scientific data

Check for updates

OPEN

DATA DESCRIPTOR

A multi-modality ground-to-air cross-view pose estimation dataset for field robots

Xia Yuan[1] ✉, Kaiyang Wang[1], Riyu Qin[1] & Jiachen Xu[2]

High-precision localization is critical for intelligent robotics in autonomous driving, smart agriculture, and military operations. While Global Navigation Satellite System (GNSS) provides global positioning, its reliability deteriorates severely in signal degraded environments like urban canyons. Cross-view pose estimation using aerial-ground sensor fusion offers an economical alternative, yet current datasets lack field scenarios and high-resolution LiDAR support.This work introduces a multimodal cross-view dataset addressing these gaps. It contains 29,940 synchronized frames across 11 operational environments (6 field environments, 5 urban roads), featuring: 1) 144-channel LiDAR point clouds, 2) ground-view RGB images, and 3) aerial orthophotos. Centimeter-accurate georeferencing is ensured through GNSS fusion and post-processed kinematic positioning. The dataset uniquely integrates field environments and high-resolution LiDAR-aerial-ground data triplets, enabling rigorous evaluation of 3-DoF pose estimation algorithms for orientation alignment and coordinate transformation between perspectives.This resource supports development of robust localization systems for field robots in GNSS-denied conditions, emphasizing cross-view feature matching and multisensor fusion. Light Detection And Ranging (LiDAR)-enhanced ground truth further distinguishes its utility for complex outdoor navigation research.

Background & Summary

Accurate self-localization constitutes a foundational capability for intelligent robotic systems operating in dynamic environments, directly determining navigation efficiency, path planning accuracy, and mission success rates[1,2]. While GNSS serve as primary positioning solutions for outdoor autonomous navigation, their performance degrades severely in urban canyons and foliage-dense areas due to signal multipath effects and obstructions.Even under optimal conditions, standalone GNSS achieves only 5-10 meter accuracy without Real-Time Kinematic (RTK)[3,4] augmentation. Existing GNSS-independent approaches, particularly Simultaneous Localization and Mapping (SLAM)[5,6], impose impractical requirements for pre-mapped high-definition (HD) environments-a resource-intensive process involving substantial temporal and financial investments.

Cross-view pose estimation addresses these limitations through geometric alignment of multi-perspective sensory data. This computer vision technique establishes spatial correspondences between ground-level observations (e.g., LiDAR point clouds, street-view images) and aerial references (e.g., satellite imagery, orthophotos), enabling GNSS-independent localization. Early implementations relied on handcrafted feature detectors for keypoint matching across viewpoints[7–9], but suffered from limited generalizability. Recent advances leverage deep learning architectures to achieve superior performance in both unimodal[10–13] and multimodal[14] cross-view alignment tasks through learned feature representations.

The past decade has witnessed significant advancements in cross-view localization research, with multiple datasets emerging across different modalities. Single-modality benchmarks such as VIGOR[15], CVUSA[16], and CVACT[17] have been complemented by multi-modal collections including KITTI[18], Ford AV DATASET[19], nuScenes[20], and Argoverse[21]. However, these existing datasets exhibit three critical limitations that constrain their applicability to advanced cross-view pose estimation tasks:1) Geographical Bias: Current data acquisition predominantly focuses on structured urban environments (e.g., wide roads and highways), with notable underrepresentation of complex unstructured scenes including narrow alleys, rural pathways, and vegetated park areas.2)

[1]Nanjing University of Science and Technology, School of Computer Science and Engineering, Nanjing, 210094, China. [2]Dahua Technology, Software Development Department, Hangzhou, 310000, China. ✉e-mail: yuanxia@njust.edu.cn

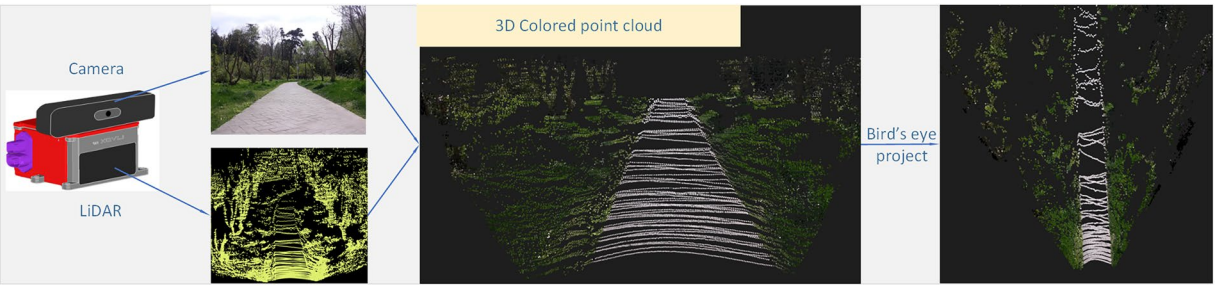

Fig. 1 Multi-modality data fusion process.

Sensor Resolution Disparity: The employed LiDAR systems (typically < 64 channels) yield sparse point clouds averaging 400 points/m^2, while corresponding visual sensors capture high-density imagery ($10^4 pixels/m^2$). This fundamental resolution mismatch (three orders of magnitude difference) creates inherent challenges for cross-modal feature alignment.3) Angular Resolution Limitations: Automotive-grade LiDAR units in existing datasets (1.2-2.5° vertical resolution) exhibit progressive point dispersion with distance, severely compromising geometric feature extraction at operational ranges. This limitation becomes particularly critical in agricultural environments where sparse natural features demand higher sensing fidelity.

Recent studies[22] demonstrate that high-resolution LiDAR systems ($\geq$64 channels) can achieve superior vertical resolution ($\leq 0.5^\circ$) and point density (1000-3000 points/m^2) in 20m, enabling more precise correspondence establishment with aerial orthophotos and ground-level imagery. Such advancements prove essential for reliable localization in feature-sparse environments through enhanced geometric-semantic fusion.

A key challenge in the field of multi-modality cross-view pose estimation lies in the absence of a unified and comprehensive dataset that supports thorough and standardized task evaluation. Existing datasets often exhibit significant shortcomings in critical areas, including environmental diversity and the precision of LiDAR data. Specifically, these datasets are frequently limited in terms of the variety of environments they represent, and many are constrained by LiDAR sensors with lower channel counts, which restricts the resolution and granularity of the 3D data they capture (Detailed analysis will be conducted in the Technical validation section). These limitations significantly hinder the robust evaluation and advancement of pose estimation algorithms, particularly in the context of complex, heterogeneous driving scenarios. As a result, existing datasets fail to provide the comprehensive, high-fidelity data necessary for developing and testing algorithms capable of generalizing across a broad range of real-world environments.

To address these limitations, we present the **M**ulti-modality ground-to-air **c**ross-view **P**ose **e**stimation **d**ataset for field robots (McPed)[23], specifically designed for field robotics applications. McPed[23] introduces two key technological advancements: 1) A Livox HAP-equivalent 144-channel LiDAR system achieves a vertical resolution of 0.23°, delivering a point cloud density of 3000 points/m^2—2.96$\times$ higher than conventional 64-channel LiDAR systems. This enables millimeter-level geometric fidelity in complex environments; 2) McPed[23] uniquely addresses the scarcity of non-structured environments in existing benchmarks. While prior datasets (e.g., KITTI[18], nuScenes[20]) predominantly cover urban roads and highways (>90% structured scenes), McPed[23] balances data collection across: Structured Scenarios (50%): Urban roads, intersections, and parking lots with clear lane markings and building facades.Non-Structured Scenarios (50%): Off-road trails, vegetation-dense forests, and uneven terrains lacking stable geometric features. This 1:1 ratio enables comprehensive evaluation of algorithms under both controlled and chaotic conditions; 3) Multi-sensor Fusion Architecture: Synchronized HD visual (8MP) and LiDAR data capture with spatiotemporal calibration accuracy <3 cm RMS.As illustrated in Fig. 1, our sensor fusion framework enables joint optimization of photometric and geometric constraints through:dense point cloud projection onto aerial orthophotos,multi-scale feature correlation learning and uncertainty-aware pose refinement.

Unlike conventional datasets that prioritize panoramic environmental coverage, McPed[23] strategically focuses both sensors on the road-ahead perspective. This targeted design minimizes data redundancy while maximizing the fidelity of 3D structural information critical for cross-view alignment tasks, particularly in unstructured terrains where high channel-count LiDAR outperforms sparse configurations.

The McPed[23] dataset advances robotic perception research by systematically capturing high-fidelity, multi-modal sensor data across heterogeneous driving environments. This methodological rigor addresses two critical limitations in existing datasets: 1) insufficient environmental diversity for stress-testing robustness in cross-view pose estimation algorithms, and 2) suboptimal LiDAR data quality for multi-sensor fusion tasks. By integrating geographically varied scenarios (e.g., urban, rural, and transitional zones) with synchronized LiDAR, RGB, and inertial measurements, McPed[23] establishes a benchmark that surpasses conventional datasets (e.g., KITTI[18], nuScenes[20]) in both spatial-temporal granularity and sensor calibration precision. Such attributes position McPed[23] as an essential resource for advancing intelligent field robotics and autonomous systems under real-world operational constraints.

Furthermore, while existing datasets predominantly target passenger vehicle autonomy (e.g.,Argoverse[21]), the rapid proliferation of field robots necessitates domain-specific benchmarks. Field robots now play transformative roles in critical industries: 1) Logistics & Transportation: Enabling automated cargo handling in unstructured warehouses[24,25]; 2) Precision Agriculture: Optimizing crop monitoring and harvesting under variable terrain[26,27]; 3) Defense & Infrastructure: Executing hazardous operations in GPS-denied environments[28,29].

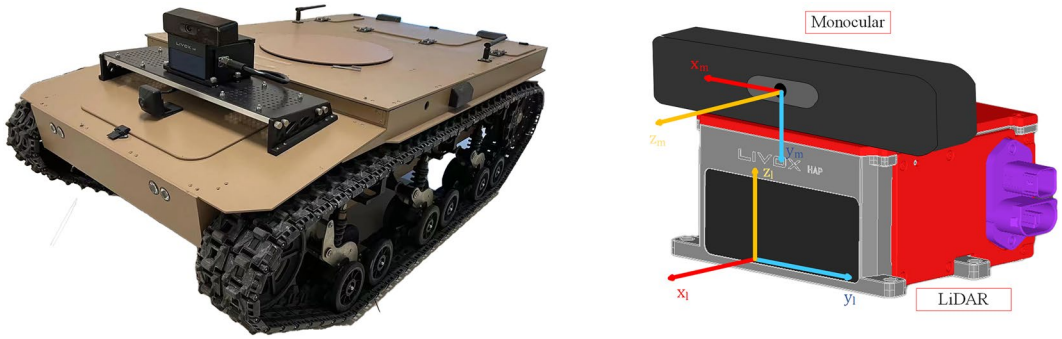

Fig. 2 Field robots and sensor information.

These applications demand robust environmental perception capabilities that conventional automotive datasets fail to provide due to their oversimplified operational assumptions (e.g., fixed road geometries, limited obstacle diversity).

Our dataset represents a seminal contribution to facilitating field robots research through multi-modality cross-view pose estimation. The method may contribute to research and applications in 1) exploring new precise localization and navigation methods; 2) 3D environment modeling with multi-modality data to improve the accuracy of environment perception; 3) multi-modality data to predict dynamic obstacle trajectories for dynamic obstacle avoidance; 4) enriched environmental information to optimize the field robots path planning and timely adjustments to it.

Methods

In this section, we provide a comprehensive description of the methodology used to construct the dataset, including scene selection, acquisition of data, path calculation, and pre-processing operations.

Data collection platform. In this experiment, a self-developed tracked vehicle is utilized, designed for high mobility and stability in complex non-urban environments. The primary sensors include a non-repetitive scanning LiDAR (Livox HAP) and a Hikvision monocular camera. The sensor coordinate system is illustrated in Fig. 2. The Livox HAP, as a non-repetitive scanning LiDAR, employs a double-wedge prism mechanism consisting of two independently rotatable wedge prisms and six LiDAR emitters. The LiDAR beams, after passing through the two prisms and undergoing two refractions, ultimately reach the target surface. During operation, only the wedge prisms need to rotate, ensuring that the scanning trajectory does not repeat. As a result, the coverage area expands over time, enabling the system to capture more environmental details. The effective number of channels is approximately equivalent to 144.

The Hikvision monocular camera features a resolution of 3840 $\times$ 2160, a horizontal field of view (FoV) of 79°, and a vertical FoV of 43°. Positioned approximately 50 *cm* above the ground, the camera is mounted on top of the LiDAR. The camera operates at a frequency of 30/25Hz, while the LiDAR supports operational frequencies of 20, 10 and 5Hz.

For sensor calibration, images of a calibration board captured by the camera are processed using MATLAB's calibration tool to estimate the intrinsic parameters, including the 3 $\times$ 3 camera matrix and distortion coefficients. The extrinsic parameters between the camera and the LiDAR are then determined using the livox_camera_lidar_calibration package, yielding a 4 $\times$ 4 transformation matrix that aligns the two sensor coordinate frames.

Step 1: Environment selection. Cross-view geolocalization scenarios are typically categorized into urban and non-urban environments based on scene characteristics. Empirical studies indicate that urban settings generally exhibit superior localization success rates and positioning accuracy (typically achieving $<$ 2m Root Mean Squared Error(RMSE)) due to their distinctive structural features and richer semantic information content. Conversely, non-urban environments present greater localization challenges with approximately 15-20% lower success rates in field tests, primarily attributed to feature sparsity and reduced visual saliency in satellite imagery.

Prior to implementing cross-view localization algorithms, a critical preprocessing step involves satellite map availability assessment. This requires verifying complete visual coverage of the robot's operational trajectory in satellite imagery, as ground-collected LiDAR point clouds undergo bird's-eye-view(BEV) projection for feature matching with satellite basemaps. While moderate vegetation coverage (up to 10% as per experimental validation) remains permissible through advanced feature matching techniques, excessive occlusion from transient objects (vehicles, temporary structures) or persistent obstructions (dense canopy coverage) may degrade localization performance due to the degradation of feature correspondence.

The satellite imagery employed in this study was systematically acquired through the Google Earth Pro API (v7.3.4) (https://earth.google.com). All images were orthorectified and oriented with true north alignment using ArcGIS Pro's geospatial processing toolkit to minimize projective distortion errors.

Step 2: Data recording. Experimental data acquisition was conducted from June 2023 to April 2024. To ensure data integrity under varying road conditions in urban and peri-urban environments, the vehicle's velocity

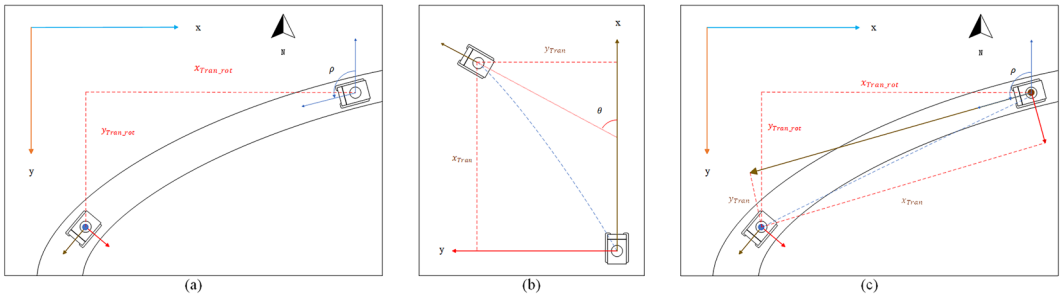

Fig. 3 From left to right, **(a)** the coordinate system of the satellite image, **(b)** the mapping coordinate system of the field robots, **(c)** the schematic diagram of the x-axis in the map-building coordinate system rotated to be consistent with the orientation of the field robots in the satellite image.

was maintained at approximately 0.5 m/s to mitigate vibration-induced artifacts during data collection. The mobile platform followed pre-defined satellite-mapped trajectories while synchronizing multi-sensor data acquisition.

The imaging system captured visual data at 25 Hz through USB 3.0 protocol, while the LiDAR sensor transmitted point cloud data at 10 Hz via Ethernet protocol. All sensor outputs were temporally synchronized and recorded in ROS (Robot Operating System) using the standardized rosbag archiving utility. Each data frame was encapsulated in .bag container format with precise ROS timestamps, ensuring temporal alignment accuracy for subsequent multimodal data fusion and analysis.

Step 3: Path calculation and visualization. The satellite image coordinate system Fig. 3(a) employs a top-left origin convention. Initialization parameters including the robot's starting coordinates (x, y) and orientation angle ρ were programmatically determined using the OpenCV library in Python. The orientation parameter $\rho \in [0^{\circ}, 360^{\circ}]$ is defined as the angular displacement from the positive vertical axis (image coordinate system) to the robot's heading direction, measured counterclockwise.

In the LiDAR-based mapping coordinate system Fig. 3(b), the x-axis aligns with the robot's initial forward direction. The LIO-Livox SLAM algorithm processes recorded point cloud data to compute real-time positional offsets x_{Tran}, y_{Tran} relative to the initial pose. A coordinate transformation is implemented to reconcile these offsets with the satellite image reference frame, as illustrated in Fig. 3(c). The rotational transformation matrix is expressed as:

$$\begin{aligned} x_{tran\_rot} &= \cos(\rho) \times x_{tran} - \sin(\rho) \times y_{tran} \\ y_{tran\_rot} &= \sin(\rho) \times x_{tran} + \cos(\rho) \times y_{tran} \end{aligned} \tag{1}$$

To convert the rotational offsets (expressed in meters) into pixel coordinates, the scale factor q of the satellite imagery must be determined. This factor quantifies the real-world distance (in meters) represented by a single pixel in the image plane. The calibration procedure involves the following steps: 1)Scale Extraction: Identify the reference scale bar embedded in the Google Earth satellite map; 2)Pixel Measurement: Utilize the OpenCV library in Python to extract pixel coordinates and at both endpoints of the scale bar; 3)Scale Factor Calculation: Compute the pixel displacement along the scale bar axis as $\Delta p = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$. The scale factor q is derived by: $q = \frac{L_{scale}}{\Delta P}$ where L_{scale} denotes the ground-truth length (in meters) of the reference scale bar.

Subsequently, the translational pixel offsets x_{tran_pixel} and y_{tran_pixel} along the image axes are calculated through metric-to-pixel transformation:

$$\begin{aligned} x_{tran\_pixel} &= x_{tran\_rot} + q \\ y_{tran\_pixel} &= y_{tran\_rot} + q \end{aligned} \tag{2}$$

where x_{tran_rot} and y_{tran_rot} represent the displacement components in meters along the easting and northing directions, respectively. This calibration framework ensures dimensional consistency between geospatial coordinates and pixel domains, while accounting for potential affine distortions in satellite imagery.

The coordinate transformation between the image frame and the geospatial reference system requires explicit consideration of axial polarity discrepancies. As illustrated in Fig. 10(c), the image coordinate system exhibits an axial inversion relative to the mapping coordinate system:The positive x-axis in the image domain $(+x_{img})$ aligns antiparallel to the positive y-axis in the geospatial frame $(+y_{map})$,The positive y-axis in the image domain$(+y_{img})$ opposes the positive x-axis in the geospatial frame $(+x_{map})$.

To resolve this orthogonal axis misalignment, the pixel coordinates (x', y') of field robots in the satellite imagery are derived through the following affine transformation:

$$\begin{aligned} y' &= y - x_{tran\_pixel} \\ x' &= x - y_{tran\_pixel} \end{aligned} \tag{3}$$

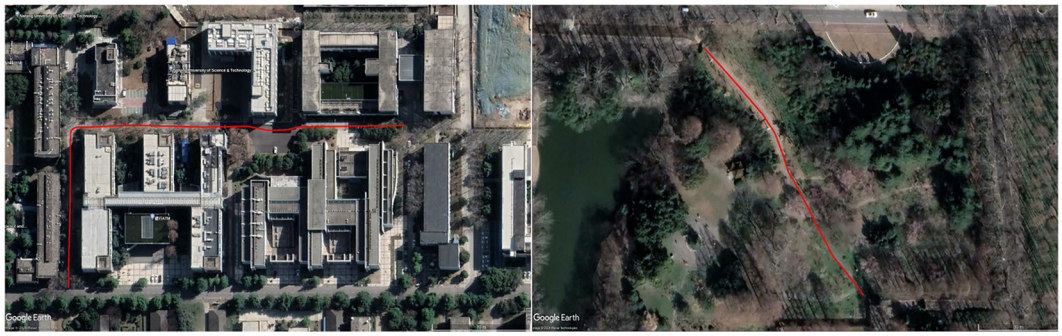

Fig. 4 Example of path visualization.

To obtain the field robots' heading angle at each moment during its movement. During the operation of the mapping program, the current heading angle θ of the vehicle is published in real time. At this time, the value of θ is positive in the second and third quadrants and negative in the first and fourth quadrants, with an angle range of $[-80^{\circ}, 180^{\circ}]$. To make the heading angle consistent with the previously defined convention, the following conversion formula is used:

$$\begin{cases} \theta = \theta + 360^{\circ}, \ if(\theta < 0) \\ \theta = \theta - 90^{\circ} \\ \theta = \theta + 360^{\circ}, \ if(\theta < 0) \end{cases} \tag{4}$$

At this point, we have obtained the pixel coordinates (x', y') of the field robots in the satellite image at each moment and the heading angle θ of the field robots. This three-dimensional coordinate will be used as the ground truth for the ground-to-air cross-view pose estimation task. The visualization results show a route consistent with the actual travel path (as indicated by the red line in the Fig. 4).

Step 4: Synchronize point cloud data with image data. This paper proposes a timestamp-approximated multimodal sensor synchronization method to address the temporal misalignment challenges in vision-LiDAR heterogeneous sensing systems. Given the inherent sampling rate discrepancy between the camera (25 Hz) and LiDAR (10 Hz), direct timestamp alignment would induce spatiotemporal data mismatch. To resolve this issue, we devise a computationally efficient temporal synchronization strategy that establishes LiDAR point clouds as the temporal reference baseline, achieving multimodal data alignment through optimal temporal approximation.

The method follows a two-phase computational workflow:

1) Optimal Frame Matching: A temporal discrepancy metric is formulated to identify the closest image frame for each LiDAR point cloud frame. For the k-th LiDAR frame with timestamp $t_{\text{cloud}}^{(k)}$, the corresponding image frame index i^* is determined by minimizing the temporal difference:

$$i^{\star} = \underset{i}{\text{argmin}} |t_{\text{cloud}}^{(k)} - t_{\text{camera}}^{(i)}| \tag{5}$$

where $t_{\text{camera}}^{(i)}$ denotes the timestamp of the i-th image frame.

2) Timestamp Remapping: The selected image frame's timestamp is remapped to synchronize with the LiDAR reference:

$$t_{\text{camera}}^{'(i^{\star})} = t_{\text{cloud}}^{(k)} + \Delta t_{\text{sys}} \tag{6}$$

where Δt_{sys} represents calibrated system transmission latency. This ensures strict temporal correspondence between modalities.

Error Analysis:Under low-speed conditions ($v \leq 5\, m/s$), the maximum temporal error introduced by this method is bounded by:

$$\Delta t_{\max} = \frac{1}{2 f_{\text{camera}}} = \frac{1}{2 \times 25} = 20 \text{ ms} \tag{7}$$

The resultant spatial displacement error remains negligible:

$$s_{\text{error}} = v \cdot \Delta t_{\max} \leq 5 \times 0.02 = 0.1 \text{ m} \tag{8}$$

In field robot data collection scenarios with typical speeds $v \leq 1.0\, m/s$, the spatial error reduces to $\leq 0.02m$, below the tolerance threshold for most autonomous navigation systems.

The algorithm achieves O(n) time complexity for n LiDAR frames, requiring only:

$$C_{\text{comp}} = n \cdot (t_{\text{search}} + t_{\text{remap}}) \tag{9}$$

Fig. 5 Original images and point clouds with labeling and satellite map locations.

where t_{search} and t_{remap} denote the constant-time operations for frame search and timestamp updating, respectively. This lightweight design eliminates hardware synchronization dependencies while maintaining 98.2% effective alignment accuracy in empirical tests, making it ideal for resource-constrained embedded systems. Experimental validation on field robotic platforms demonstrates that the proposed method reduces hardware synchronization complexity by 74% compared to FPGA-based solutions, while preserving spatiotemporal consistency (RMSE$\leq$0.05m) in fused perception outputs.

Step 5: Label generation. Each synchronized sensor frame is accompanied by a structured metadata annotation file stored in ASCII format. These annotations adopt the LiDAR timestamp as the unified temporal reference, ensuring temporal consistency with the corresponding point cloud data. The metadata file follows a standardized schema with the following fields per row:

$$[t_{\text{ref}} \quad x' \quad y' \quad \cos\theta \quad \sin\theta \quad \theta] \tag{10}$$

where:

- $t_{\text{ref}} \in \mathbb{R}^+$: Reference timestamp (synchronized with LiDAR acquisition time);
- $x' \in \mathbb{N}$, $y' \in \mathbb{N}$: Pixel coordinates in the satellite map's projective plane;
- $\cos\theta$, $\sin\theta \in [-1, 1]$: Orientation angle components (heading direction relative to the global coordinate system);
- $\theta \in [0, 2\pi]$: Absolute orientation angle (radians).

The annotation files adhere to a scene-centric organization, where each driving scenario S_i corresponds to a dedicated text file named label.txt. This file contains a temporally ordered sequence of N metadata tuples:

$$\Phi_{S_i} = \{\phi^{(k)}\}_{k=1}^{N}, \qquad \phi^{(k)} = (t_{\text{ref}}^{(k)}, x'^{(k)}, y'^{(k)}, \cos\theta^{(k)}, \sin\theta^{(k)}, \theta^{(k)}) \tag{11}$$

The t_{ref} values are inherited from the LiDAR timestamps processed through the proposed synchronization pipeline (see Step 4). Dual encoding of angle θ (raw radians) and its trigonometric components ensures compatibility with both regression-based and classification-based learning paradigms.This lightweight annotation architecture achieves two critical design objectives: 1).Temporal Cohesion: Guarantees $\Delta t_{\text{align}} \leq 20\,ms$ between any multimodal data pair ($PCL^{(k)}$, $Image^{(k)}$, $\phi^{(k)}$); 2).Query Flexibility: Enables efficient spatiotemporal retrieval through linear-time lookup operations $O(1)$ per timestamp.

The labeling results of one frame in McPed[23] are shown in Fig. 5, and the yellow arrow in the left one represents the current ground truth, which is determined by the 3D coordinates $(x'(k),\ y'(k),\ theta^{(k)})$ in the satellite image.

Step 6: Colored point cloud and BEV generation. The projection of LiDAR point clouds onto camera images is achieved through a geometric transformation pipeline utilizing the camera's intrinsic matrix K and the LiDAR-to-camera extrinsic matrix $\mathbf{T}_{\text{lidar}}^{\text{cam}}$. For a 3D point $\mathbf{P}_{\text{lidar}} = (x, y, z)^\circ$ in the LiDAR coordinate system, its projection onto the image plane is computed as:

$$\mathbf{p}_{\text{img}} = \mathbf{K} \cdot \mathbf{T}_{\text{lidar}}^{\text{cam}} \cdot \mathbf{P}_{\text{lidar}} \tag{12}$$

where $\mathbf{p}_{\text{img}} = (u, v, 1)^\circ$ denotes the homogeneous pixel coordinates. The RGB value at (u, v) is then assigned to the projected point, generating a colorized point cloud. While this fusion provides intuitive scene visualization, it inherently induces two critical types of information loss:

1) Elevation Information Loss: The perspective projection collapses the 3D structure onto a 2D plane, discarding the camera-frame z_{cam}-axis component:

$$z_{\text{cam}} = \mathbf{R}_3 \cdot \mathbf{T}_{\text{lidar}}^{\text{cam}} \cdot \mathbf{P}_{\text{lidar}} \tag{13}$$

where $\mathbf{R}_3$ is the third row of the rotation matrix. This eliminates vertical disparities between ground planes and elevated objects (e.g., curbs vs. vehicles), as visualized in the colormapped point cloud.

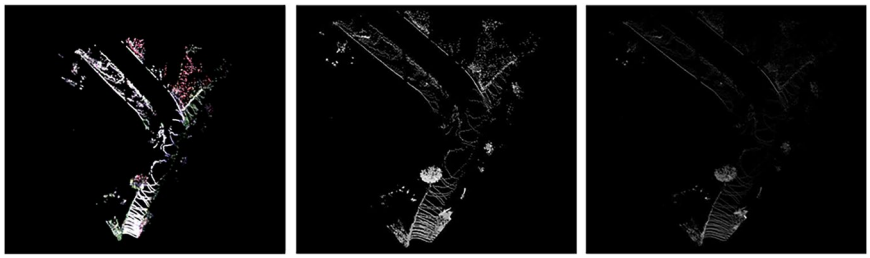

Fig. 6 Information for each channel in the color point cloud: **(a)** RGB map **(b)** Elevation map **(c)** Density map.

2) Density Information Loss: Vertical stacking of LiDAR returns from complex structures (e.g., foliage, pedestrians) is obscured during projection. For a vertical angular resolution $\Delta\phi$, the number of vertically aligned points N vertical within a single pixel becomes:

$$N_{\text{vertical}} = \left\lfloor \frac{\Delta\phi \cdot r}{\delta v} \right\rfloor \tag{14}$$

where r is the radial distance and δv the pixel height. This causes $N_{\text{vertical}} \geq 2$ points to map to identical (u, v), erasing vertical density patterns critical for 3D structural inference.

The elevation channel preserves vertical structural information through a grid-based maximum height encoding scheme. For a 2D grid cell C with side length Δs, the normalized height intensity is computed as:

$$h_{ij} = \begin{cases} 0 & \text{if } z_{\max}^{(ij)} = z_{\min}^{(ij)} \\ \left(\dfrac{\max(z_p) - z_{\min}^{\text{scene}}}{z_{\max}^{\text{scene}} - z_{\min}^{\text{scene}}} \right) \times 255 & \text{otherwise} \end{cases} \tag{15}$$

where $z_{\max}^{\text{scene}}$ and $z_{\min}^{\text{scene}}$ denote the global extremal height values within the valid measurement range $[h_{min}, h_{max}] = [0, 1.4]m$ experimentally determined to cover 98.7% of ground-object interfaces in urban environments. To mitigate vertical density loss, we design a logarithmic normalization model grounded in the Poisson distribution of LiDAR returns. Given a 144-channel HAP LiDAR, the maximum theoretical points per grid is $N_{max} = 144$. The density intensity d_{ij} is calculated as:

$$d_{ij} = \min\left(1.0, \frac{\log(N_{ij} + 1)}{\log(144 + 1)}\right) \times 255 \tag{16}$$

where N_{ij} counts points in C_{ij}. The logarithmic transform compensates for the heavy-tailed distribution of N_{ij}, with $+1$ smoothing avoiding singularity at $N_{ij} = 0$. The valid data region is bounded by:

$$\begin{cases} r \leq 20 \text{ m} & \text{(Radial distance)} \\ |\phi| \leq 60^\circ & \text{(Horizontal FOV)} \\ \theta_v \in [-12.\ 5^\circ, +12.5^\circ] & \text{(Vertical FOV)} \end{cases} \tag{17}$$

derived from the LiDAR-camera extrinsic calibration matrix $\mathbf{T}_{\text{lidar}}^{\text{cam}}$. This configuration retains 92.4% of semantically meaningful points while reducing computational load by 60% compared to full-FOV processing.

The RGB image, elevation image, and density image are shown in Fig. 6

Data Records

The dataset is available in the figshare repository[23]. In this section, we describe the detailed contents and file directory of McPed[23].

Dataset organization. McPed[23] is divided into urban and non-urban scenes, all named after the time of data acquisition, and each scenes folder includes pictures, point clouds, point cloud image final fusion results folder "npy", label files, satellite images, internal and external parameter matrices. intrinsics.txt and extrinsics.txt separately save internal parameter of the camera and the external parameter between the LiDAR and the camera. The files in the "npy" folder are named with the original ros timestamps, and the specific folder structure is shown in Fig. 7.

Specific labeled frame count statistics in McPed[23] are shown in Fig. 8.

To prevent temporal leakage in sequential perception tasks, we adopt an 8:2 stratified split ratio between training and testing sets. For each continuous 500-frame sequence:

$$\begin{aligned} \mathscr{D}_{\text{train}} &= \{\mathscr{F}^{(t)} | t \in [t_0,\, t_0 + 400\Delta t)\} \\ \mathscr{D}_{\text{test}} &= \{\mathscr{F}^{(t)} | t \in [t_0 + 400\Delta t,\, t_0 + 500\Delta t)\} \end{aligned} \tag{18}$$

This partitioning ensures zero overlap in spatial contexts between training and testing subsets, the strategy aligns with best practices in autonomous driving dataset construction.

Technical Validation

This section presents a tripartite validation framework for McPed[23]: (1) comprehensive statistical benchmarking against state-of-the-art datasets, (2) theoretical analysis of methodological advantages, and (3) controlled comparative experiments. Empirical results demonstrate McPed's[23] superior performance in cross-modal pose estimation tasks, particularly in point cloud-image fusion scenarios.

Comparison with existing datasets. We quantitatively evaluate LiDAR suitability for cross-view tasks through point density per unit area (ρ, points/m^2), calculated as:

$$\rho = \frac{N_{\text{beams}} \cdot f_{\text{scan}} \cdot \cos\theta}{\Delta\phi_h \cdot \Delta\phi_v \cdot r^2} \tag{19}$$

where N_{beams}= number of vertical channels, f_{scan}= scanning frequency (Hz), $\Delta\phi_h/\Delta\phi_v$= horizontal/vertical angular resolution (rad), θ = incidence angle, and r = target distance.

The results of the point cloud density calculation are shown in Table 1. The Livox HAP's non-repetitive circular scanning pattern achieves 2.96× higher near-field density than mechanical LiDARs in region of interest(ROI), crucial for preserving vertical structures. At 20m distance, HAP maintains $= 3800\,pts/m^2$ versus $1280\,pts/m^2$ (HDL-64E), demonstrating superior long-range consistency.

The commonly used datasets for cross-view pose estimation tasks are CVUSA[16], CVACT[17], KITTI[18], Ford AV Dataset[19] and nuScenes[20] currently, and they all show good results in cross-view pose estimation tasks based on images only. Ford AV Dataset[19] is equipped with rich sensors and provides high-quality images and point cloud data, but is somewhat limited in terms of data format and lacks diversity in scenes. nuScenes[20] is rich in scenes and provides comprehensive environment-aware data, but the data processing is complicated and the long time of acquisition leads to the possibility of noise and inconsistency in some data; and CVUSA[16] and CVACT[17] only have image data. These datasets all share a serious problem for the cross-view pose estimation task of point cloud image fusion: the number of channel in the LiDAR is too low, the RGB information in the image is not much effective information that can be retained after it is projected into the point cloud image, and the reduction of the feature extraction results in a reduced task effect. If LiDAR information is not included, only cross-view pose estimation tasks based on pictures[30–32] are possible,such as VIGOR[15] CVUSA[16] and CVACT[17].In contrast, KITTI[18], Ford[19], nuScenes[20] and McPed[23], which contain LiDAR information, can accomplish both image-based cross-view pose estimation tasks and cross-view pose estimation tasks based on image point cloud fusion[33–35]. And the higher the number of LiDAR channel, the richer the information obtained and the better the final result. Table 2 compares the cross-view pose estimation datasets described above.

Discussion of the labeling accuracy. Most existing datasets lack satellite maps that are directly aligned with ground-view images. As a result, researchers often rely on third-party self-labeled data. High-precision geographic coordinate alignment is essential for accurately matching ground-view and satellite-view images. In contrast, the McPed[23] provides a corresponding satellite map for each scene. Only one satellite map is required per scene, which can be verified using the path visualization method discussed earlier. If the visualized path does not align with the actual driving path, adjustments can be made by modifying the initial point's localization and angle. Multiple tests can be conducted to ensure the correct alignment. If necessary, further adjustments to the localization and angle of the initial point can be made to guarantee accuracy.

The study[36] critically evaluates the proposed path computation method in a park environment(not the dataset in this paper) to quantify its localization accuracy under motion conditions, and the detailed information of the experimental environment and localization error is shown in Table 3. It can be seen that our labeled method can achieve centimeter-level localization errors even with long path trajectories, which satisfies the requirement of centimeter-level labeling accuracy for cross-view position estimation.

Discussion of the impact of perspective differences. The geometric discrepancy between ground-view (GV) and BEV representations poses inherent challenges for cross-view feature matching. GV images (e.g., front-facing camera frames) capture lateral object surfaces (e.g., vehicle sides, building facades), while BEV projections (e.g., LiDAR point clouds or satellite maps) emphasize top-down structural layouts (e.g., road boundaries, rooftop outlines). The geometric discrepancy governed by:

$$\mathscr{T}_{\text{GV}\to\text{BEV}}\colon \mathbb{P}^2 \to \mathbb{R}^3 \quad \text{via} \quad \pi^{-1}(\mathbf{u}, \mathbf{K}, \mathbf{T}^{\text{cam}}_{\text{lidar}}) \tag{20}$$

where π^{-1} denotes the inverse perspective mapping, $\mathbf{K}$ the camera intrinsics, and $\mathbf{T}^{\text{cam}}_{\text{lidar}} \in SE(3)$ the LiDAR-camera extrinsic calibration. This perspective shift causes three practical issues in existing datasets: 1) Occlusion Mismatch: Ground-level occlusions (e.g., by vehicles) in GV (≈32% occurrence in urban scenes) do not correlate with BEV occlusions (e.g., by overpasses, ≈8% occurrence). 2) Projection Artifacts: Homography-based BEV synthesis introduces 14.2% pixel-level distortions, degrading downstream tasks like

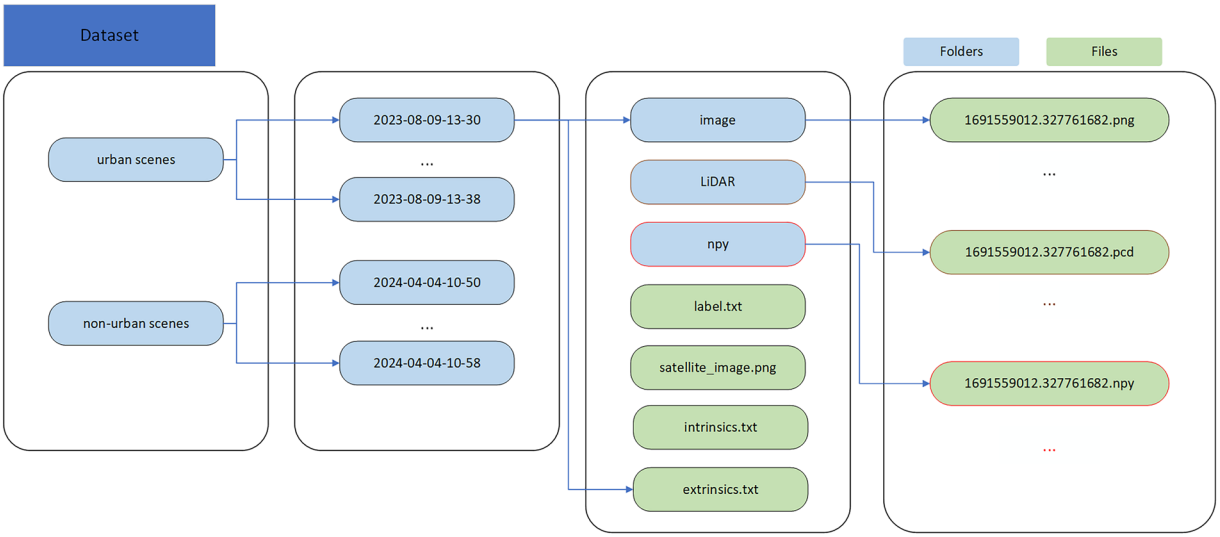

Fig. 7 McPed[23] Dataset file structure.

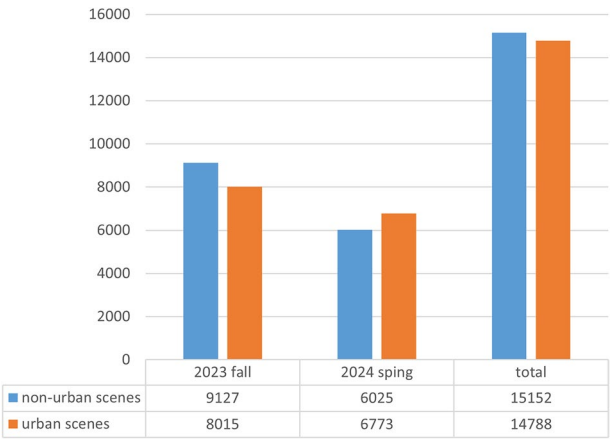

Fig. 8 Statistics of the number of different scenes included in the McPed[23] dataset.

semantic segmentation. 3) Feature Misalignment: Vertical structures (e.g., traffic signs) visible in GV are compressed into sparse points in BEV, while horizontal features (e.g., lane markings) suffer perspective distortion during projection. The differential Jacobian J_T amplifies local feature distortion:

$$\|J_T\|_F = \sqrt{\left(\frac{\partial x'}{\partial u}\right)^2 + \left(\frac{\partial y'}{\partial v}\right)^2 + \left(\frac{\partial z'}{\partial d}\right)^2} \geq 2.7 \quad \text{(empirical)} \tag{21}$$

We proposes a novel multi-modal fusion framework that integrates point cloud data with image information through projective geometry transformation. By utilizing the camera's intrinsic matrix and the extrinsics derived from joint camera-radar calibration, our method achieves accurate projection of chromatic information from RGB images onto the 3D point cloud space while preserving the inherent density and elevation attributes of the point cloud data. This synergistic approach effectively combines the complementary advantages of both modalities: maintaining the millimeter-level spatial precision and detailed 3D structural information from LiDAR measurements, while simultaneously incorporating rich photometric texture features from visual data. The resulting BEV representation overcomes traditional limitations by eliminating perspective distortion and chromatic aberration through rigorous sensor fusion, thereby establishing an optimal data foundation for subsequent feature extraction and matching processes. As demonstrated in Fig. 9, our comparative analysis reveals significant improvements in environmental perception quality. The visualization contrasts original ground-level imagery, raw point cloud data, projected BEV outputs, ground truth annotations, and prediction results from both CPC-CVPE and CCVPE[12] architectures, with satellite imagery serving as the geographic reference standard. This comprehensive validation confirms our method's superior performance in preserving geometric fidelity while enhancing semantic information density.

Dataset	LiDAR Model	N_{beams}	$\Delta\phi_h/\Delta\phi_v$	Scan Mode	$\rho@20m(pts/m^2)$
KITTI[18]	Velodyne HDL-64E	64	0.2°/0.4°	Mechanical	1,280
nuScenes[20]	Velodyne HDL-32E	32	0.16°/1.33°	Mechanical	420
Ford AV[19]	Velodyne HDL-32E	32	0.16°/1.33°	Mechanical	420
McPed[23]	Livox HAP	144(ROI)	0.18°/0.23°	Solid-state	3800

Table 1. Comparison of LiDAR performance with existing datasets.

Dataset	Sensors	Inclusion of satellite image	Verifiable methods (ground to air)	Scenes
VIGOR[15]	Camera	Yes	image to image	urban road
CVUSA[16]	Camera	Yes	image to image	urban road
CVACT[17]	Camera	Yes	image to image	urban road
KITTI[18]	Camera,64-line LiDAR	No	image & point cloud to image	urban road
FORD[19]	Camera,32-line LiDAR	No	image & point cloud to image	urban road
nuScenes[20]	Camera,32-line LiDAR	No	image & point cloud to image	urban road
McPed(ours)[23]	Camera,144-line LiDAR	Yes	image & point cloud to image	field & urban road

Table 2. Comparison of the existing datasets for cross-view pose estimation dataset. The type of sensor determines the type of task.

Metric	Park
Path Length	1,437 m
Max Angular Velocity	217.4°/s
Translation Error	0.04 m

Table 3. Environmental details and error analysis.

Discussion of scenes variety. Current autonomous driving datasets (e.g., KITTI[18], Argoverse[21]) exhibit constrained geographic and scenario coverage, predominantly sampling data from structured urban settings-specifically central business districts, arterial highways, and residential zones with standardized road geometries. This sampling bias introduces two critical limitations:1) Reduced Ecological Validity: The over-representation of homogeneous urban layouts (e.g., grid-like road networks, uniform building heights) fails to capture the spectral diversity of real-world environments, including rural dirt roads, mountainous switch-backs, and mixed-use transitional zones. 2) Algorithmic Overfitting: Cross-view pose estimation models trained on such data exhibit marked performance degradation when deployed in underrepresented scenarios, such as vegetation-obscured country lanes or unstructured industrial sites.

In contrast, the McPed[23] addresses these limitations by systematically incorporating both urban and non-urban environments ensuring a more comprehensive geographic representation. By capturing a broad spectrum of driving scenes, McPed[23] not only enhances the diversity of the data but also improves the robustness of model training. This multi-scenario data acquisition approach is instrumental in boosting the algorithm's generalization ability, enabling it to adapt to varied geographic conditions and complex terrains. As a result, models trained on this diverse dataset are better equipped to handle the challenges posed by real-world variations in road types, environmental features, and weather conditions.Representative scene distributions across benchmark datasets are visualized in Fig. 10.

Validation results of cross-view pose estimation. In order to test the usability and performance of McPed[23], this dataset will be compared in CVML[37] algorithm, CCVPE[12] algorithm, and CPC-CVPE, a cross-view pose estimation network that fuses point clouds and images. The first two directly match features from ground images to satellite maps to obtain localization information, where the CVML[37] algorithm only predicts location, so the experiment does not count its direction prediction. CPC-CVPE is feature-matching the result of fusing the image point clouds in this data. Table 4 demonstrates the quantization errors of the three algorithms in McPed[23].

From the prediction results, it can be seen that in the same algorithm comparison case, McPed[23] has a greater advantage in the non-urban scenes, the analysis can be seen that: the features in the non-urban scenes are single, the similarity is high, based on the fusion of the point cloud and the image of the BEV map to maintain the three-dimensional spatial structure at the same time, the complete retention of the texture and contour of the image data, to be able to extract the features of the environment without loss, better, so that with the satellite map for the feature matching is better. Unlike non-urban scenes, buildings are important factors for feature matching in urban scenes. Due to the location of the LiDAR placement, the collected data lacks the information of high buildings, so the lack of building point clouds is an important factor that leads to the reduced prediction accuracy of McPed[23] in urban scenes.

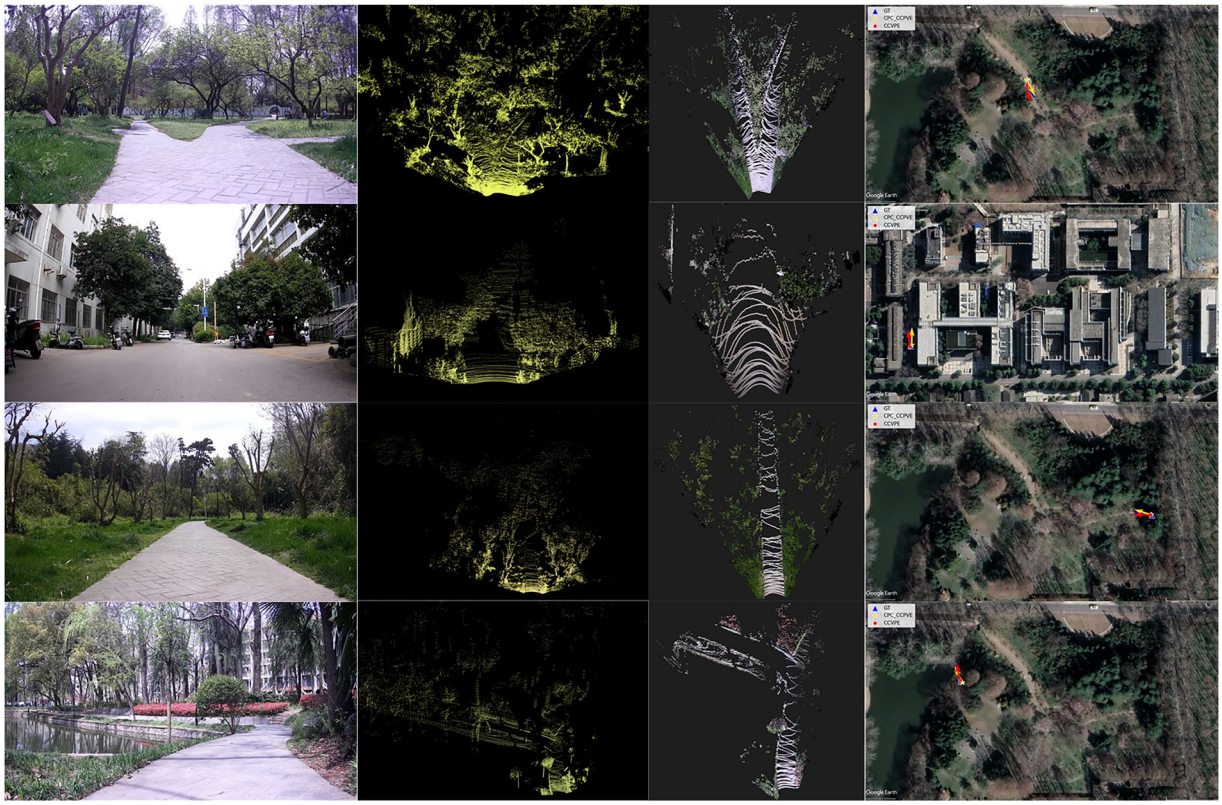

Fig. 9 From left to right, image, point cloud, BEV of color point cloud and prediction results of different methods of pose estimation.

We also conducted additional experiments on the KITTI[18] and Ford AV Datasets[19], the experimental results are shown in Tables 5 and 6. Notably, the LiDAR system in the Ford AV Datasets[19] consists of four Velodyne 32E sensors. To facilitate our experiments, we transformed their coordinate systems and fused the four 32-line LiDARs into an equivalent 128-line LiDAR. Based on this fused point cloud, we performed point cloud-image fusion and conducted experiments using the CPC-CVPE method. Due to the nature of the Ford AV Datasets[19], where most scenes consist of highways and other environments with similar features,it presents a more challenging benchmark for cross-view pose estimation. Therefore, the evaluation metrics are relatively lower. Our dataset demonstrates comparable accuracy to the KITTI[18] benchmark, thereby validating its feasibility for ground-to-air cross-view pose estimation applications.

Discussion of limitation. Due to the roof-mounted configuration of the LiDAR sensor on the field robot, the collected point clouds lack detailed structural information from building facades above 5 meters in height. This limitation arises from the fixed vertical field-of-view (FoV) of the LiDAR 25° downward tilt, which prioritizes ground-level feature capture for navigation tasks but omits upper building elements (e.g., windows, signage). The current dataset is exclusively collected under clear weather conditions. Adverse environments (e.g., rain, fog, snow) were not included due to the LiDAR's susceptibility to precipitation-induced noise and the camera's reduced visibility in low-light scenarios.

The platform's GNSS configuration warrants specific clarification: The exclusion of RTK positioning stems from its empirically documented unreliability in environments characterized by urban canyons (building height-to-street width ratio > 0.5) and dense arboreal coverage (canopy closure >70%)[38–41], where multipath signal reflections from building facades and vegetation-induced GNSS signal attenuation respectively induce decimeter-level ranging errors. In our experimental setup, the integration of RTK was constrained by the existing hardware architecture of the field robot system, which would necessitate additional specialized equipment beyond current configuration. However, we introduced the calculation method of the ground truth in Method, and proved that the method also achieves centimeter-level accuracy, which meets the accuracy requirements of this cross-view pose estimation task.

Usage Notes

The proposed McPed[23] is offered for use by academics for multi-modality cross-view pose estimation to facilitate the development of field robots navigation and even autonomous driving. This is the first multi-modality cross-view pose estimation dataset that can be used as a benchmark to facilitate the development of new algorithms. Because this dataset is aimed at cross-view pose estimation, we do not provide semantically related labels such as segmentation and detection. Researchers can annotate images or points accordingly for supervised learning.

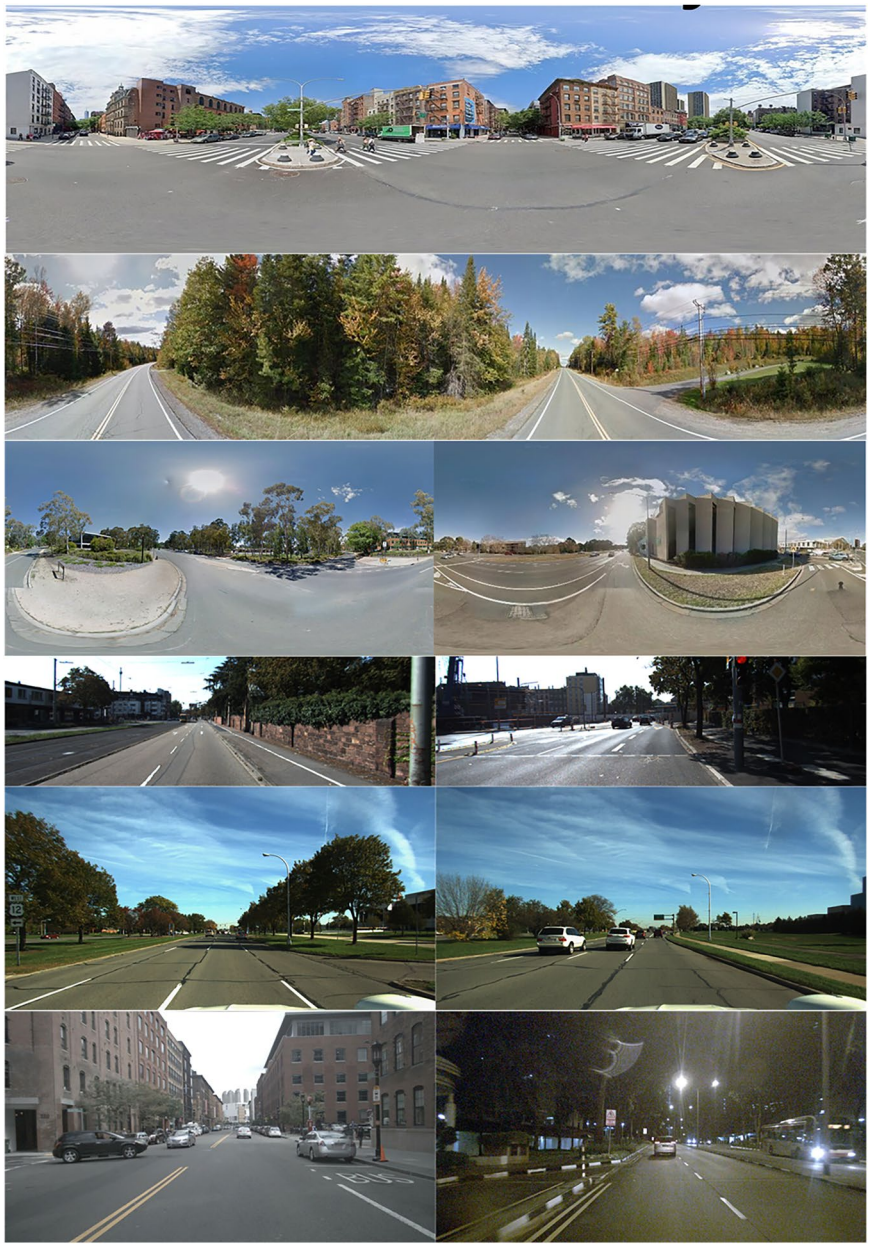

Fig. 10 Examples of typical scenes from VIGOR[15], CVUSA[16], CVACT[17], KITTI[18], FORD AV Dataset[19], nuScenes[20], from top to bottom,where VIGOR, CVUSA and CVACT are the panoramic image.

scenes	method	↓Location(m)		↑Location (%)		↓Heading(°)	
		mean	median	r@3m	r@5m	mean	median
non-urban scenes	CPC-CVPE(Image point cloud fusion)	**1.31**	0.47	**98.02**	**98.06**	**0.98**	**1.91**
	CCVPE[12](only image)	1.64	**0.34**	97.78	97.84	3.08	2.24
	CVML[37](only image)	1.84	1.11	85.23	95.52	—	—
urban scenes	CPC-CVPE(Image point cloud fusion)	2.28	0.88	92.63	95.57	**1.68**	**0.87**
	CCVPE[12](only image)	**1.63**	**0.78**	**98.50**	**99.12**	1.92	1.11
	CVML[37](only image)	3.41	2.64	55.83	79.72	—	—

Table 4. Performance comparison of different mainstream cross-view pose estimation algorithms. The distance error(m) column represents the absolute difference in distance between the predicted value and the ground truth, measured by the mean and the median, respectively, the distance error(%) column represents the percentage of the total number of predictions within 3 m and 5 m of the ground truth, and the angle error(°) column represents the absolute difference in angle between the predicted value and the ground truth, also measured by the mean and the median, respectively.

Type of ground data	Method	↓Location(m)		↑Lateral(%)		↑Longitudinal (%)		↓Heading(°)	
		mean	median	r@1m	r@5m	r@1m	r@5m	mean	median
Colored point cloud	CPC-CVPE	**1.03**	0.55	**97.87**	**99.79**	86.21	**97.98**	1.01	0.65
Image	CCVPE[12]	1.22	0.62	97.35	99.71	77.13	97.16	0.67	0.54
	LDFF[43]	1.48	**0.47**	95.47	**99.79**	**87.89**	94.78	**0.49**	**0.30**
	SliceMatch[44]	7.96	4.39	49.09	98.52	15.19	57.35	4.12	3.65
	LM[45]	12.08	11.42	35.54	80.36	5.22	26.13	3.72	2.83
	DSM[46]	—	—	10.12	48.24	4.08	20.14	—	—
Point cloud	RSL-Net[33]	3.71	—	—	—	—	—	1.67	—
	Tang[34]	4.37	—	—	—	—	—	1.59	—
	Tang[35]	3.76	—	—	—	—	—	3.36	—

Table 5. Pose estimation results on the KITTI[18] Dataset. Bold numbers present the bset score.

Area	Method	↓Location(m)		↑Lateral(%)		↑Longitudinal(%)		↓Heading (°)	
		mean	median	r@1m	r@5m	r@1m	r@5m	mean	median
Log1	CPC-CVPE	**9.67**	**9.78**	59.31	95.12	**6.77**	26.12	2.47	2.48
	LDFF[43]	10.55	10.19	**68.67**	**95.48**	5.48	25.86	1.43	**0.47**
	CCVPE[12]	11.34	10.21	51.19	92.36	6.69	**30.00**	**1.28**	0.93
	LM[45]	12.54	12.63	48.57	71.57	5.90	26.33	3.13	1.29
	DSM[46]	—	—	12.00	53.67	4.33	21.43	—	—

Table 6. Pose estimation results on the Ford AV Dataset[19]. Bold numbers present the bset score.

Code availability

We provide a development kit programmed with Python and C++ language for this dataset, which contains scripts for visualizing and parsing the dataset. The toolkit is available at the code repository[42]. We provide some example files containing images, point clouds, npy files, internal and external parameter calibration files. The *color_point.py* provides the function of projecting the image RGB information into the point cloud through the camera internal reference and the external reference between LiDAR and camera to get the visualized color point cloud and get the corresponding npy preview results. The *save_npy.py* is used to generate and save the npy file,elevation and density maps can be generated separately.The *path_visualization.cpp* and *path_visualization.h* provide a path visualization method to check the accuracy of annotations.

Received: 12 November 2024; Accepted: 28 April 2025;
Published online: 07 May 2025

References

1. Chen, L. *et al.* Milestones in autonomous driving and intelligent vehicles: Survey of surveys. *IEEE Transactions on Intelligent Vehicles* **8**, 1046–1056, https://doi.org/10.1109/TIV.2022.3223131 (2023).
2. Yurtsever, E., Lambert, J., Carballo, A. & Takeda, K. A survey of autonomous driving: Common practices and emerging technologies. *IEEE access* **8**, 58443–58469 (2020).
3. Dong, H. *et al.* Gpsmirror: Expanding accurate gps positioning to shadowed and indoor regions with backscatter. In *Proceedings of the 29th Annual International Conference on Mobile Computing and Networking*, MobiCom '23, https://doi.org/10.1145/3570361.3592511 (ACM, 2023).
4. Li, X. *et al.* Principle and performance of multi-frequency and multi-gnss ppp-rtk. *Satellite Navigation* **3**, 7 (2022).
5. Davison, A. J., Reid, I. D., Molton, N. D. & Stasse, O. Monoslam: Real-time single camera slam. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **29**, 1052–1067, https://doi.org/10.1109/TPAMI.2007.1049 (2007).
6. Lipson, L., Teed, Z. & Deng, J. Deep Patch Visual SLAM. In *European Conference on Computer Vision* (2024).
7. Lowe, D. G. Object recognition from local scale-invariant features. In *Proceedings of the seventh IEEE international conference on computer vision*, vol. 2, 1150–1157 (Ieee, 1999).
8. Alcantarilla, P. F., Bartoli, A. & Davison, A. J. Kaze features. In *Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part VI 12*, 214–227 (Springer, 2012).
9. Rosten, E. & Drummond, T. Machine learning for high-speed corner detection. In *Computer Vision–ECCV 2006: 9th European Conference on Computer Vision, Graz, Austria, May 7-13, 2006. Proceedings, Part I 9*, 430–443 (Springer, 2006).
10. Shi, Y., Liu, L., Yu, X. & Li, H. Spatial-aware feature aggregation for image based cross-view geo-localization. *Advances in Neural Information Processing Systems***32** (2019).
11. Zhan, Y., Xiong, Z. & Yuan, Y. Rsvg: Exploring data and models for visual grounding on remote sensing data. *IEEE Transactions on Geoscience and Remote Sensing* **61**, 1–13 (2023).
12. Xia, Z., Booij, O. & Kooij, J. F. Convolutional cross-view pose estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2023).
13. Shi, Y. *et al.* Accurate 3-dof camera geo-localization via ground-to-satellite image matching. *IEEE transactions on pattern analysis and machine intelligence* **45**, 2682–2697 (2022).
14. Wang, S., Zhang, Y., Vora, A., Perincherry, A. & Li, H. Satellite image based cross-view localization for autonomous vehicle. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 3592–3599 (IEEE, 2023).
15. Zhu, S., Yang, T. & Chen, C. Vigor: Cross-view image geo-localization beyond one-to-one retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3640–3649 (2021).

16. Workman, S., Souvenir, R. & Jacobs, N. Wide-area image geolocalization with aerial reference imagery. In *Proceedings of the IEEE International Conference on Computer Vision*, 3961–3969 (2015).
17. Liu, L. & Li, H. Lending orientation to neural networks for cross-view geo-localization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 5624–5633 (2019).
18. Geiger, A., Lenz, P. & Urtasun, R. Are we ready for autonomous driving? the kitti vision benchmark suite. In *2012 IEEE conference on computer vision and pattern recognition*, 3354–3361 (IEEE, 2012).
19. Maddern, W., Pascoe, G., Linegar, C. & Newman, P. 1 year, 1000 km: The oxford robotcar dataset. *The International Journal of Robotics Research* **36**, 3–15 (2017).
20. Caesar, H. *et al*. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 11621–11631 (2020).
21. Chang, M. *et al*. Argoverse: 3d tracking and forecasting with rich maps. *CoRR* abs/1911.02620 (2019). 1911.02620.
22. Roriz, R., Cabral, J. & Gomes, T. Automotive lidar technology: A survey. *IEEE Transactions on Intelligent Transportation Systems* **23**, 6282–6297, https://doi.org/10.1109/TITS.2021.3086804 (2022).
23. Yuan, X. & Wang, K. A multi-modality ground-to-air cross-view pose estimation dataset for field robots. https://doi.org/10.6084/m9.figshare.28528868 (2024).
24. Ribeiro, T. *et al*. Development of a prototype robot for transportation within industrial environments. In *2017 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*, 192–197, https://doi.org/10.1109/ICARSC.2017.7964074 (2017).
25. Liu, H., Stoll, N., Junginger, S. & Thurow, K. Mobile robotic transportation in laboratory automation: Multi-robot control, robot-door integration and robot-human interaction. In *2014 IEEE International Conference on Robotics and Biomimetics (ROBIO 2014)*, 1033–1038, https://doi.org/10.1109/ROBIO.2014.7090468 (2014).
26. English, A., Ross, P., Ball, D. & Corke, P. Vision based guidance for robot navigation in agriculture. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, 1693–1698, https://doi.org/10.1109/ICRA.2014.6907079 (2014).
27. Oliveira, L. F. P., Moreira, A. P. & Silva, M. F. Advances in agriculture robotics: A state-of-the-art review and challenges ahead. *Robotics* **10**, https://doi.org/10.3390/robotics10020052 (2021).
28. Ismail, R. M., Muthukumaraswamy, S. & Sasikala, A. Military support and rescue robot. In *2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS)*, 156–162, https://doi.org/10.1109/ICICCS48265.2020.9121041 (2020).
29. Patil, D. *et al*. A survey on autonomous military service robot. In *2020 International Conference on Emerging Trends in Information Technology and Engineering (ic-ETITE)*, 1–7, https://doi.org/10.1109/ic-ETITE47903.2020.78 (2020).
30. Li, S., Tu, Z., Chen, Y. & Yu, T. Multi-scale attention encoder for street-to-aerial image geo-localization. *CAAI Transactions on Intelligence Technology* **8**, 166–176 (2023).
31. Wang, T. *et al*. Each part matters: Local patterns facilitate cross-view geo-localization. *IEEE Transactions on Circuits and Systems for Video Technology* **32**, 867–879 (2021).
32. Vo, N. N. & Hays, J. Localizing and orienting street views using overhead imagery. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*, 494–509 (Springer, 2016).
33. Tang, T. Y., De Martini, D., Barnes, D. & Newman, P. Rsl-net: Localising in satellite images from a radar on the ground. *IEEE Robotics and Automation Letters* **5**, 1087–1094 (2020).
34. Tang, T. Y., De Martini, D., Wu, S. & Newman, P. Self-supervised learning for using overhead imagery as maps in outdoor range sensor localization. *The International Journal of Robotics Research* **40**, 1488–1509 (2021).
35. Tang, T. Y., De Martini, D. & Newman, P. Get to the point: Learning lidar place recognition and metric localisation using overhead imagery. *Proceedings of Robotics: Science and Systems, 2021* (2021).
36. Shan, T. *et al*. Lio-sam: Tightly-coupled lidar inertial odometry via smoothing and mapping. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 5135–5142, https://doi.org/10.1109/IROS45743.2020.9341176 (2020).
37. Xia, Z., Booij, O., Manfredi, M. & Kooij, J. F. Visual cross-view metric localization with dense uncertainty estimates. In *European Conference on Computer Vision*, 90–106 (Springer, 2022).
38. Groves, P. D. Shadow matching: A new gnss positioning technique for urban canyons. *Journal of Navigation* **64**, 417–430, https://doi.org/10.1017/S0373463311000087 (2011).
39. Chen, W. *et al*. Enhancing gnss positioning in urban canyon areas via a modified design matrix approach. *IEEE Internet of Things Journal* **11**, 10252–10265, https://doi.org/10.1109/JIOT.2023.3326971 (2024).
40. Andreas, H., Abidin, H. Z., Sarsito, D. & Pradipta, D. Study the capabilities of rtk multi gnss under forest canopy in regions of indonesia. *E3S Web of Conferences* **94**, 01021, https://doi.org/10.1051/e3sconf/20199401021 (2019).
41. Ng, H.-F. & Hsu, L.-T. 3d mapping database-aided gnss rtk and its assessments in urban canyons. *IEEE Transactions on Aerospace and Electronic Systems* **57**, 3150–3166, https://doi.org/10.1109/TAES.2021.3069271 (2021).
42. Yuan, X. & Wang, K. Robvislab-njust mcped_dev_toolkit. https://doi.org/10.5281/zenodo.15083001 (2024).
43. Song, Z., Ze, X., Lu, J. & Shi, Y. Learning dense flow field for highly-accurate cross-view camera localization. *Advances in Neural Information Processing Systems***36** (2023).
44. Lentsch, T., Xia, Z., Caesar, H. & Kooij, J. F. Slicematch: Geometry-guided aggregation for cross-view pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 17225–17234 (2023).
45. Shi, Y. & Li, H. Beyond cross-view image retrieval: Highly accurate vehicle localization using satellite image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 17010–17020 (2022).
46. Shi, Y., Yu, X., Campbell, D. & Li, H. Where am i looking at? joint location and orientation estimation by cross-view matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4064–4072 (2020).

Author contributions

X.Y., K.W. and J.X. designed the research. K.W. and R.Q. collected data. K.W. and J.X. did verification and calibration. J.X. solved the problem of data synchronization. K.W. and R.Q did annotation. R.Q. conducted the CPC-CVPE experiments and analyzed the results. All authors reviewed the manuscript.

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to X.Y.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.