

# Lidar 지도를 이용한 시각적 위치 추정 방법론 비교

양 찬 옥 · 허 건 수\*

한양대학교 미래자동차공학과

## Comparison of Loosely and Tightly Coupled Visual Inertial Odometry Using Lidar Prior Map

Chanuk Yang · Kunsoo Huh\*

Department of Automotive Engineering, Hanyang University, Seoul 04763, Korea  
(Received 21 October 2024 / Revised 28 October 2024 / Accepted 30 October 2024)

**Abstract** : This paper explored visual inertial odometry(VIO) using a Lidar prior map, focused on comparing loosely coupled and tightly coupled approaches. The study investigated how the integration of visual and inertial data, combined with Lidar information, enhances localization performance, particularly under varying conditions of feature matching reliability. The research proposed a comprehensive preprocessing pipeline, including reference image alignment and depth map generation, to support accurate visual relocalization. The integration of stereo cameras, IMU data, and wheel odometry was examined within both VIO frameworks. The methods are tested using a complex urban dataset to evaluate their effectiveness in real-world autonomous driving scenarios. By comparing the performance of each approach, this research is expected to provide valuable insights into their applicability in choosing a method. The performance of the two approaches was assessed under different conditions to determine which offers the most reliable results for VIO tasks.

**Key words** : Visual odometry(시각적 오도메트리), Visual localization(시각적 측위), Lidar map(라이다 지도), Tightly coupled(강 결합), Loosely coupled(약 결합)

### Nomenclature

p : position, m  
v : velocity, m/s  
q : quaternion  
b : bias  
r : residual  
g : gravity, m/s<sup>2</sup>  
z : measurement data  
R : rotation matrix

i : index of keyframe  
k : index of lidar point  
t : time step

### Subscripts

I : IMU frame  
G : global frame  
C : camera frame

## 1. 서론

시각적 오도메트리(Visual Odometry)는 자율주행 자동차에 있어서 각광받고 있는 측위 알고리즘이다. 자동차 업계에서는 측위를 하기 위해 GPS(Global Positioning System) 등 다양한 센서를 활용하고 있지만, 정밀한 GPS는 양산 차량에 탑재하기에는 가격이 비싸 탑재하기 어려울 수 있다. 또한 GPS는 도심에 들어가면 다중 경로 오차(Multi path error)로 인해 위치의 오류가 커질 수 있다. 카메라는 환경 센서 중에서 저렴한 편이고, 실제 일부 양산 차량에서도 장착되어 있다. 자율주행 자동차에서는

\*Corresponding author, E-mail: khuh2@hanyang.ac.kr

This is an Open-Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License(<http://creativecommons.org/licenses/by-nc/3.0>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium provided the original work is properly cited.

측위 알고리즘에서 카메라를 활용하는 연구가 많이 이루어져 있고, 최근 연구는 실시간으로 처리 및 고도화 하는 연구가 주를 이루고 있다.

정밀한 시각적 오도메트리는 자율주행 자동차에서 반드시 필요한 요건이고, 이를 위해 다양한 시각적 재방문 (Visual relocalization) 연구가 진행되고 있다. ORB SLAM<sup>1)</sup>과 VINS Mono<sup>2)</sup>는 대표적인 SLAM(Simultaneous Localization and Mapping) 알고리즘으로, 이들은 모두 시각적 재방문 추정을 통한 정밀한 위치 추정을 가능하게 한다. 최근에는 딥러닝 기반의 시각적 재방문 연구도 활발히 이루어지고 있다. 예를 들어 Patch NetVLAD<sup>3)</sup>는 딥러닝을 활용하여 이미지 두 장을 통해 위치를 추정하고, Back to the Feature<sup>4)</sup>는 특징 점(Feature point) 추출 및 매칭을 통해 위치를 추론하는 방식을 제시한다. SLAM 알고리즘에서는 전통적인 방식으로 DBow(Distributed Bag of Words)와 PnP RANSAC(Perspective n Point with Random Sample Consensus)을 이용한 방법과 딥러닝을 통한 방법이 모두 사용되고 있다.

자율주행에서 주로 활용되는 연구로는 고 정밀 지도(HD map)와 라이다 지도(Lidar prior map)를 활용하는 방법이 있다. 고 정밀 지도는 자율주행을 위한 다양한 정보로 재구성된 지도이지만, 측위에 사용할 수 있는 정보가 상대적으로 적어 위치 추정에 한계를 가진다. 반면 라이다 지도는 더 많은 정보를 포함하고 있어 위치 추정에 더 적합하다. 예를 들어 HDL-SLAM<sup>5)</sup>은 라이다 데이터를 기반으로 하는 정밀한 SLAM 알고리즘 중 하나이고, 이를 통해 만들어진 지도로 위치를 추정할 수 있다. 라이다 센서를 사용하면 위치 추정의 정확도를 크게 향상시킬 수 있지만, 라이다 센서의 가격으로 인해 양산 차량에 적용하는 데 현실적인 어려움이 있다.

이를 해결하기 위한 대안으로 카메라와 라이다 지도를 결합한 연구가 진행되고 있다. 한 가지 접근 방식은 시각적 SLAM에서 생성된 로컬 맵을 라이다 포인트 클라우드 맵과 매칭하는 것이다. Sun 등<sup>6)</sup>은 단안 카메라로 시각적 오도메트리에서 얻은 포인트를 라이다 포인트 클라우드와 정렬하여, 오도메트리에서의 스케일을 추정해 일치시켰다. Zuo 등<sup>7)</sup>은 MSCKF(Multi State Constrained Kalman Filter) 기반 시각적 오도메트리와 라이다 지도를 NDT(Normal Distribution Transform)로 매칭하고, 사전 지도(Prior map)의 부정확성에 대한 민감도를 분석했다. Yabuuchi<sup>8)</sup>, Yabuuchi과 Kato<sup>9)</sup>는 저가형 경량 카메라만을 사용하여 라이다와 지도의 매칭을 시도했으며, 실제 데이터를 사용하여 조명 및 계절 변화에 강인한 결과를 보였다. 이들은 Center-line vector map을 통합하여 매칭을 수행하고, 장기간 측위에서도 정확한 위치 추정을 보였

다. Zhang 등<sup>10)</sup>은 시멘틱 일관성을 사용하여 평면과 점간 ICP(Iterative Closest Point) 매칭을 수행하였고, 어파인(Affine) 변환을 추정하는 방식으로 매칭 전략을 분리했다. Visual SLAM에서 생성된 특징 점 맵과 라이다 포인트 클라우드를 매칭하는 방식은 시각적 오도메트리의 정확도에 크게 의존하며, 부정확한 특징 점 맵은 측위 오차로 이어질 수 있다.

또 다른 접근 방식은 라이다 지도와 카메라 이미지 간의 매칭이다. Wolcott과 Eustice<sup>11)</sup>는 카메라 이미지에서 추출된 다수의 깊이 이미지(Depth image)와 라이다 지도를 이용해 NMI(Normalized Mutual Information)로 코스트 맵을 생성하고, 이를 이용해 매칭을 시도했다. Kim 등<sup>12)</sup>은 스테레오 카메라에서 추출한 깊이 이미지와 라이다 지도의 레인지 이미지(Range image)를 매칭하여 포즈(Pose)를 추정하였다. Caselitz 등<sup>13)</sup>은 Visual SLAM에서 재구성된 포인트와 라이다 포인트 클라우드 맵의 Voxel grid를 매칭하여 측위를 진행하였다. Yu 등<sup>14)</sup>은 이미지의 2D 선과 라이다 지도의 3D 선을 매칭하여 위치를 추정하였다. Zheng 등<sup>15)</sup>은 라인 추적과 매칭을 통해 포즈 추정을 진행하였다.

본 논문에서는 라이다 지도를 활용한 시각적 오도메트리의 성능을 비교하고자 한다. PnP(Perspective n Point) 알고리즘을 통해 얻어진 포즈를 직접 활용하는 약 결합 방식과, 측정 데이터의 집합을 결합하여 최적화하는 강 결합 방식을 비교한다. 또한 라이다 지도를 이용한 시각적 재방문 과정에서 딥러닝을 통한 방식으로 진행하며, 오프라인 전처리를 통해 참조 이미지에 맞는 깊이 이미지와 참조 전역 포즈(Reference global pose)를 추출한다. IMU(Inertial Measurement Unit)를 결합한 방식으로 시각적 오도메트리의 정밀도를 높이고, 이를 최적화하여 위치 추정을 진행한다.

## 2. 라이다 맵 전 처리

라이다 지도를 시각적 오도메트리에 활용하기 위해서는 먼저 지도를 전처리해야 한다. 전 처리 과정은 Fig. 1에 나타나 있으며, 오프라인에서 미리 수행된다. 필요한 데이터는 고 정밀 GPS 데이터, 참조 이미지, 그리고 라이다 전체 맵이다. GPS 데이터에는 포즈 정보가 포함되며, 이를 바탕으로 참조 이미지에 해당하는 참조 포즈와 깊이 이미지를 추출한다.

참조 이미지를 취득한 후, 해당 시간에 가장 유사한 GPS 데이터를 이용해 카메라 포즈를 얻는다. 이 포즈는 라이다 지도를 래스터화(Rasterize)하여 관심 영역의 포인트 클라우드 맵을 생성하는 데 사용된다. 이후 이 포인트 클라우드 맵은 카메라의 내재적 매개변수(Intrinsic

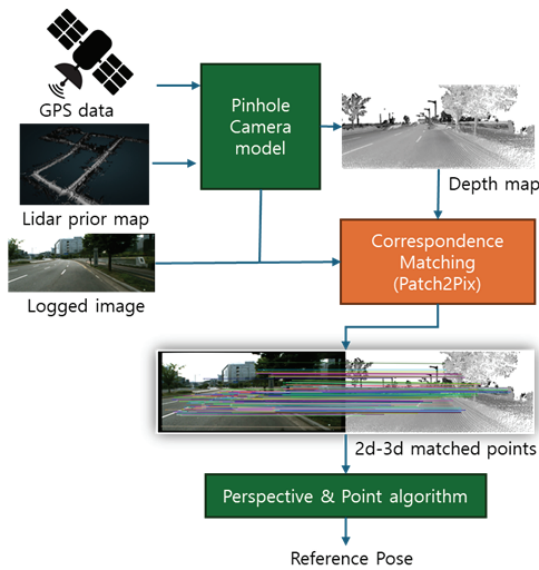


Fig. 1 Flow chart of lidar prior map preprocessing

parameter)에 맞춰 이미지 평면(Image plane)에 사영 되어 깊이를 맵을 생성한다.

그러나 고 정밀 GPS 포즈의 오차로 인해, 깊이 맵과 참조 이미지는 완전히 일치하지 않을 수 있다. 이를 보완하기 위해 PnP 알고리즘을 사용하여 참조 포즈를 정밀하게 추정한다. PnP 알고리즘을 사용하려면 참조 이미지와 깊이 맵 간의 특징점을 추출해야 하지만, 고전적인 방법으로는 특징 점 간의 매칭을 얻기가 어렵다. 이를 해결하기 위해 Superpoint<sup>16)</sup> 알고리즘을 사용하여 특징점을 추출하고 Patch2Pix<sup>17)</sup> 알고리즘으로 대응점 매칭(Correspondence matching)을 진행한다.

이 과정을 통해 3D 포인트를 얻고, 이를 바탕으로 PnP 알고리즘을 이용해 포즈 차이를 계산한다. OpenCV 라이브러리를 사용하여 PnP 알고리즘을 구현하며, 포즈 차이를 계산한 후 Point cloud map을 다시 사영하여 참조 이미지와 오버랩 되는 깊이 이미지를 얻을 수 있다.

### 3. 시각적 관성 오도메트리 알고리즘

시각적 오도메트리 알고리즘에 대한 전체 구성도는 Fig. 2에 있고, 이 알고리즘은 실시간으로 처리된다. 시각적 오도메트리는 크게 3가지 쓰레드로 처리되며, 각각 프론트 엔드(Frontend), 백 엔드(Backend), 그리고 시각적 재방문이 병렬적으로 처리된다. 프론트 엔드는 입력 데이터인 스테레오 이미지, 6DOF IMU, 그리고 휠 오도메트리를 받아 로컬 포즈 및 맵 포인트 위치를 계산하고, 키프레임(Keyframe)을 선정하는 과정이 진행된다. 백 엔드는 저장된 키프레임들을 가져오고, 각각의 데이터 관

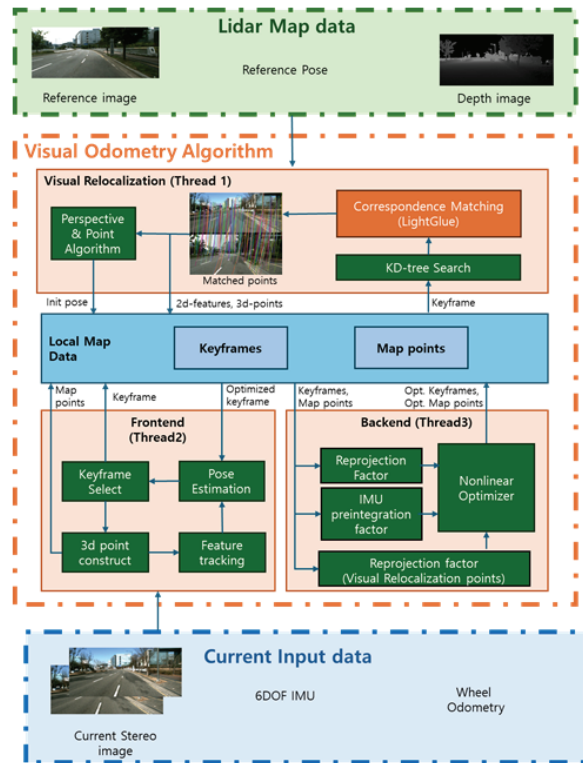


Fig. 2 Schematic of visual inertial odometry algorithm

계를 잔차(Residual) 관계로서 나타내어 비선형 최적화(Nonlinear optimization)를 진행한다. 이를 통해 최적화된 키프레임을 얻을 수 있다. 한편 시각적 재방문은 이전 절에 기술하였던 전 처리된 라이다 지도를 이용해 키프레임과 맵 데이터간 매칭을 진행한다. 이를 통해 매칭된 2D 특징점들과 3D 포인트들, 그리고 초기 포즈를 획득할 수 있다.

#### 3.1 시각적 재방문

추정된 포즈가 주어져 있거나, 초기 포즈를 알고 있다고 가정하고 추정된 포즈가 어느정도 정밀도를 가지고 있다면, 추정된 포즈를 이용해 맵 데이터에서 가장 가까운 참조 포즈와 참조 이미지, 그리고 깊이 이미지를 획득할 수 있다. 가장 가까운 참조 포즈를 찾기 위해서는 KD-Tree 검색을 진행한다.

획득한 참조 이미지는 키프레임 이미지와 대조하여 대응점 매칭 포인트들을 찾는다. 이때 본 연구에서는 딥러닝 기반 알고리즘인 Lightglue<sup>18)</sup>를 통하여 대응점 매칭을 진행한다. 각각 일치하는 매칭 포인트를 찾게 된다면, 참조 이미지에 있는 특징 점 위치들은 깊이 이미지를 통하여 3차원 포인트들을 복원할 수 있고, 참조 포즈를 통하여 전역적 맵 포인트를 획득할 수 있다. 이를 통해 얻

어진 키프레임 이미지에 해당하는 2차원 특징점들과 일대일 관계에 있는 전역적 맵 포인트들은 키프레임에 저장된다.

또한 앞서 얻었던 데이터들은 PnP를 통하여 전역적 포즈를 추정할 수 있다. 같은 관점과 같은 조도, 그리고 변하지 않은 주변 환경 아래에 있는 키프레임 이미지와 참조 이미지 간에서는 정확한 전역적 포즈를 획득할 수 있다. 그러나 두 이미지간 환경이 다르거나, 조도와 관점이 다르다면 오차를 포함한 전역적 포즈를 얻을 수 있다. 따라서 이를 직접 이용하는 대신 키프레임에 저장되어 있는 전역적 포즈를 비교하여 사용한다. 이때 본 연구에서 비교하는 방법은 기본행렬(Fundamental matrix)을 구하여 활용하는 방법으로 진행한다. 전역적 포즈에 대한 각각의 상대 포즈를 구하고, 이를 내재적 파라미터와 함께 활용해 기본행렬을 구한다. 각각 구해진 기본행렬은 매칭된 Feature간 잔차를 구한다. 이를 통해서 잔차가 일정 값으로 넘어가면 이상치로 처리한다. 두 기본행렬 중 이상치가 더 적은 곳이 있다면 해당하는 포즈가 좀더 정확하다는 의미이고, 선정된 포즈를 키프레임의 초기 포즈로 저장한다.

### 3.2 프론트 엔드

프론트 엔드에서는 인접한 시간 순서에 있는 이미지 간의 관계를 통해 상대 포즈, 3D 맵 포인트 및 이에 해당하는 트래킹 된 특징점을 얻을 수 있다. 또한 3차원 맵 포인트는 스테레오 이미지를 통하여 구할 수 있고, 스테레오 이미지 간 특징 점 추출은 FAST 특징 점, 매칭 알고리즘은 Good features to track을 통해 매칭하였다. 매칭된 특징점들은 삼각 측량법(Triangulation algorithm)을 통해 초기 3D 맵 포인트를 만들어낼 수 있다.

현재 시점 이미지의 특징점들은 인접한 시간 순서에 있는 과거 이미지를 특징 점 트래킹을 진행하여 얻을 수 있다. 이는 Open cv 라이브러리<sup>19)</sup>에 있는 Good features to track 알고리즘을 이용한다. 또한 특징점을 좀 더 정확하게 트래킹 하기 위해, 현재 이미지에 초기 3차원 맵 포인트를 휠 오도메트리와 활용하여 사영 시켜 트래킹 알고리즘의 초기값으로 이용한다. 포즈는 트래킹 된 2차원 특징점과 이에 해당하는 3차원 맵 포인트를 이용하여 번들 조정(Bundle adjustment)를 통해 구할 수 있다.

키프레임 선정(Keyframe Selection)은 트래킹 된 특징점의 개수에 따라서 정해진다. 인접한 이미지 간에 특징점 트래킹이 일정 개수 이하로 떨어지게 된다면, 이를 키프레임으로 선정한다. 선정된 키프레임에서 스테레오 이미지를 통한 포인트 구축이 진행되고, 앞서 기술하였던 특징 점 트래킹 및 포즈추정이 순환하여 프론트 엔드

알고리즘이 동작한다.

### 3.3 백 엔드

최적화 과정을 기술하기에 앞서 수식을 정의하고자 한다. 최적화 대상은 키프레임의 포즈들이다. 이를 수식으로 정의하면 다음과 같다.

$$\mathbf{x}_i = [{}^i_G\mathbf{p}^\top, {}^i_G\mathbf{v}^\top, {}^i_G\mathbf{q}^\top, \mathbf{b}_a^\top, \mathbf{b}_g^\top]^\top \quad (1)$$

$$\mathcal{X} = [\mathbf{x}_0^\top, \mathbf{x}_1^\top, \mathbf{x}_2^\top, \dots, \mathbf{x}_N^\top]^\top \quad (2)$$

여기서  ${}^i_G\mathbf{p}^\top$ ,  ${}^i_G\mathbf{v}^\top$ ,  ${}^i_G\mathbf{q}^\top$ ,  $\mathbf{b}_a^\top$ , 그리고  $\mathbf{b}_g^\top$ 는 각각 전역 좌표계 기준 IMU의 3차원 위치, 속도, 쿼터니언, 가속도 편향, 그리고 자이로 편향이다.

#### 3.3.1 초기화 과정

최적화 식이 복잡해질수록 국소 최소값(Local minimum)으로 수렴하여 잘못된 해를 구할 수 있다. 따라서 전역적 최소값(Global minimum)으로 수렴하기 위해서는 초기값 설정이 중요한 역할을 한다. 특히 포즈의 초기값에 따라서 백 엔드 모듈 성능이 크게 바뀔 수 있다. 포즈 초기값은 앞서 기술한 시각적 재방문 모듈에서 획득한 초기 포즈 위주로 설정한다. 이렇게 초기 포즈 값을 설정한다면 최적화 과정에서 잘못된 해가 도출되는 것을 방지할 수 있다. 만약 초기 포즈가 주어지지 않는다면 프론트 엔드에서 누적되어 계산된 포즈로 설정한다.

#### 3.3.2 최적화를 위한 팩터

인접한 키프레임간 상대 포즈를 정확히 나타내기 위해서 키프레임 간 IMU 적분 팩터(IMU pre-integration factor)<sup>20)</sup>를 이용한다. 키프레임 간 IMU적분 팩터는 이전 키프레임과 현재 키프레임 사이에 축적된 IMU 데이터를 적분하여 예상되는 위치, 속도와 편향을 잔차로 표시해 나타낸다. 이에 대한 수식은 다음과 같다.

$$\begin{aligned} \mathbf{r}_{\Delta\mathbf{R}_i} &= \text{Log} (f_1(\mathbf{x}_i, \mathbf{x}_{i-1})\mathbf{R}_{i-1}^\top\mathbf{R}_i) \\ \mathbf{r}_{\Delta\mathbf{v}_i} &= \mathbf{R}_{i-1}^\top (\mathbf{v}_i - \mathbf{v}_{i-1} - \mathbf{g}\Delta t_{i-1,i}) \\ &\quad - f_2(\mathbf{x}_i, \mathbf{x}_{i-1}) \\ \mathbf{r}_{\Delta\mathbf{p}_i} &= \mathbf{R}_{i-1}^\top (\mathbf{p}_i - \mathbf{p}_{i-1} - \mathbf{v}_{i-1}\Delta t_{i-1,i} \\ &\quad - \frac{1}{2}\mathbf{g}\Delta t_{i-1,i}^2) - f_3(\mathbf{x}_i, \mathbf{x}_{i-1}) \end{aligned} \quad (3)$$

$$\mathbf{r}_{imu}(\mathbf{z}_i, T, \mathcal{X}) = \mathbf{r}_{\Delta\mathbf{R}_i} + \mathbf{r}_{\Delta\mathbf{v}_i} + \mathbf{r}_{\Delta\mathbf{p}_i}$$

$$\mathbf{z}_{i,T} = ((\mathbf{z}_g, \mathbf{z}_a)_t, (\mathbf{z}_g, \mathbf{z}_a)_{t+1}, \dots, (\mathbf{z}_g, \mathbf{z}_a)_{t+T}) \quad (4)$$



여기서  $i$ 와  $i-1$ 은 키프레임 아이디에 해당하고,  $z_{i,T}$ 는 각각 자이로 데이터와 가속도 데이터가 인접한 키프레임이 차이나는 T step 만큼 누적된 데이터이다.  $i$ 에 해당하는 키프레임 상태는  $i-1$ 의 키프레임 상태의 기준으로 IMU 데이터들을 적분하여 나타내고, 이를 뺀 값을 각각 회전, 속도, 그리고 위치의 잔차로 표시한다. 또한  $f_1$ ,  $f_2$  그리고  $f_3$ 는 해당 잔차를 편향에 대하여 테일러 급수를 통하여 나타낸 값이다. 보통 이 함수들은 1차 선형 관계로 나타낸다.

### 3.3.3 강 결합 방식

강 결합 방식은 시각적 재방문에서 획득한 2D-3D 매칭 쌍들을 활용하여 이용한다. 잔차는 구형 모델 재사영 팩터(Spherical reprojection factor)를 통해 만들어진다. 이에 대한 수식은 다음과 같다.

$$\hat{p}_i = {}_c^T \mathbf{R} \left( {}_c^T \mathbf{R}_i ({}^c \mathbf{m}_{i,k} - {}_c^T \mathbf{p}_i) - {}_c^T \mathbf{t} \right)$$

$$P_i = \pi_1^{-1}({}^c \mathbf{f}_{i,k}) \tag{5}$$

$$\mathbf{r}_l(z_i, \mathcal{X}) = \frac{\hat{p}_i}{\|\hat{p}_i\|} - \frac{P_i}{\|P_i\|}$$

$$\mathbf{z}_l = (\mathbf{m}_{i,k}, \mathbf{f}_{i,k}) \tag{6}$$

여기서  $k$ 는  $i$ 번째 이미지에서 해당하는  $k$ 번째 대응점 매칭 인덱스이다. 강 결합 맵 매칭을 포함한 코스트를 표현하면 다음과 같다.

$$\mathcal{X}^* = \min_{\mathcal{X}} \{ \|\mathbf{r}_p - \mathbf{H}_p \mathcal{X}\|^2 + \sum_i \|\mathbf{r}_{imu}(z_{i,T}, \mathcal{X})\|^2 + \sum_{i,k} \|\mathbf{r}_l(z_i, \mathcal{X})\|^2 \}. \tag{7}$$

여기서  $\mathbf{r}_p$ 는 사전(Prior) 잔차이고,  $\mathbf{H}_p$ 는 사전 잔차의 자코비안(Jacobian) 행렬이다. 강 결합의 경우  $\mathbf{r}_l$ 을 포함하는 경우이고, 약 결합은 이를 제외한 나머지 잔차 값을 통해 최적화를 진행한다.

## 4. 데이터셋 결과

본 연구 내용을 검증하기 위하여 Complex Urban Dataset을 이용한다. Complex urban dataset은 이미지, 휠 오도메트리, IMU 데이터뿐만 아니라, 라이다 맵을 갖고 있어 본 연구에서 검증하기에 적합하다. 또한 실측 정보(Ground Truth) 포즈를 갖고 있어 알고리즘의 성능을 정량적으로 검증할 수 있다. 테스트 시나리오는 38번으로 설정하였고 라이다 맵은 28번 혹은 38번 시나리오 기반

맵으로 설정하였다. 본 알고리즘은 Intel i7-6,700 K 4.00 GHz CPU(8 cores)와 Nvidia Geforce RTX 4080 Super GPU를 사용하여 검증하였다.

### 4.1 맵과 테스트 시나리오가 동일했을 때의 결과

전 처리 라이다 지도 데이터를 만들 때의 시나리오와 테스트할 시나리오가 동일한 경우, 시각적 재방문이 잘 된다는 가정에서 백 엔드의 결과를 테스트할 수 있다. 강 결합과 약 결합의 결과는 Fig. 3과 같다. 약 결합 최적화와 강 결합 최적화 모두 성능이 우수하고 큰 차이가 없는 것을 확인할 수 있다. 특히 종방향 에러가 표시된 그래프에서 에러가 적은 것을 확인할 수 있다. 이는 정보가 많은 지도를 이용한다면, 카메라와 IMU, 그리고 휠 오도메트리를 이용해서 종 방향 오차를 줄일 수 있다는 것을 시사한다.

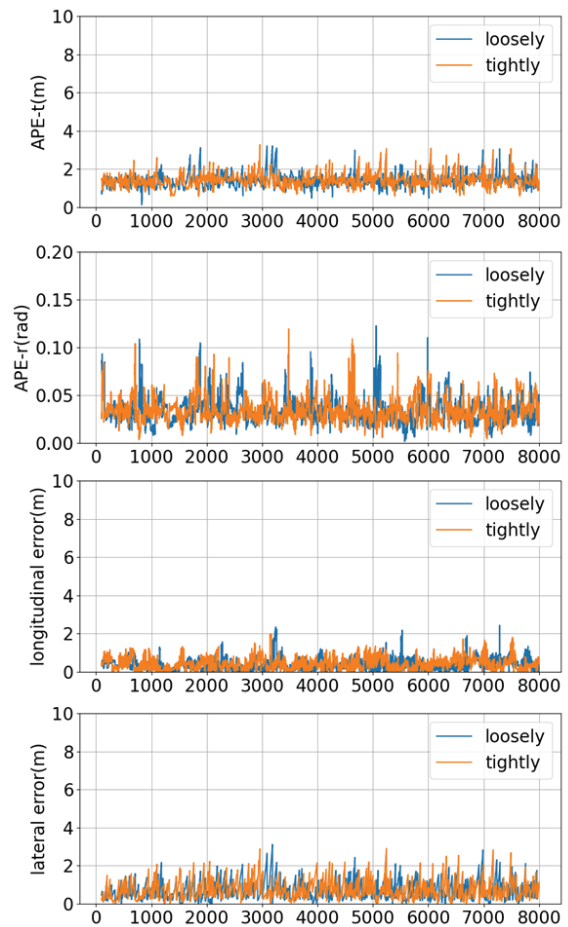


Fig. 3 Performance results on self-localization

## 4.2 맵과 시나리오가 다를 때의 결과

실제 상황에서는 전 처리된 라이다 지도 데이터가 만들어진 차선과, 차량이 실제 주행했을 때 관측된 차선이 다르거나 주변 환경이 변할 수 있다. 이를 고려하여 맵과 시나리오가 다를 때를 테스트하였다. 본 연구에서는 Complex urban dataset 중 38번 맵과 28번 테스트 시나리오를 통해 비교하였다.

이 테스트에서의 강 결합과 약 결합의 결과는 Fig. 4와 같다. 성능을 비교해본 결과, 종 방향 에러의 평균은 약 결합일 때 약 1.03 m, 강 결합일 때는 1.05 m이고, 횡방향 에러의 평균은 약 결합일 때는 1.61 m, 강 결합일 때는 1.92 m이다. 이 성능에서는 아직 약 결합이 강 결합보다 평균적으로 근소하게 성능이 나은 것을 확인할 수 있다. 이러한 성능 차이가 발생하는 이유는 시각적 재방문을 통해 추출된 매칭 포인트들이 정확하지 않았을 것이라고 예상된다. 이로 인해서 약 결합은 하나의 포즈로 추출되어 잔차를 줄여서 나타낼 여지가 있지만, 강 결합은 여

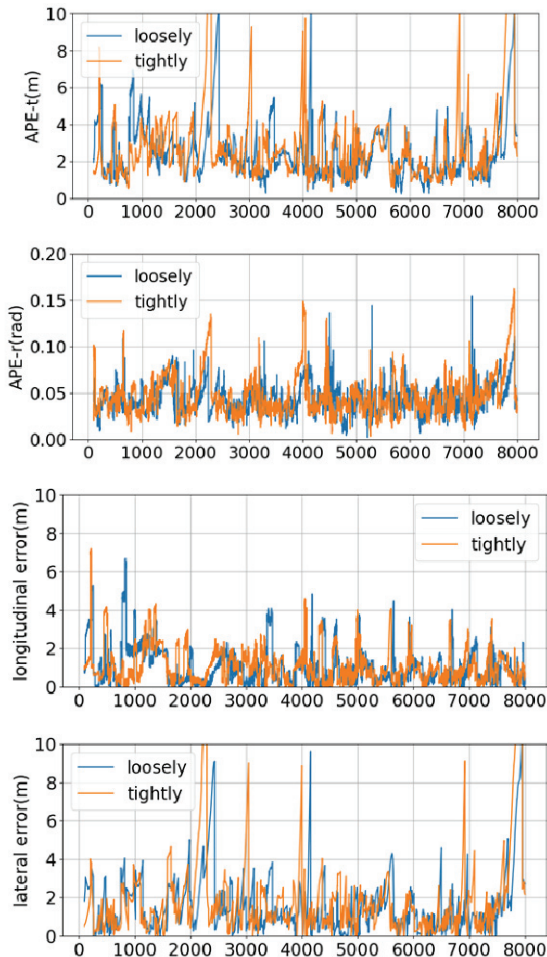


Fig. 4 Performance results on re-localization

러 개의 매칭 포인트들의 오차가 누적되어 성능 저하가 상대적으로 더 발생하였다.

또한 Reid 등<sup>21)</sup>에 따르면, 미국 지역 도로에서 종 방향 횡방향 오차가 모두 0.44 m 이상에서는 경고 신호를 줘야 한다고 기술하였다. 본 실험에서 확인한 결과에서는 이러한 성능 결과를 개선할 필요성이 있다. 첫번째 방법은 차선 별로 참조 이미지와 참조 전역 포즈를 할당하는 것이다. 이 방법은 차량의 위치에 따른 참조 이미지를 알맞게 찾을 수 있기 때문에 성능을 끌어올릴 수 있다. 그러나 이러한 방법은 많은 실험이 필요로 할 수 있다. 한편 두번째 방법은 본 알고리즘에서 사용하였던 Lightglue를 개선하는 것이다. Lightglue는 조도가 변한 이미지 환경에서는 성능이 뛰어나지만, 차선 차이에 따른 이미지 매칭은 성능 개선이 필요로 하다. 따라서 Lightglue 대신, 반복적인 패턴에 좀 더 강인한 매칭 알고리즘을 쓴다면 이를 개선할 수 있을 것이다.

## 5. 결론

본 논문에서는 라이다 지도를 이용한 최적화 기반의 측위 방식을 개발하였으며, 강 결합 방식과 약 결합 방식의 성능을 비교하고자 하였다. 기존 연구들에서는 주로 강 결합 방식을 통해 성능을 높였지만, 본 연구에서는 약 결합 방식으로도 충분히 성능을 개선할 수 있음을 확인하였다. 강 결합 방식에서 성능 향상을 위해서는 우선적으로 대응점 매칭이 대부분 성공적으로 이루어졌을 때 가능한데, 만약 대응점 매칭의 신뢰성이 크지 않은 경우, 약 결합 방식으로 최적화하는 것이 성능 면에서 더 효율적일 수 있음을 보여주었다.

## 후 기

본 논문은 산업통상자원부 및 산업기술평가관리원의 자율주행기술개발혁신사업(20018101, T-Car 기반 자율주행 인지예측/지능제어, 차량부품/시스템통합평가 기술개발)의 지원을 받아 수행되었습니다. 본 논문은 2024년도 정부(산업통상자원부)의 재원으로 한국산업기술진흥원의 지원(P0020536, 2024년 산업혁신인재성장지원사업)을 받아 수행되었습니다.

## References

- 1) C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. M. Montiel and J. D. Tardós, "ORB-SLAM3: An Accurate Open-Source Library for Visual, Visual-Inertial, and Multimap SLAM," IEEE Transactions on Robotics, Vol.37, No.6, pp.1874-1890, 2021.

- 2) T. Qin, P. Li and S. Shen, "VINS-Mono: A Robust and Versatile Monocular Visual-Inertial State Estimator," *IEEE Transactions on Robotics*, Vol.34, No.4, pp.1004-1020, 2018.
- 3) S. Hausler, S. Garg, M. Xu, M. Milford and T. Fischer, "Patch-NetVLAD: Multi-Scale Fusion of Locally-Global Descriptors for Place Recognition," *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp.14141-14152, 2021.
- 4) P.-E. Sarlin, A. Unagar, M. Larsson, H. Germain, C. Toft, V. Larsson, M. Pollefeys, V. Lepetit, L. Hammarstrand and F. Kahl, "Back to the Feature: Learning Robust Camera Localization from Pixels to Pose," *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp.3247-3257, 2021.
- 5) K. Koide, J. Miura, M. Yokozuka, S. Oishi and A. Banno, "Interactive 3D Graph SLAM for Map Correction," *IEEE Robotics and Automation Letters*, Vol.6, No.1, pp.40-47, 2020.
- 6) M. Sun, S. Yang and H. Liu, "Scale-Aware Camera Localization in 3D LiDAR Maps with a Monocular Visual Odometry," *Computer Animation and Virtual Worlds*, Vol.30, No.3-4, pp.e1879, 2019.
- 7) X. Zuo, P. Geneva, Y. Yang, W. Ye, Y. Liu and G. Huang, "Visual-Inertial Localization with Prior LiDAR Map Constraints," *IEEE Robotics and Automation Letters*, Vol.4, No.4, pp.3394-3401, 2019.
- 8) K. Yabuuchi, D. R. Wong, T. Ishita, Y. Kitsukawa and S. Kato, "Visual Localization for Autonomous Driving Using Pre-Built Point Cloud Maps," *2021 IEEE Intelligent Vehicles Symposium (IV)*, pp.913-919, 2021.
- 9) K. Yabuuchi and S. Kato, "VMVG-Loc: Visual Localization for Autonomous Driving Using Vector Map and Voxel Grid Map," *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp.6976-6983, 2022.
- 10) C. Zhang, H. Zhao, C. Wang, X. Tang and M. Yang, "Cross-Modal Monocular Localization in Prior LiDAR Maps Utilizing Semantic Consistency," *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp.4004-4010, 2023.
- 11) R. W. Wolcott and R. M. Eustice, "Visual Localization within LiDAR Maps for Automated Urban Driving," *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp.176-183, 2014.
- 12) Y. Kim, J. Jeong and A. Kim, "Stereo Camera Localization in 3D LiDAR Maps," *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp.1-9, 2018.
- 13) T. Caselitz, B. Steder, M. Ruhnke and W. Burgard, "Monocular Camera Localization in 3D LiDAR Maps," *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp.1926-1931, 2016.
- 14) H. Yu, W. Zhen, W. Yang, J. Zhang and S. Scherer, "Monocular Camera Localization in Prior LiDAR Maps with 2D-3D Line Correspondences," *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp.4588-4594, 2020.
- 15) X. Zheng, W. Wen and L.-T. Hsu, "Tightly-Coupled Line Feature-Aided Visual Inertial Localization within Lightweight 3D Prior Map for Intelligent Vehicles," *2023 IEEE 26th International Conference on Intelligent Transportation Systems (ITSC)*, pp.6019-6026, 2023.
- 16) D. DeTone, T. Malisiewicz and A. Rabinovich, "SuperPoint: Self-Supervised Interest Point Detection and Description," *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp.224-236, 2018.
- 17) S. Zhou, L. Luo, X. Bai and W. Zhou, "Patch2Pix: Epipolar-Guided Pixel-Level Correspondences," *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.466-475, 2021.
- 18) P.-E. Sarlin, D. DeTone, T. Malisiewicz and A. Rabinovich, "LightGlue: Local Feature Matching at the Speed of Light," *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.12345-12356, 2023.
- 19) G. Bradski, "The OpenCV Library," *Dr. Dobb's Journal of Software Tools*, 2000.
- 20) C. Forster, L. Carlone, F. Dellaert and D. Scaramuzza, "IMU Preintegration on Manifold for Efficient Visual-Inertial Maximum-a-Posteriori Estimation," *Robotics: Science and Systems XI*, 2015.
- 21) T. G. R. Reid, S. E. Houts, R. Cammarata, G. Mills, S. Agarwal, A. Vora and G. Pandey, "Localization Requirements for Autonomous Vehicles," *arXiv preprint arXiv:1906.01061*, 2019.