Scalable Wi-Fi RSS-Based Indoor Localization via Automatic Vision-Assisted Calibration

Abdulkadir Bilge, Erdem Ergen, Burak Soner, Sinem Çöleri
Koç University
Istanbul, Türkiye
{abilge20, eergen20, bsoner, scoleri}@ku.edu.tr

***Abstract*—Wi-Fi-based positioning promises a scalable and privacy-preserving solution for location-based services in indoor environments such as malls, airports, and campuses. RSS-based methods are widely deployable as RSS data is available on all Wi-Fi-capable devices, but RSS is highly sensitive to multipath, channel variations, and receiver characteristics. While supervised learning methods offer improved robustness, they require large amounts of labeled data, which is often costly to obtain. We introduce a lightweight framework that solves this by automating high-resolution synchronized RSS-location data collection using a short, camera-assisted calibration phase. An overhead camera is calibrated only once with ArUco markers and then tracks a device collecting RSS data from broadcast packets of nearby access points across Wi-Fi channels. The resulting (x, y, RSS) dataset is used to automatically train mobile-deployable localization algorithms, avoiding the privacy concerns of continuous video monitoring. We quantify the accuracy limits of such vision-assisted RSS data collection under key factors such as tracking precision and label synchronization. Using the collected experimental data, we benchmark traditional and supervised learning approaches under varying signal conditions and device types, demonstrating improved accuracy and generalization, validating the utility of the proposed framework for practical use. All code, tools, and datasets are released as open source.**

***Index Terms*—Wi-Fi positioning, received signal strength (RSS), vision-based calibration, robust localization, deep learning.**

I. Introduction

Indoor localization supports many applications in retail, airport navigation, and security, especially when individuals or assets with mobile devices require self-localization [1]. Among available methods, Wi-Fi-based positioning using signals from stationary access points (APs) is particularly popular due to its passive nature and the ubiquity of Wi-Fi infrastructure. Unlike camera-based systems, it also preserves privacy since user Wi-Fi devices are uniquely identifiable without personal information, avoiding the trade-offs between spatial resolution and privacy concerns [2].

Wi-Fi based localization methods use different characteristics of the Wi-Fi signal: the received signal strength (RSS), the channel state information (CSI), or signal round-trip-time (RTT). CSI and RTT approaches use detailed physical-layer data such as round-trip time [3] and per-subcarrier complex phase-amplitude characteristics [4], and can achieve even sub-centimeter accuracy in controlled environments [5, 6]. However, their deployment is limited due to hardware constraints: CSI and RTT data are not accessible on most commercial devices without specialized tools such as Nexmon [7] or Intel CSI Tool [8], or custom firmware [9]. Among these methods, RSS remains the most practical signal type for localization as it is available on virtually all devices [10].

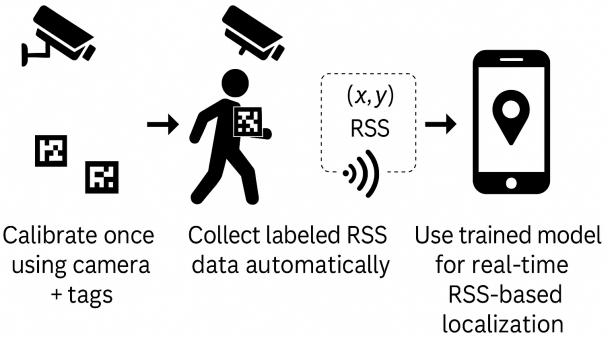

Fig. 1. One-time calibration with visual object trackers and markers allows automatically collecting labeled RSS-location data without tedious manual annotation. Ample data allows trained models to generalize with high accuracy, unlocking scalable localization using RSS alone.

Despite its practicality, RSS-based localization suffers from significant robustness issues. Traditional fingerprinting and geometric methods (e.g., RSS signal model-based triangulation) are affected by environmental changes, device variability [9, 10], and especially multipath effects. Since Wi-Fi systems are designed for coverage rather than directionality, the functional relationship between RSS and location is often weak and unreliable. Dynamic factors such as human motion or layout changes further degrade performance and require frequent recalibration. Data-driven methods using neural networks and large training datasets have been proposed to improve robustness [11], but they often involve costly data collection and generalize poorly across settings—such as between rooms or across different devices. To address these limitations, recent work has explored auxiliary sensing methods like vision-based tracking for automated calibration. Examples include webcam- or smartphone-based self-labeling [12, 13], but these lack a complete analysis of how calibration quality impacts localization performance.

In this work, we present a lightweight, vision-assisted RSS based localization framework that automates data collection using a calibrated overhead camera (e.g., a ceiling-mounted security camera). The system uses ArUco markers for initial calibration and tracks a marker-attached device during a short

data collection phase, producing synchronized RSS and location pairs at high spatial resolution. Using this setup, we collect experimental datasets and benchmark a range of localization models, including both traditional and supervised learning-based methods. We systematically evaluate how well these models generalize under different conditions, such as varying signal strengths across environments and receiver device types. The system is modular, practical, and open-source, designed for easy deployment and reproducible benchmarking of RSS-based localization algorithms.

II. Related Work

Two main families of traditional RSS-based indoor localization methods are fingerprinting and geometric approaches. Fingerprinting relies on an offline phase where RSS values are collected at known locations and an online phase where real-time measurements are matched to this database. A common technique is nearest neighbor classification in the raw RSS feature space. While this approach can offer robustness when the data is rich, it suffers from coarse spatial resolution, as dense fingerprinting requires labor-intensive site surveys.

In contrast, geometric methods estimate the location of a device by modeling signal propagation directly (e.g., finding position fixes at the intersections of modeled "equi-RSS" hyperbola curves). These methods can theoretically provide fine-grained localization, but perform poorly in Wi-Fi environments, where access points are designed for omnidirectional coverage and heavily exploit multipath. Consequently, assumptions about equi-distance signal contours break down under multipath and non-line-of-sight conditions. Hybrid approaches that interpolate between sparse fingerprint locations using signal propagation models have also been proposed, but these are difficult to scale across environments due to variability in signal characteristics.

Recent supervised learning–based methods aim to address this tradeoff between robustness and resolution, as depicted in Fig. 2. One class of approaches still performs classification, but in a learned feature space rather than on raw RSS [14]. Another class performs regression from RSS measurements to physical coordinates. These models, which also recently explore using modern architectures such as Transformers [15] and prototypical networks [16], can achieve higher spatial resolution, but require large amounts of labeled data, which is expensive to collect. Our proposed framework directly addresses this challenge by automating the collection of high-resolution labeled data via vision-assisted calibration.

III. Proposed Framework

Our framework automates high-resolution RSS–location data collection and end-to-end model training using a short, vision-assisted calibration. The setup involves two synchronized components: (i) an overhead camera for ground-truth location labeling, and (ii) an RSS collector operating in Wi-Fi monitor (a.k.a. promiscuous) mode. The vision pipeline is discarded after the calibration phase, and only the RSS data on end-user devices are used for positioning.

The camera is calibrated once, using ArUco markers placed at known positions, with the standard Perspective-n-Point (PnP) algorithm [17]. During data collection, the (x,y) location of the person carrying the RSS collector is tracked using a YOLOv8+SORT pipeline (carries an ArUco marker for identification), and the RSS collector captures broadcast Wi-Fi packets in monitor mode at 10 Hz (AP broadcast messages), channel hopping across Wi-Fi bands to capture rich multipath characteristics.

We apply a moving average filter with a fixed-size window to both RSS and (x,y) streams, as shown in Fig. 3, to improve correlation between RSS and location data (e.g., correlation coefficients are 2x higher for a window of 50 samples). However, this also bounds the user to slow movements (both during calibration and deployment) since with a sampling rate of 10 Hz for RSS, a window of e.g., $N{=}50$ samples means 5 seconds of movement is averaged. While much smaller windows can also be used for capturing faster movements, this slow-sampling problem of RSS-based position estimation due to slowly repeating broadcast messages is well-known [1].

After data acquisition, the framework trains localization models using the collected data. These include traditional nearest-neighbor classifiers, signal model-based interpolation approaches, and supervised learning models. Since the filtering step described above improves correlation, using convolutional neural networks (CNN), which are basically nonlinear combinations of such FIR filters, is a sensible choice while benchmarking supervised learning methods: A compact 1-D

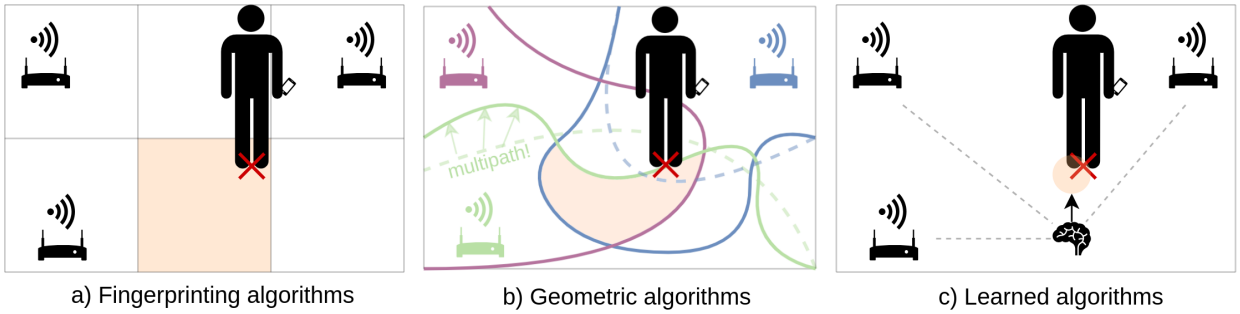

Fig. 2. RSS-based localization: (a) fingerprinting methods predict the nearest neighbor, resulting in coarse resolution; (b) geometric methods rely on signal models and are sensitive to multipath; (c) data-driven methods learn robust mappings from RSS to location across devices and environments.

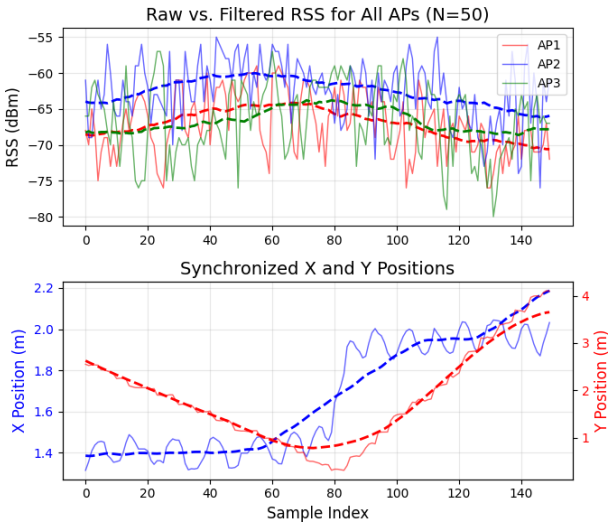

Fig. 3. Filtering collected data for higher correlation between RSS (top) and location (x, y) data (bottom). Dashed lines indicate filtered versions.

CNN over fixed-length windows captures these local temporal dependencies with a small number of parameters, making them easier to deploy on mobile platforms compared to other sophisticated architectures such as LSTMs and Transformers. Moreover, since CNNs are stationary models in time (i.e., inference at each time step is independent of inference outputs in other timesteps, the model has no memory), we can randomize the few training samples we have and remove the bias of specific motion profiles, achieving better generalization. Trained models are then exported for inference on end-user mobile devices, which require only RSS input for real-time localization—no camera, markers, or additional infrastructure.

The framework, depicted in Fig. 4, includes user-friendly graphical interfaces for calibration, data collection and training. This enables rapid and automated deployment of robust localization models for any environment, such as against varying crowd densities and different receiver devices.

IV. Accuracy Analysis

We quantify the accuracy limits of our vision-assisted calibration pipeline, which bound the performance of any RSS-based model trained on its labels. The analysis covers two aspects: (i) *spatial* errors from pixel-to-world mapping and (ii) *temporal* errors from synchronizing RSS with visual labels.

A. Spatial Attributes

Let σ_t be the standard deviation of the estimation error between the true and estimated 2-D position on the floor plane (i.e., perpendicular projection). We model the total spatial error variance[1] as the sum of four independent components:

$$\sigma_t^2 = \sigma_{\text{px}}^2 + \sigma_{\text{tag}}^2 + \sigma_{\text{det}}^2 + \sigma_{\text{foot}}^2, \quad (1)$$

[1]The effect of device height and other minor effects like condition number of the homography matrix are ignored here since they are negligible compared to the other terms considered for the proposed setup.

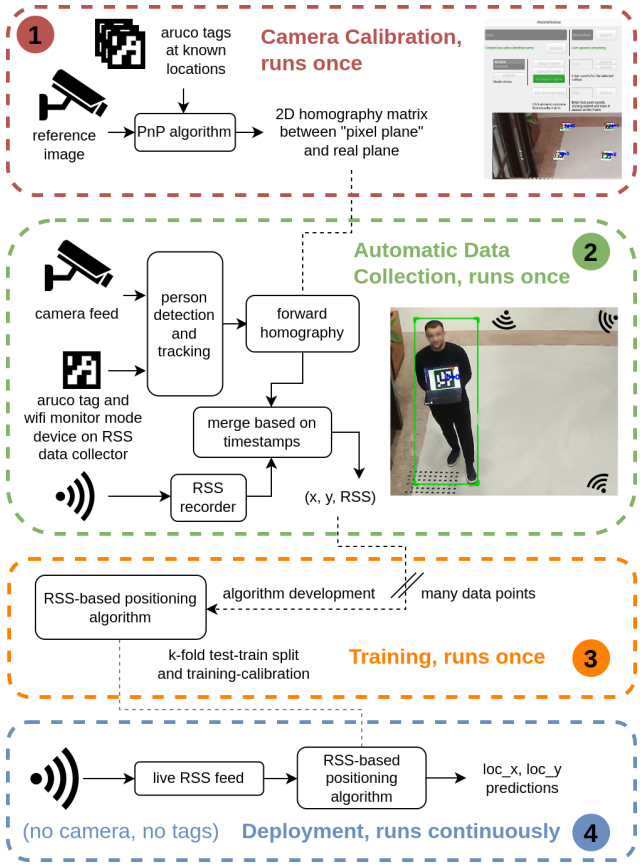

Fig. 4. System architecture of the proposed framework. The process consists of four stages: (1) one-time camera calibration, (2) automatic data collection with synchronized RSS and visual tracking, (3) algorithm training, and (4) continuous real-time localization using only RSS input.

where the individual terms are defined as follows:

- *Pixel–scale quantization* σ_{px} captures the limit imposed by finite sensor resolution; each pixel subtends a fixed ground-plane distance determined by camera height and field of view.
- *Tag-placement error* σ_{tag} arises because ArUco markers cannot be positioned exactly at their nominal locations. The resulting perturbation propagates through the PnP solution and affects every reconstructed point.
- *Detector–tracker variance* σ_{det} reflects frame-to-frame jitter of the YOLOv8 detector and SORT tracker, measured empirically from annotated video frames (sampling in our setup showed this to be ≈ 3 px in most scenarios).
- *Foot-point/ground-plane heuristic* σ_{foot}: The framework projects the lower midpoint of the bounding box onto the floor plane and assumes that is the device position. Camera angles and user movements induces a systematic error in the location label.

The four variance terms in (1) were simulated (10k Monte-Carlo iterations) for different settings to characterize these

spatial error attributes. We provide the source code and details for this simulator in our open source repository. Table I summarizes the results for our *baseline* installation (3m camera height, 5m vertical field of view, tag misplacement σ_{tag} =0.10m) and for two enlarged-FoV scenarios that illustrate how localization accuracy degrades with both camera scale and careless marker placement.

For the baseline installation the combined spatial label uncertainty is $\sigma_t \approx 9.7$ cm, well below the meter-level RMSE reported by state-of-the-art RSS localization systems [11]. For other configurations, such as when the camera FoV is increased, the homography-related sensitivities dominate, raising σ_t to 27 cm. If, in addition, marker placement errors are increased to 30 cm variance, the tag-placement term becomes the principal contributor and the overall uncertainty rises to 76 cm. These results confirm that moderate tag placement errors around 5-10 cm do not cause critical label uncertainty levels, but large placement errors and low pixel-density camera FoV do cause the label errors to go up to the meter level.

B. Temporal Attributes

Let Δt be the time misalignment between a camera frame and its nearest RSS sample (in time). If the device moves at speed v, the induced spatial error is $\varepsilon_{\text{temp}} = v\,\Delta t$. We can model Δt as the maximum of the three dominant latencies

$$\Delta t = \max\{1/f_{\text{cam}},\, 1/f_{\text{rss}},\, \Delta t_{\text{align}}\},$$

where $f_{\text{cam}} = 30$ Hz, $f_{\text{rss}} \approx 10$ Hz (broadcast messages) and Δt_{align} is the clock alignment error between the RSS collector and the vision-assistant collecting the location data. Δt_{align} is bounded to less than 10 ms by using a Network Time Protocol (NTP) provider to match the clocks of the two devices. Then, to bound this timing-induced error, $\varepsilon_{\text{temp}}$, to less than 5 cm, we need to impose $v \leq 0.5$ m/s during calibration. A similar latency budget applies at deployment, since methods will need to aggregate at least 5-10 RSS samples before updating the user position, necessitating users to remain static for at least a few seconds [1].

C. Combined Label Uncertainty

Combining (1) with $\varepsilon_{\text{temp}}$ gives the overall 1-σ label uncertainty for the data the proposed setup produces:

$$\sigma_{\text{label}} = \sqrt{\sigma_t^2 + \varepsilon_{\text{temp}}^2}\ ,$$

which serves as an upper bound on the achievable localization error for any algorithm trained using these labels. As shown in the simulations above, for the base configuration we considered, this error is smaller than 10 cm.

V. Experimental Results

Four independent recordings were collected with the "Base" camera configuration from Table I and three commercial off-the-shelf routers configured as APs, positioned around an approximately rectangular region of 4m x 6m, centered at the camera FoV. The recordings are called Experiments #5-#8, where #5–#7 ($883, 1865, 2317$ labeled samples) were captured with receiver A (Intel - iwlwifi), while Exp. #8 (1092 samples) used receiver B (MediaTek - mt7921e), introducing receiver and OS driver variations to the dataset alongside the RSS collector and bystander movement variations present in each recording. We first use this dataset to analyze our framework and the RSS-based localization problem by training simple CNNs [2] in different evaluation runs, and then we compare the CNN with two traditional methods to provide a benchmark[3].

TABLE I
Simulated label uncertainty due to spatial attributes

Scenario	σ_{px} [m]	σ_{tag} [m]	σ_{det} [m]	σ_{foot} [m]	σ_t [m]
Base (5 m FoV, 10 cm tags)	0.0023	0.050	0.014	0.082	**0.097**
Large FoV (25 m, 10 cm tags)	0.0116	0.251	0.069	0.082	**0.273**
Large FoV (25 m, 30 cm tags)	0.0116	0.755	0.069	0.082	**0.762**

1) **kNN**: nearest-neighbor fingerprinting on raw RSS,
2) **kNN+Interpolation**: weighted sum of M closest kNN estimations, weights are Euclidean distance to the input RSS vector, thereby assuming a $1/r^2$ signal model where r is distance from the AP.
3) **CNN**: the lightweight 1-D convolutional network.

We report three evaluation runs (R#):

- **R1) Data–efficiency:** The CNN is trained on random 5% – 95% train-test splits of the whole dataset (all 4 recordings) to understand how much data collection will be necessary during calibration for proper generalization. Results show that only a few minutes of calibration per configuration gives the best possible result; an error under the label uncertainty level.
- **R2) Generalization:** The CNN is trained on 3 out of 4 recordings, and is tested on the 4th, cycled through all combinations. Results show that the CNN does generalize well to the y axis in similar settings (e.g., Exp. 5-6-7), but fails to provide accurate estimations for the x axis.
- **R3) Method comparison:** kNN, kNN+Interp and CNN are compared on the total dataset, where the CNN is the clear overall winner.

A. R1: Data–efficiency

We merge the 4 recordings and then train-test the CNN on random 5% – 95% train-test splits of the dataset. Figure 5 shows the mean error on the test set with the curve and the standard deviation with the shaded error region, averaged over the 4 experiments versus the fraction of the train-test split. Results show that with only ≈150 seconds of calibration (1500 labels) the CNN already attains a result very close to the best possible performance. Note that the label uncertainty for

[2]The CNN: 4 layers with bias and ReLU, kernel sizes are {13, 11, 9, 7} and channel sizes are {8, 16, 8, 2}; final layer produces x and y predictions.

[3]All data, scripts, and trained models are available in our public repository: https://github.com/sonebu/wifi_rss_positioning_visioncalib

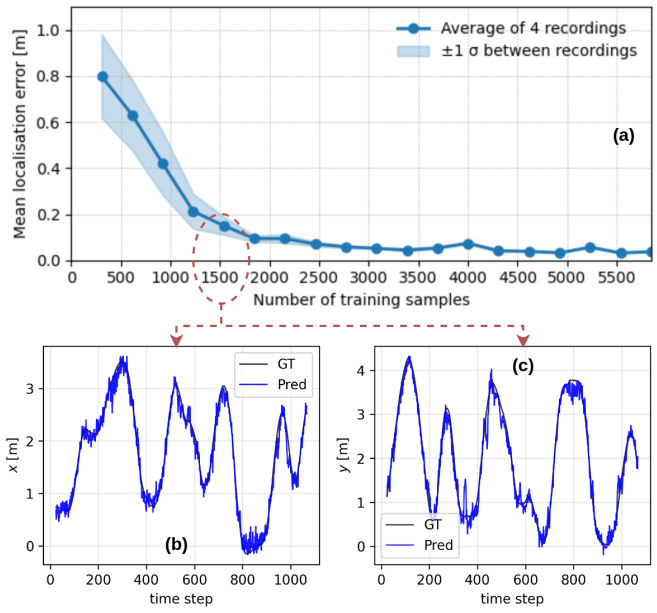

Fig. 5. Results in (a) show CNN performance vs. number of labeled samples, averaged over all 4 recordings. Results in (b) and (c) show the x and y estimation performance of the CNN trained with just 1500 samples, on Experiment #8, the case with receiver B.

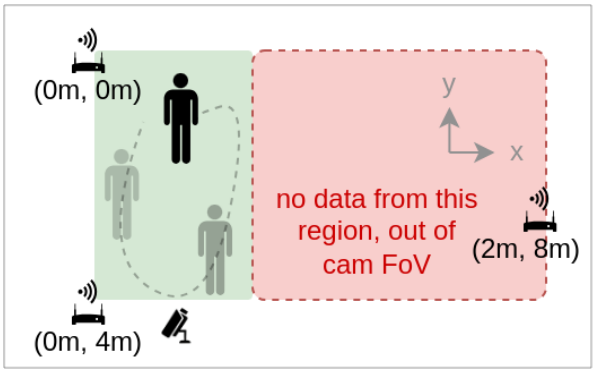

Fig. 6. AP placement and RSS collector movement profile during calibration in the experimental setup. Limited coverage in the x axis decreased modeling performance.

this configuration is around 10 cm, so after ≈1500 samples of calibration the error would be dominated by the label uncertainty.

B. R2: Cross-recording generalization

We train the CNN on 3 out of 4 recordings and test it on the 4th, for all combinations, to show the generalization performance of the data collected and the CNN. The results in Figure 7 show that although overall generalization accuracy across unseen domains is not as high as the in-domain case shown in Fig. 5, it is sufficient for sub-1m estimation, and it is much better for the y axis compared to the x axis. This stems from asymmetric nature of the AP placement in the experimental setup versus the movement profile chosen for calibration. As shown in Figure 6, while the RSS collector

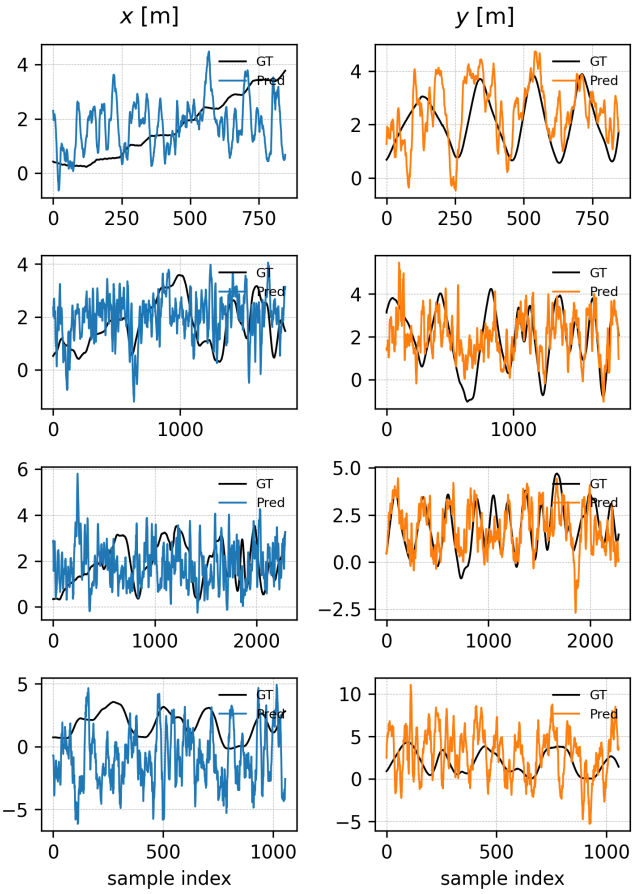

Fig. 7. Generalization test: The CNN is trained on 3 out of 4 recordings and evaluated on the 4th (rows show different hold-out combinations, Experiments #5 to #8, respectively). Left column shows the performance in x and right column shows the performance in y.

moves over the whole y range during calibration, the x range is not covered as much because of limited camera FoV, resulting in lower sensitivity of the RSS data towards x movements. This highlights the importance of choosing movement profiles during calibration; RSS collectors should cover the whole region defined by the APs in order to produce proper calibration data. Moreover, the effect of receiver variation is also observed; Generalization to Exp. #8 is not as good as in the other Experiments since there are fewer samples recorded with receiver B, highlighting the importance of using balanced datasets with a few different receiver devices during calibration.

C. R3: Benchmark, Traditional vs. Learning-based Methods

We compare the lightweight CNN with two classical RSS-fingerprinting baselines on the full dataset. Table II reports the mean and standard deviation of the L_2 error on each recording. Plain kNN attains $\sim 2\,\text{m}$ median accuracy, which improves only marginally when the inverse-square interpolation model is added. In contrast, the CNN pushes the error below 15 cm in all cases (< 10 cm on receiver A). These results confirm that once high-quality labels are available, such as with the

TABLE II
TRADITIONAL VS LEARNING-BASED METHODS, MEAN ± STD. DEVIATION.

ID	kNN	kNN+Interp	**CNN (ours)**
Exp-5	1.884 ± 0.981 m	1.597 ± 0.793 m	**0.094 ± 0.093 m**
Exp-6	1.874 ± 1.103 m	1.585 ± 0.885 m	**0.097 ± 0.093 m**
Exp-7	1.865 ± 1.003 m	1.584 ± 0.838 m	**0.097 ± 0.090 m**
Exp-8	2.376 ± 1.074 m	2.037 ± 0.961 m	**0.144 ± 0.150 m**

proposed data collection and calibration framework, even a lightweight network can deliver cm-scale positioning performance with substantially better generalization than traditional methods in the literature.

D. Limitations

We recognize the following limitations in this study: Our evaluation spans a single site with three AP layouts and two different receivers, but broader datasets (device diversity, camera FoV/lighting, crowd density, and edge-case motion profiles) would enhance applicability. We benchmark a compact 1-D CNN for deployability with limited data, but richer sequence/attention models and unsupervised/self-supervised adaptation may further improve cross-device/site generalization when more data/compute are available. Moreover, long-term drift and AP churn were not studied, but lightweight online adaptation or periodic re-calibration approaches can be investigated. These are left to future work.

VI. CONCLUSION

We presented a lightweight framework that combines a one-time, vision-assisted calibration with monitor-mode RSS collection to generate cm-accurate location labels at scale for Wi-Fi RSS-based positioning. Fundamental analyses showed that the spatial–temporal uncertainty introduced by the camera, marker placement and label synchronization remains below 10 cm. Using the collected labels, we trained a compact CNN and demonstrated that it generalizes much more accurately with high-quality data compared to existing methods (e.g., kNN + $1/r^2$ interpolation) for unseen recordings and different receivers. These results confirm that high-quality labels provided by our lightweight framework are key for robust RSS-based Wi-Fi positioning. Future work will explore self-supervised adaptation and Transformer-based modeling for automatic adaptation to new environments, potentially without requiring vision-based calibration.

REFERENCES

[1] Fen Liu et al. “Survey on WiFi-based indoor positioning techniques”. In: *IET Communications* 14.9 (June 2020), pp. 1372–1383.

[2] Jinyuan Zhao et al. “Privacy-Preserving Indoor Localization via Active Scene Illumination”. In: *CVPRW 2018*. Salt Lake City, UT, USA, June 2018, pp. 1580–1589.

[3] Hongji Cao et al. “Indoor Positioning Method Using WiFi RTT Based on LOS Identification and Range Calibration”. In: *ISPRS International Journal of Geo-Information* 9.11 (Nov. 2020), p. 627.

[4] C. Chen, G. Zhou, and Y. Lin. “Cross-Domain WiFi Sensing with Channel State Information: A Survey”. In: *ACM Computing Surveys* 55.11 (Nov. 2023), p. 231. DOI: 10.1145/3570325.

[5] Q. He, E. Yang, and S. Fang. “A Survey on Human Profile Information Inference via Wireless Signals”. In: *IEEE Communications Surveys & Tutorials* (Mar. 2024). DOI: 10.1109/COMST.2024.3373397.

[6] R. Du et al. “An Overview on IEEE 802.11bf: WLAN Sensing”. In: *IEEE Communications Surveys & Tutorials* (June 2024). DOI: 10.1109/COMST.2024.3408899.

[7] Jörg Schäfer et al. “Human Activity Recognition Using CSI Information with Nexmon”. In: *Applied Sciences* 11.19 (Sept. 2021), p. 8860.

[8] Daniel Halperin et al. “Tool Release: Gathering 802.11n Traces with Channel State Information”. In: *ACM SIGCOMM Computer Communication Review* 41.1 (Jan. 2011), pp. 53–53.

[9] I. Ahmad, A. Ullah, and W. Choi. “WiFi-Based Human Sensing with Deep Learning: Recent Advances, Challenges, and Opportunities”. In: *IEEE Open Journal of the Communications Society* (Jan. 2024). DOI: 10.1109/OJCOMS.2024.011100.

[10] J. Dai et al. “A Survey of Latest Wi-Fi Assisted Indoor Positioning on Different Principles”. In: *Sensors* 23.18 (Sept. 2023), p. 7961. DOI: 10.3390/s23187961.

[11] X. Feng, K. A. Nguyen, and Z. Luo. “A survey of deep learning approaches for WiFi-based indoor positioning”. In: *Journal of Information and Telecommunication* (June 2021). DOI: 10.1080/24751839.2021.1975425.

[12] Laura Radaelli, Yael Moses, and Christian S. Jensen. “Using Cameras to Improve Wi-Fi Based Indoor Positioning”. In: *W2GIS 2014*. Vol. 8470. Lecture Notes in Computer Science. Seoul, South Korea: Springer, May 2014, pp. 166–183.

[13] Dongdeok Kim and Young-Joo Suh. “Automatic Fingerprint Data Labeling Using WiFi Signal and Smartphone Camera for Indoor Positioning”. In: *Wireless Communications and Mobile Computing* 2024.1 (June 2024), p. 8663246.

[14] Selda Gïney et al. “Wi-Fi based indoor positioning system with using deep neural network”. In: *2020 43rd International Conference on Telecommunications and Signal Processing (TSP)*. IEEE. Milan, Italy, July 2020, pp. 225–228.

[15] Zhen Wu et al. “Attention Mechanism and LSTM Network for Fingerprint-Based Indoor Location System”. In: *Sensors* 24.5 (Feb. 2024), p. 1398.

[16] Zhenjie Ma and Ke Shi. “Few-Shot Learning for WiFi Fingerprinting Indoor Positioning”. In: *Sensors* 23.20 (Oct. 2023), p. 8458.

[17] Shiqi Li, Chi Xu, and Ming Xie. “A Robust O(n) Solution to the Perspective-n-Point Problem”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34.7 (July 2012), pp. 1444–1450.