D. LSD-SLAM: Large-scale direct monocular SLAM. Computer Vision – ECCV 2014. Cham, 2014. Pp. 834–849.

6. Wang S., Clark R., Wen H., Trigoni N. DeepVO: Towards end-to-end visual odometry with deep recurrent convolutional neural networks. *2017 IEEE International Conference on Robotics and Automation (ICRA)*. 2017. Pp. 2043–2050. https://doi.org/10.48550/arXiv.1709.08429

7. Wang W., Hu Y., Scherer S. TartanVO: A generalizable learning-based visual odometry. *Conference on Robot Learning*. PMLR, 2021. Pp. 1761–1772. https://doi.org/10.48550/arXiv.2011.00359

8. Teed Z., Deng J. DROID-SLAM: Deep Visual SLAM for Monocular, Stereo, and RGB-D Cameras. *Advances in Neural Information Processing Systems*. 2021. Vol. 34. Pp. 16558–16569. https://doi.org/10.48550/arXiv.2108.10869

9. Ranftl R., Lasinger K., Hafner D., Schindler K., Koltun V. Towards robust monocular depth estimation for mixing 3D movies. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2020. Vol. 44, No. 3. Pp. 1623–1637. https://doi.org/10.1109/TPAMI.2020.3019967

10. Bhat S. F., Birkl R., Wofk D., Wonka P., Müller M. ZoeDepth: Zero-shot transfer by combining relative and metric depth. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023. Pp. 19432–19442. https://doi.org/10.48550/arXiv.2302.12288

11. Wang R., Xu S., Dai C., Xiang J., Deng Y., Tong X., Yang J. Moge: Unlocking accurate monocular geometry estimation for open-domain images with optimal training supervision. *Proceedings of the Computer Vision and Pattern Recognition Conference.* 2025. Pp. 5261–5271. https://doi.org/10.48550/arXiv.2410.19115

12. Forster C., Carlone L., Dellaert F., Scaramuzza D. On-Manifold Preintegration for Real-Time Visual-Inertial Odometry. *IEEE Transactions on Robotics.* 2017. Vol. 33, No. 1. Pp. 1–20. https://doi.org/10.48550/arXiv.1512.02363

13. Von Stumberg L., Cremers D. DM-VIO: Delayed Marginalization Visual-Inertial Odometry. *IEEE Robotics and Automation Letters*. 2022. Vol. 7, No. 2. Pp. 1408–1415. https://doi.org/10.1109/LRA.2021.3140129

14. Usenko V., Demmel N., Schubert D., Stückler J., Cremers D. Visual-Inertial Mapping with Non-Linear Factor Recovery. *IEEE Robotics and Automation Letters*. 2020. Vol. 5, No. 2. Pp. 422–429. https://doi.org/10.48550/arXiv.1904.06504

15. Burri M., Nikolic J., Gohl P., Schneider T., Rehder J., Omari S., Achtelik M., Siegwart R. The EuRoC micro aerial vehicle datasets. *The International Journal of Robotics Research*. 2016. Vol. 35, No. 10. Pp. 1157–1163. https://doi.org/10.1177/0278364915620033

16. Klenk S., Chui J., Demmel N., Cremers D. TUM-VIE: The TUM Stereo Visual-Inertial Event Dataset. *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS).* 2021. Pp. 5495–5502. https://doi.org/10.48550/arXiv.2108.07329

Bihanskyi B.M., Kovaliuk D.O. VISUAL-INERTIAL ODOMETRY ALGORITHM USING A GEOMETRY-AWARE NEURAL NETWORK

This article addresses the pressing problem of enhancing the robustness and accuracy of visual-inertial odometry (VIO) systems for autonomous robot navigation. The primary focus is on operation in low-texture environments (e.g., monochromatic walls, industrial corridors), where traditional sparse feature-based methods, such as VINS-Mono, become ineffective due to the inability to stably track feature points.

A novel architecture for a Dense VIO system is proposed, based on the tightly-coupled integration of a geometry-aware neural network and probabilistic optimization methods. The key innovation lies in the utilization of the Microsoft MoGe model to generate dense metric depth maps directly from a monocular video stream. This eliminates the need for classical corner feature detection in favor of leveraging complete geometric information about the scene.

The mathematical model of the system, implemented within a unified factor graph framework, is described in detail. The visual factor is formulated based on the registration of sequential point clouds using the Point-to-Plane ICP algorithm, which minimizes the geometric surface alignment error. The inertial preintegration factor ensures the fusion of high-frequency IMU data with visual measurements, guaranteeing metric scale consistency, accurate estimation of accelerometer and gyroscope biases, and observability of the gravity vector.

The software implementation was performed using the SymForce library, with visualization provided via Rerun. A comprehensive comparative analysis was conducted against VINS-Mono, BASALT, and DM-VIO on datasets featuring complex scenarios, including abrupt lighting changes. Evaluation of the Absolute Trajectory Error (ATE) demonstrated that the proposed algorithm ensures reduced drift and more stable scale maintenance in textureless zones, confirming the advantages of dense methods.

Key words: *mobile robots, control, autonomous navigation, visual-inertial odometry, SLAM, neural networks, MoGe, Computer Vision.*

Дата надходження статті: 23.11.2025
Дата прийняття статті: 10.12.2025
Опубліковано: 30.12.2025