MITSUBISHI ELECTRIC RESEARCH LABORATORIES
https://www.merl.com

Object Trajectory Estimation with Multi-Band Wi-Fi Neural Dynamic Fusion

Kato, Sorachi; Wang, Pu; Koike-Akino, Toshiaki; Fujihashi, Takuya; Mansour, Hassan; Boufounos, Petros T.

TR2024-019 March 16, 2024

Abstract

In contrast to existing multi-band Wi-Fi fusion in a frame-to-frame basis for simple classification, this paper considers asynchronous sequence-to-sequence fusion between sub-7 GHz channel state information (CSI) and 60 GHz beam SNR for more challenging down- stream tasks such as continuous regression. To handle the timing disparity between the two channel measurements, we extend our recently proposed dual-decoder neural dynamic (DDND) framework with latent ordinary differential equations (ODEs), align the distinct latent dynamic states at the same time instances, and introduce a post-ODE fusion framework. The resulting neural dynamic fusion (NDF) framework is trained in an end-to-end fashion with a modified variational autoencoder loss function. Evaluation over a newly collected in-house multi-band Wi-Fi dataset shows the advantage of the proposed NDF method over frame-based and DDND methods.

IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP) 2024

Mitsubishi Electric Research Laboratories, Inc.
201 Broadway, Cambridge, Massachusetts 02139

OBJECT TRAJECTORY ESTIMATION WITH MULTI-BAND WI-FI NEURAL DYNAMIC FUSION

Sorachi Kato[1,2], *Pu Wang*[1], *Toshiaki Koike-Akino*[1], *Takuya Fujihashi*[2], *Hassan Mansour*[1], *Petros Boufounos*[1]

[1]Mitsubishi Electric Research Laboratories (MERL), Cambridge, MA 02139, USA
[2]Graduate School of Information Science and Technology, Osaka University, Suita, Osaka, Japan

ABSTRACT

In contrast to existing multi-band Wi-Fi fusion in a frame-to-frame basis for simple classification, this paper considers asynchronous sequence-to-sequence fusion between sub-7 GHz channel state information (CSI) and 60 GHz beam SNR for more challenging downstream tasks such as continuous regression. To handle the timing disparity between the two channel measurements, we extend our recently proposed dual-decoder neural dynamic (DDND) framework with latent ordinary differential equations (ODEs), align the distinct latent dynamic states at the same time instances, and introduce a post-ODE fusion framework. The resulting neural dynamic fusion (NDF) framework is trained in an end-to-end fashion with a modified variational autoencoder loss function. Evaluation over a newly collected in-house multi-band Wi-Fi dataset shows the advantage of the proposed NDF method over frame-based and DDND methods.

***Index Terms*—** WLAN sensing, 802.11bf, Wi-Fi sensing, ISAC, localization, multi-band fusion, and dynamic learning.

1. INTRODUCTION

Wi-Fi sensing, e.g., device localization and device-free human sensing, has received much attention in the past decade from both academia and industry. This trend has been manifested by the establishment of a new task group (TG) - 802.11bf WLAN Sensing – in September 2020, to go beyond data transmission and meet industry demands for robust and reliable wireless sensing.

Existing studies are primarily based on the coarse-grained receiver signal strength indicator (RSSI) and the fine-grained channel state information (CSI) in terms of channel frequency response over OFDM subcarriers [1–5]. At a high frame rate, CSI reflects intrinsic channel features from frequency subcarriers (delay) and multiple transmitter-receiver pairs (angle) but may experience channel instability to even small-scale environment changes. These features can be extracted from frame-based or sequence-based frameworks [6–9]. On the other hand, mid-grained mmWave beam training measurements at 60 GHz, e.g., beam SNR, have shown better channel stability over time [10–18]. These beam SNR measurements originate from sector-level directional beam training, a mandatory step for mmWave Wi-Fi to compensate for large path loss and establish the link between the AP and the user. However, they suffer from low frame rate and irregular sample intervals due to the beam training overhead and follow-up association steps. To deal with such intermittent sampling issues over multiple frames, [19] proposed a dual-decoder neural dynamic learning (DDND) framework that learns the underlying latent dynamics in a continuous-time fashion by exploring the neural ODE framework [20–23].

The work of S. Kato was done during his visit and internship at MERL.

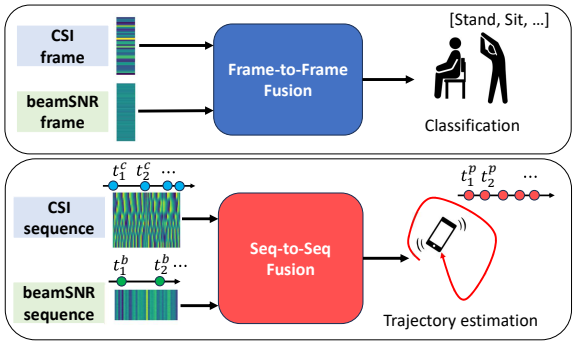


Fig. 1: Multi-band Wi-Fi fusion from *frame-to-frame* basis of [16] for classification (top) to asynchronous *sequence-to-sequence* basis for continuous-time regression (bottom).

Fusion-based approaches have been considered in the literature for robustness and better accuracy. Heterogeneous sensor fusion was studied between Wi-Fi and other modalities, e.g., Bluetooth and acoustics [24–26]. Within Wi-Fi channel measurements, CSI and RSSI can be simply concatenated for a joint feature extraction [27]. [28] proposed to fuse the phase and amplitude of the fine-grained CSI for localization. When multiple access points (APs) are deployed in the scene, multi-AP fusion was proposed in [29, 30] by exploring the generalized interview and intraview discriminant correlation analysis and, respectively, the maximum mean discrepancy (MMD) criterion. To the best of our knowledge, our previous work in [16] is the only effort considering multi-band Wi-Fi fusion between CSI and beam SNR. However, it is limited to simple classification tasks, e.g., pose classification (over 8 stationary poses), seat occupancy sensing (8 static patterns), and fixed-grid localization. Despite being sampled at different time instances, the two channel measurements can be simply combined on a *frame-to-frame* basis as these asynchronous samples correspond to the same stationary label (e.g., pose, occupancy, location) and their respective sampling time becomes irrelevant for the fusion; see the top plot of Fig. 1.

In this paper, we substantially advance the multi-band Wi-Fi fusion framework, evolving from the static frame-to-frame basis of [16] to a dynamic asynchronous *sequence-to-sequence* basis (see the bottom plot of Fig. 1.), thus supporting more challenging downstream tasks, e.g., continuous regression and continuous-time object trajectory estimation, as opposed to simple classification problems. Referring to the neural dynamic fusion (NDF) framework, it is achieved by extending the beam SNR-only DDND framework of [19] to two separate neural ODE decoders defined on a shared time axis, forcing the two decoders to generate virtual latent dynamic states in the same time instances and combining these synchronized

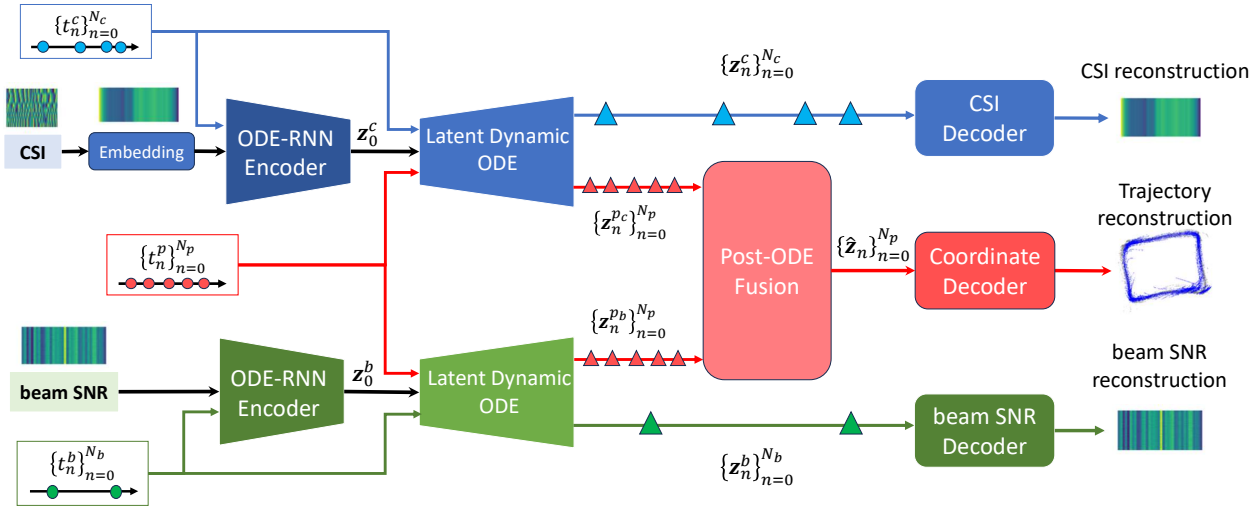


Fig. 2: Asynchronous multi-band Wi-Fi fusion with neural dynamic learning for object trajectory estimation.

post-ODE latent dynamic states via a fusion block for continuous trajectory estimation. To further align the two distinct latent spaces over time, we encode both the CSI and the beam SNR sequences to estimate their respective initial conditions at a common starting time t_0, which can precede the first sample from either channel measurement. To train the proposed multi-decoder neural dynamic fusion network, we consider a loss function that is a weighted sum of waveform reconstruction losses at asynchronous time instances and coordinate estimation errors at these synchronized time instances. With a newly collected in-house multi-band Wi-Fi dataset for robot trajectory estimation, comprehensive performance evaluation confirms the effectiveness of the proposed neural dynamic fusion over a list of baseline methods.

2. PROBLEM FORMULATION

We formulate trajectory estimation as a regression with asynchronous CSI and beam SNR sequences. At time t_n^b, we collect a set of M beam SNR values $\mathbf{b}_n = [b_1, b_2, \cdots, b_M]^\top \in \mathbb{R}^{M_b \times 1}$, each corresponding to one beam training pattern. For CSI, at time t_n^c, we collect a channel frequency response matrix $\mathbf{C}_n \in \mathbb{C}^{N_{\rm Tx} N_{\rm Rx} \times N_{\rm s}}$ over $N_{\rm s}$ OFDM subcarriers, $N_{\rm Tx}$ transmitting antennas, and $N_{\rm Rx}$ receiving antennas. For a time window of length ΔT_w, we collect N_b beam SNR samples and N_c CSI samples with which we aim to estimate the trajectory at N_p time instances $t_n^p, n = 1, \cdots, N_p$. Note that N_b and N_c may vary from one time window to another.

The problem of interest is to estimate the coordinate of a trajectory at any time point t_n^p within the time window with the beam SNR and CSI input sequences along with their respective time stamps,

$$\{\mathbf{b}_n, t_n^b\}_{n=0}^{N_b}, \ \{\mathbf{C}_n, t_n^c\}_{n=0}^{N_c} \rightarrow \{\mathbf{p}_n\}_{n=0}^{N_p}, \tag{1}$$

where $\mathbf{p}_n = [x_n, y_n]^\top$ consists of two-dimensional coordinates at t_n^p.

3. MULTI-BAND NEURAL DYNAMIC FUSION

The proposed multi-band sequence-to-sequence Wi-Fi fusion framework is shown in Fig. 2. From left to right, we have two ODE-RNN encoders for both beam SNR and CSI, two separate latent dynamic ODE blocks, a post-ODE fusion block, and three decoders for waveform reconstruction and coordinate estimation. In the following, we introduce each block in more details[1].

3.1. Embedding Layers

For the fine-grained CSI matrix $\mathbf{C}_n$, we follow standard calibration steps to remove sampling time offset (STO) between transmitter and receiver due to sampling frequency offsets and packet detection errors and carrier frequency offset (CFO) among receiver RF chains [31, 32]. The calibrated complex-valued CSI matrix at t_n^c is then flattened and mapped to an embedding space via a pre-trained one-dimensional convolution (Conv1D) network. Specifically, we have $\mathcal{E}(\mathbf{C}_n) = \mathbf{c}_n \in \mathbb{R}^{M_c \times 1}$ as the embedding vector of the CSI. For the mid-grained beam SNR $\mathbf{b}_n$, we directly take the raw beam SNR as the input to the following ODE-RNN encoders. To maintain the balance between the two input dimensions, we have approximately $M_b \approx M_c$.

3.2. ODE-RNN Encoders

For the two separate encoders for beam SNR and CSI, respectively, we follow the ODE-RNN encoder architecture [19, 21] to take the reversed input sequences $t_N, t_{N-1}, \cdots, t_1$ to estimate the initial latent conditions at the common starting time t_0[2].

Let us start from beam SNR encoding. With an ODE-RNN encoder, each recurrent unit updates its hidden vector $\mathbf{h}_n \in \mathbb{R}^{L_h \times 1}$ with an auxiliary vector $\mathbf{h}'_{n+1}$ and $\mathbf{b}_n$.

$$\mathbf{h}_n = \mathcal{G}_{\theta_g}(\mathbf{h}'_n, \mathbf{b}_n), \tag{2}$$

where $\mathcal{G}_{\theta_g}$ can be either GRU or LSTM unit with learnable parameters θ_g. Then the ODE-RNN encoder utilizes an ODE function $\mathcal{O}_{\theta_e^b}$ to describe the propagation of the latent vector in a continuous-time fashion,

$$\frac{d\mathbf{h}(t)}{dt} = \mathcal{O}_{\theta_e^b}(\mathbf{h}(t), t), \tag{3}$$

[1]Note that the mathematical ornamental characters represent neural networks, and the θ subscript at the bottom right denote learnable parameters.

[2]Since the ODE-RNN encoder applies to both beam SNR and CSI, we ignore the superindex, e.g., $\mathbf{h}_n^{b/c}$ or $t_n^{b/c}$ for simplicity.

where the ODE function is parameterized by a multi-layer perceptron (MLP) network with learnable parameters θ_e^b. Utilizing a numerical ODE solver $\mathcal{S}$, e.g., Euler and Runge-Kutta solvers, one can then propagate the hidden vector $\mathbf{h}_{n+1}$ at time t_{n+1} to the auxiliary vector $\mathbf{h}'_n$ at the current time t_n (recall that the time is in a reversed order for estimating the initial condition):

$$\begin{aligned}\mathbf{h}'_n &= \mathcal{S}(\mathcal{O}_{\theta_e^b}, \mathbf{h}_{n+1}, (t_n, t_{n+1})) \\ &= \mathbf{h}_{n+1} + \int_{\tau=t_{n+1}}^{t_n} \mathcal{O}_{\theta_e^b}(\mathbf{h}(\tau), \tau) d\tau. \end{aligned} \tag{4}$$

By iterating between (2) and (4), we can propagate the latent encoding vector from t_N to t_0, and the same procedure can be adopted for CSI encoding with another set of parameters θ_e^c.

3.3. Latent Dynamic Learning

Once the hidden state $\mathbf{h}_0$ at t_0 is obtained, $\mathbf{h}_0$ is used to generate $\mathbf{z}_0$, the initial condition in the latent space for latent dynamic learning. Following the VAE framework [33], the posterior distribution of $\mathbf{z}_0$ is approximated as

$$q_{\theta_e}(\mathbf{z}_0|\mathbf{h}_0) = \mathcal{N}(\boldsymbol{\mu}_{\mathbf{z}_0}, \boldsymbol{\sigma}_{\mathbf{z}_0}) \tag{5}$$

where the mean and standard deviation are mapped from $\mathbf{h}_0$

$$\boldsymbol{\mu}_{\mathbf{z}_0}, \boldsymbol{\sigma}_{\mathbf{z}_0} = \mathcal{M}_{\theta_s}(\mathbf{h}_0) \tag{6}$$

with $\mathcal{M}$ denoting an MLP. We sample $\mathbf{z}_0^b \in \mathbb{R}^{L_z^b}$ and $\mathbf{z}_0^c \in \mathbb{R}^{L_z^c}$ from their respective posterior mean and standard deviations as:

$$\begin{aligned}\mathbf{z}_0^b &= \boldsymbol{\mu}_{\mathbf{z}_0}^b + \boldsymbol{\sigma}_{\mathbf{z}_0}^b \odot \boldsymbol{\epsilon}_1,\ \boldsymbol{\epsilon}_1 \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{L_z^b}), \\ \mathbf{z}_0^c &= \boldsymbol{\mu}_{\mathbf{z}_0}^c + \boldsymbol{\sigma}_{\mathbf{z}_0}^c \odot \boldsymbol{\epsilon}_2,\ \boldsymbol{\epsilon}_2 \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{L_z^c}), \end{aligned} \tag{7}$$

where $\odot$ represents Hadamard product.

For the latent dynamic learning, we further enforce that the above initial latent conditions $\mathbf{z}_0^b$ and $\mathbf{z}_0^c$ are aligned at the common starting time $t_0^b = t_0^c = t_0$. In this way, the latent dynamic learning block takes the sampled initial latent condition at t_0 and propagates to the latent dynamic state at any query time t_n achieved by using another continuous-time ODE function $\mathcal{O}_d$ modeled by a neural network with parameters $\boldsymbol{\theta}_d^{b/c}$.

We first query the beam SNR-related and CSI-related dynamic learning blocks with their respective sampling time $t_n^{b/c}$

$$\begin{aligned}\mathbf{z}_n^b &\triangleq \mathbf{z}_{t_n^b}^b = \mathbf{z}_0^b + \int_{t_0}^{t_n^b} \mathcal{O}_{\theta_d^b}(\mathbf{z}_t, t) dt = \mathcal{S}(\mathcal{O}_{\theta_d^b}, \mathbf{z}_0^b, (t_0, t_n^b)), \\ \mathbf{z}_n^c &\triangleq \mathbf{z}_{t_n^c}^c = \mathbf{z}_0^c + \int_{t_0}^{t_n^c} \mathcal{O}_{\theta_d^c}(\mathbf{z}_t, t) dt = \mathcal{S}(\mathcal{O}_{\theta_d^c}, \mathbf{z}_0^c, (t_0, t_n^c)). \end{aligned} \tag{8}$$

These latent states for beam SNR and CSI are then fed to the waveform decoders (see Section 3.5) for waveform/feature reconstruction.

Due to the continuous-time dynamic modeling capability, we then query the beam SNR-related and CSI-related dynamic learning blocks at a set of unseen but shared time instances t_n^p for supervised training and downstream tasks. Specifically, we have

$$\begin{aligned}\mathbf{z}_n^{p_b} &\triangleq \mathbf{z}_{t_n^p}^b = \mathbf{z}_0^b + \int_{t_0}^{t_n^p} \mathcal{O}_{\theta_d^b}(\mathbf{z}_t, t) dt = \mathcal{S}(\mathcal{O}_{\theta_d^b}, \mathbf{z}_0^b, (t_0, t_n^p)), \\ \mathbf{z}_n^{p_c} &\triangleq \mathbf{z}_{t_n^p}^c = \mathbf{z}_0^c + \int_{t_0}^{t_n^p} \mathcal{O}_{\theta_d^c}(\mathbf{z}_t, t) dt = \mathcal{S}(\mathcal{O}_{\theta_d^c}, \mathbf{z}_0^c, (t_0, t_n^p)), \end{aligned} \tag{9}$$

where the above ODE associated parameters $\boldsymbol{\theta}_d^b$ and $\boldsymbol{\theta}_d^c$ are the same as the ones in (8).

3.4. Post-ODE Latent Fusion

It is seen from (9) that the latent dynamic states $\mathbf{z}_n^{p_b}$ for the beam SNR are synchronized with their corresponding CSI latent dynamic state $\mathbf{z}_n^{p_c}$ at t_n^p. Such a latent dynamic alignment motivates us to combine the two post-ODE dynamic pathways into a fused dynamic pathway that might be more essential for downstream tasks such as the object trajectory estimation.

To this end, we first project the aligned post-ODE states into higher dimensions

$$\widehat{\mathbf{z}}_n^{b/c} = \mathcal{M}_{\boldsymbol{\theta}_p^{b/c}}(\mathbf{z}_n^{p_{b/c}}) \tag{10}$$

using an MLP with learnable weights $\boldsymbol{\theta}_p^{b/c}$, then concatenate these projected latent states, and finally compress it into a fused latent state $\widehat{\mathbf{z}}_n$ at t_n^p

$$\widehat{\mathbf{z}}_n = \mathcal{M}_{\boldsymbol{\theta}_f}([\widehat{\mathbf{z}}_n^b, \widehat{\mathbf{z}}_n^c]^\top), \quad n = 1, \cdots, N_p \tag{11}$$

where $\boldsymbol{\theta}_f$ consists of the weight matrices and bias terms for the fusion MLP. It is expected that the post-ODE latent fusion allows for a combination of the dynamics of distinct Wi-Fi propagation characteristics at different frequency bands in a complementary way and enhances the ability to represent a unified latent space for downstream tasks.

3.5. Multi-Head Decoders

From the latent dynamic learning and fusion blocks, we have the latent dynamic states in three distinct sets of time instances:

- $\mathbf{z}_n^b, n = 1, \cdots, N_b$, at the beam SNR sampling time t_n^b;
- $\mathbf{z}_n^c, n = 1, \cdots, N_c$, at the CSI sampling time t_n^c;
- $\widehat{\mathbf{z}}_n, n = 1, \cdots, N_p$, at shared time instances t_n^p.

For the first two sets of latent dynamic states, we use two separate MLP heads to decode them back to the beam SNR waveform or CSI embedding feature spaces as

$$\widehat{\mathbf{b}}_n = \mathcal{M}_{\boldsymbol{\theta}_b}(\mathbf{z}_n^b), \quad \widehat{\mathbf{c}}_n = \mathcal{M}_{\boldsymbol{\theta}_c}(\mathbf{z}_n^c). \tag{12}$$

For the fused latent states, we use another MLP head to project them into a coordinate trajectory as

$$\widehat{\mathbf{p}}_n = \mathcal{M}_{\boldsymbol{\theta}_p}(\widehat{\mathbf{z}}_n), \quad n = 1, \cdots, N_p. \tag{13}$$

All MLP heads are shared over time steps.

3.6. Loss Function

In the following, we adopt a loss function modified from the evidence lower bound (ELBO) of the standard VAE [33] for the proposed NDF framework without providing the detailed derivation

$$\begin{aligned} L &= \sum_{n=0}^{N_p} \|\hat{\mathbf{p}}_n - \mathbf{p}_n\|_2 \\ &+ \lambda_1 \sum_{n=0}^{N_c} \|\hat{\mathbf{c}}_n - \mathbf{c}_n\|_2 - \lambda_2 \sum_{l}^{L_z^c} \left(1 + \log(\sigma_l^c)^2 - (\mu_l^c)^2 - (\sigma_l^c)^2\right) \\ &+ \lambda_3 \sum_{n=0}^{N_b} \|\hat{\mathbf{b}}_n - \mathbf{b}_n\|_2 - \lambda_4 \sum_{l}^{L_z^b} \left(1 + \log(\sigma_l^b)^2 - (\mu_l^b)^2 - (\sigma_l^b)^2\right). \end{aligned} \tag{14}$$

where the hyperparameters $\lambda_{1/2/3/4}$ play the trade-off roles between the supervised trajectory estimation errors to the waveform (beam SNR) and embedding feature (CSI) reconstruction errors.

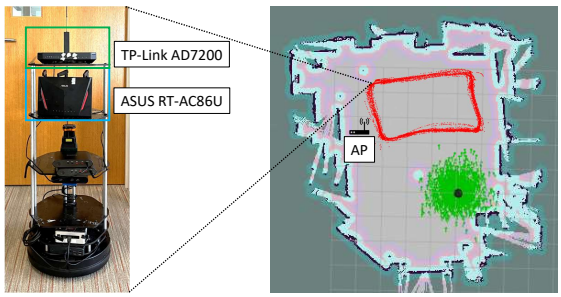


Fig. 3: A TurtleBot testbed and trajectory configuration for data collection. The map on the right was created by a LiDAR on TurtleBot.

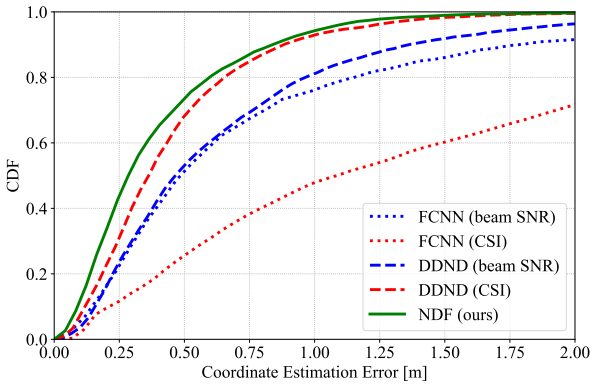


Fig. 4: CDF of localization errors.

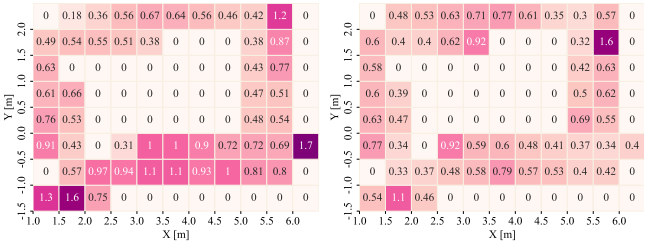
4. PERFORMANCE EVALUATION

4.1. In-House Testbed and Data Collection

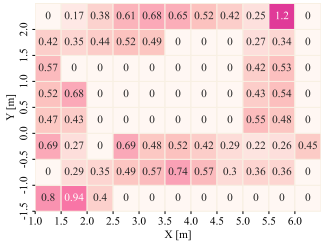
We built a testbed to automatically collect CSIs and beam SNRs from a moving device, shown in Fig. 3. We used ASUS RT-AC86U for IEEE 802.11ac CSI collection, and TP-Link Talon AD7200 for IEEE 802.11ad beam SNR collection. Two routers were installed on a TurtleBot and continuously collected CSI and beam SNR while driving the TurtleBot along a predefined route shown as red scatter points in Fig. 3. The TurtleBot was equipped with a LiDAR to grasp its position while moving, and recorded the coordinate information used as labels for training. We had $M_b = 36$ for the beam SNR and $(N_{\mathrm{Rx}}, N_{\mathrm{Tx}}, N_{\mathrm{s}}) = (4, 2, 234)$ for the CSI.

4.2. Implementation

We set $\Delta T_w = 5$ seconds to group the corresponding CSI, beam SNR, and coordinate labels. We divided all 5 s sequences into training and test sets with a ratio of 8:2. The beam SNRs $\{\mathbf{b}_n\}_{n=0}^{N_b}$ were normalized to $[0, 1]$, and so were the timestamps $\{\mathbf{t}_n^b\}_{n=0}^{N_b}$, $\{\mathbf{t}_n^c\}_{n=0}^{N_c}$, and $\{\mathbf{t}_n^p\}_{n=0}^{N_p}$ to be compatible with the ODE layers. M_c for the embedded CSI feature vectors $\{\mathbf{c}_n\}_{n=0}^{N_c}$ was set to 36, and the features were also normalized to $[0, 1]$. The hidden status dimension of GRU L_h was set to 20, and the latent dimension $L_z^c = L_z^b = 20$. For the loss function, we empirically set $\lambda_1 = \lambda_3 = 0.1$ for the waveform reconstruction and $\lambda_2 = \lambda_4 = 0.01$ for the KL divergence loss. Neural networks were implemented using PyTorch 2.0.1 on Python 3.11 and trained on a GPU with CUDA 12.1 enabled.



(a) DDND (beam SNR): 0.646 / 0.465 (b) DDND (CSI): 0.450 / 0.363



(c) NDF (ours): 0.389 / 0.285

Fig. 5: Visualization of average localization errors over 50×50 cm^2 grids with mean and median errors listed in the caption.

4.3. Comparison to Baseline Methods

We implemented the baseline methods: a fully connected neural network (FCNN) and DDND. FCNN is the frame-based method and DDND is the sequence-based method, and both methods estimate the coordinates only from each CSI/beam SNR frames. In FCNN and DDND, CSI is fed into the pre-trained embedding layers, explained in Sec. 3.1, before the coordinate estimation. The criterion for localization performance is the mean Euclidean distance errors between the ground truth and estimated locations.

Fig. 4 shows the performance of the proposed and baseline methods in terms of the cumulative distribution function (CDF) of the mean Euclidean distance between the ground truth and estimated coordinates. It shows that NDF overwhelms not only frame-based baselines but also single-band DDNDs. Learning and modeling the dynamics with CSI or beam SNR alone is still a good deal, but combining different radio observations from multiple frequency bands in the unified latent space further promotes trajectory learning and helps the model obtain better understanding as to the relationship between radio fluctuation and location along with the continuous time sequence.

Figs. 5 (a) to (c) visualize the average Euclidean distance error for each 50 cm square grid of the target space. It shows that the proposed NDF avoids significant errors at certain coordinates, as shown in Fig. 5 (c), and reduces the average error compared to single-band DDNDs. This indicates NDF provides a much clearer trajectory with less variation than other baselines.

5. CONCLUSION

In this paper, we proposed an asynchronous multi-band Wi-Fi fusion framework using latent ODE learning. Specifically, the framework projects CSI and beam SNR onto their own latent space, utilizes a post-ODE neural dynamics fusion to align these measurements in the latent space, and estimates object trajectory from the aligned and fused latent variables. Real-world experiments validate the proposed neural dynamic fusion framework.

6. REFERENCES

[1] Moustafa Youssef et al., "The horus location determination system," *Wirel. Netw.*, vol. 14, no. 3, pp. 357–374, 2008.

[2] Minh Tu Hoang et al., "Recurrent neural networks for accurate RSSI indoor localization," *IEEE Internet Things J.*, vol. 6, no. 6, pp. 10639–10651, 2019.

[3] Hao Chen et al., "ConFi: Convolutional neural networks based indoor Wi-Fi localization using channel state information," *IEEE Access*, vol. 5, pp. 18066–18074, 2017.

[4] Jianyang Ding et al., "WiFi CSI-based human activity recognition using deep recurrent neural network," *IEEE Access*, vol. 7, pp. 174257–174269, 2019.

[5] Lingyan Zhang et al., "Device-Free tracking via joint velocity and AOA estimation with commodity WiFi," *IEEE Sens. J.*, vol. 19, no. 22, pp. 10662–10673, 2019.

[6] Xuyu Wang et al., "CSI-based fingerprinting for indoor localization: A deep learning approach," *IEEE Trans. Veh. Technol.*, vol. 66, no. 1, pp. 763–776, 2017.

[7] Minh Tu Hoang et al., "A CNN-LSTM quantifier for single access point CSI indoor localization," *arXiv preprint arXiv:2005.06394*, pp. 1–10, 2020.

[8] Haotai Sun et al., "WiFi based fingerprinting positioning based on seq2seq model," *Sensors*, vol. 20, no. 13, pp. 3767, 2020.

[9] Jianyuan Yu et al., "Centimeter-level indoor localization using channel state information with recurrent neural networks," in *PLANS*, 2020, pp. 1317–1323.

[10] Milutin Pajovic et al., "Fingerprinting-based indoor localization with commercial MMWave WiFi–Part I: RSS and Beam Indices," in *GLOBECOM*, 2019, pp. 1–6.

[11] Pu Wang et al., "Fingerprinting-based indoor localization with commercial mmwave WiFi–Part II: Spatial beam SNRs," in *GLOBECOM*, 2019, pp. 1–6.

[12] Toshiaki Koike-Akino et al., "Fingerprinting-based indoor localization with commercial mmwave WiFi: A deep learning approach," *IEEE Access*, vol. 8, pp. 84879–84892, 2020.

[13] Dolores Garcia et al., "POLAR: Passive object localization with IEEE 802.11ad using phased antenna arrays," in *INFOCOM*, 2020, pp. 1838–1847.

[14] Pu Wang et al., "Fingerprinting-based indoor localization with commercial mmwave WiFi: NLOS propagation," in *GLOBECOM*, 2020, pp. 1–6.

[15] Jianyuan Yu et al., "Human pose and seat occupancy classification with commercial mmwave WiFi," in *GLOBECOM Workshop on Integrated Sensing and Communication (ISAC)*, 2020, pp. 1–6.

[16] Jianyuan Yu et al., "Multi-band Wi-Fi sensing with matched feature granularity," *IEEE Internet Things J.*, vol. 9, no. 23, pp. 23810–23825, 2022.

[17] Alejandro Blanco et al., "Augmenting mmwave localization accuracy through sub-6 GHz on off-the-shelf devices," in *MobiSys*, 2022, pp. 477–490.

[18] Toshiaki Koike-Akino et al., "Quantum transfer learning for Wi-Fi sensing," in *ICC*, 2022, pp. 654–659.

[19] Cristian J Vaca-Rubio et al., "mmwave Wi-Fi trajectory estimation with continuous-time neural dynamic learning," in *ICASSP*, 2023, pp. 1–5.

[20] Ricky T. Q. Chen et al., "Neural ordinary differential equations," in *Adv. Neural Inf. Process.*, 2018, vol. 31, pp. 1–13.

[21] Yulia Rubanova et al., "Latent ordinary differential equations for irregularly-sampled time series," in *Adv. Neural Inf. Process.*, 2019, vol. 32, pp. 1–11.

[22] Aiqing Zhu et al., "On numerical integration in neural ordinary differential equations," in *ICML*, 2022, pp. 27527–27547.

[23] Ho Huu Nghia Nguyen et al., "Improving neural ordinary differential equations with Nesterov's accelerated gradient method," *Adv. Neural Inf. Process.*, vol. 35, pp. 7712–7726, 2022.

[24] Lyu-Han Chen et al., "Intelligent fusion of Wi-Fi and inertial sensor-based positioning systems for indoor pedestrian navigation," *IEEE Sens. J.*, vol. 14, no. 11, pp. 4034–4042, 2014.

[25] Hongfei Xue et al., "DeepFusion: A deep learning framework for the fusion of heterogeneous sensory data," in *MobiHoc*, 2019, pp. 151–160.

[26] Xiansheng Guo et al., "A survey on fusion-based indoor positioning," *IEEE Commun. Surv. Tutor.*, vol. 22, no. 1, pp. 566–594, 2020.

[27] Afaz Uddin Ahmed et al., "Multi-radio data fusion for indoor localization using Bluetooth and WiFi," in *PECCS*, 2019, pp. 13–24.

[28] Xiaochao Dang et al., "A novel passive indoor localization method by fusion CSI amplitude and phase information," *Sensors*, vol. 19, no. 4, 2019.

[29] Tahsina Farah Sanam et al., "A multi-view discriminant learning approach for indoor localization using amplitude and phase features of CSI," *IEEE Access*, vol. 8, pp. 59947–59959, 2020.

[30] Zhihui Gao et al., "Crisloc: Reconstructable csi fingerprinting for indoor smartphone localization," *IEEE Internet Things J.*, vol. 8, no. 5, pp. 3422–3437, 2020.

[31] Manikanta Kotaru et al., "SpotFi: Decimeter level localization using WiFi," in *SIGCOMM*, 2015, pp. 269–282.

[32] Dongheng Zhang et al., "Calibrating phase offsets for commodity WiFi," *IEEE Syst. J.*, vol. 14, no. 1, pp. 661–664, 2020.

[33] Diederik P Kingma et al., "Auto-Encoding variational bayes," *arXiv preprint arXiv:1312.6114*, pp. 1–14, 2013.