2026 16th International Conference on Communications (COMM) | 979-8-3195-0655-9/26/$31.00 ©2026 IEEE | DOI: 10.1109/COMM70054.2026.11591365

Temporal Windowing for Cross-Session Wi-Fi CSI Presence Detection

Andrei Țaga*, Răzvan Crăciunescu†, Răzvan Mihai‡, Mihnea Udrea§, Constantin Pintilie¶
Faculty of Electronics, Telecommunications and Information Technology
National University of Science and Technology POLITEHNICA Bucharest
Bucharest, Romania
*andrei_madalin.taga@upb.ro †razvan.craciunescu@upb.ro ‡marius_razvan.mihai@upb.ro
§mihnea.udrea@upb.ro ¶constantin.pintilie@gmail.com

***Abstract*—Wi-Fi Channel State Information (CSI) enables device-free presence sensing, but temporal windowing can strongly affect cross-session detection performance. In this work, we analyze binary presence detection under a strict leave-one-session-out (LOSO) protocol using four continuous-stream CSI acquisitions collected under baseline, small-shift, non-line-of-sight (NLoS), and cross-room conditions. From the same recorded streams, we generate offline re-windowed datasets with temporal window lengths $W \in \{4, 8, 16\}$ and hop sizes $H \in \{1, 2, 4\}$. Each sample is represented by CSI amplitude and its temporal difference, and all configurations are evaluated using the same lightweight convolutional neural network (CNN). We report accuracy, F1-score, and Matthews correlation coefficient (MCC). Results show that increasing the window length consistently improves cross-session robustness, while hop size has a comparatively smaller effect. The best overall performance is obtained for $W = 16$, indicating that window length is the dominant temporal design factor in this CSI-based presence detection pipeline.**

***Index Terms*—Wi-Fi sensing, channel state information, device-free sensing, presence detection, temporal windowing, cross-session evaluation, LOSO**

I. Introduction

Wi-Fi sensing enables *device-free* monitoring by exploiting how human presence perturbs the wireless propagation channel, without requiring the person to carry any device. Compared to camera or wearable-based solutions, Wi-Fi sensing can be more privacy-friendly and can reuse existing communication infrastructure. Channel State Information (CSI) provides fine-grained, per-subcarrier channel measurements and has therefore become a widely used signal source for indoor sensing tasks such as presence detection and activity recognition [1]–[4].

A major challenge is that CSI signatures are highly environment-dependent [1], [5]. Changes in geometry, multipath structure, attenuation, or non-line-of-sight (NLoS) conditions can shift the channel response and degrade generalization across sessions. Therefore, strict cross-session evaluation is more informative than random train-test splits within a single recording session [5], [6].

In this paper, we study binary *presence* detection from Wi-Fi CSI under a strict leave-one-session-out (LOSO) protocol using four continuous-stream acquisitions collected under baseline, small-shift, NLoS, and cross-room conditions. Starting from the same recorded streams, we generate datasets using temporal window lengths $W \in \{4, 8, 16\}$ and hop sizes $H \in \{1, 2, 4\}$, and evaluate all configurations using the same lightweight convolutional neural network (CNN) with a two-channel input formed by CSI amplitude and its temporal difference (RAW+Δ). Rather than proposing a new model, the objective is to isolate the effect of temporal window design itself under session-independent evaluation.

Contributions

- A controlled LOSO evaluation on four CSI recording conditions with increasing deployment shift.
- A temporal windowing study based on offline re-windowing of the same continuous streams for $W \in \{4, 8, 16\}$ and $H \in \{1, 2, 4\}$.
- An analysis that isolates temporal window design from model design by keeping the input representation and CNN architecture fixed across all experiments.

The remainder of this paper is organized as follows. Section II reviews the most relevant literature on CSI-based sensing, cross-domain generalization, and deep wireless sensing models. Section III presents the dataset, preprocessing pipeline, model, and evaluation protocol. Section IV reports the experimental results, while Section V discusses the main findings, limitations, and future directions. Section VI concludes the paper.

II. Literature Review

Wi-Fi CSI has become a widely used signal source for device-free sensing because fine-grained subcarrier measurements can capture human-induced perturbations of the wireless channel [1]. Early CSI-based systems demonstrated that such measurements are informative for indoor activity inference [2], [3], while human presence detection was addressed more directly in [4]. These studies established the feasibility of CSI-based sensing, but were typically developed and evaluated in relatively controlled environments.

979-8-3195-0655-9/26/$31.00 ©2026 IEEE

Authorized licensed use limited to: Polytechnic University of Bucharest. Downloaded on October 08,2026 at 12:44:23 UTC from IEEE Xplore. Restrictions apply.

A persistent difficulty is generalization across changed deployment conditions. Variations in geometry, multipath structure, attenuation, and NLoS propagation can significantly alter CSI statistics, which often leads to performance degradation when models are transferred across sessions, rooms, or sites [5], [6].

At the same time, recent work in deep wireless sensing has shown that convolutional and spectrogram-based models can learn useful sensing features directly from structured CSI inputs [7]. Augmentation-based approaches have also been explored to improve the robustness and understanding of deep Wi-Fi sensing systems [8]. In most of this literature, however, the emphasis is placed on network design, learned representations, or robustness enhancement mechanisms.

In contrast, the goal of this paper is not to propose a new model architecture. Instead, we isolate a simpler but important design factor: temporal windowing. Using the same continuous CSI streams, the same RAW+Δ representation, and the same lightweight CNN across all experiments, we analyze how window length and hop size affect performance under strict LOSO evaluation. This makes it possible to study temporal segmentation directly, without conflating it with changes in model architecture or acquisition conditions.

III. Methodology

A. Dataset and Data Collection

Fig. 1 summarizes the end-to-end pipeline used in this work, from continuous CSI stream acquisition to offline sample generation and LOSO evaluation. The dataset used here is built from continuous Wi-Fi CSI recordings and is specifically designed to analyze the effect of temporal windowing on cross-session presence detection.

We collected Wi-Fi CSI data for binary presence detection under four recording conditions chosen to induce realistic deployment shifts: *baseline*, *small-shift*, *NLoS*, and *room2*. Each session contains two recordings, corresponding to the *no-presence* and *presence* classes. In the *presence* case, the same subject moved continuously along an approximately square path in the area between the transmitter and receiver, while in the *no-presence* case the monitored area remained empty. Each recording lasted approximately 8 minutes and yielded about 1200 retained CSI packets. The four recording conditions are summarized in Table I. CSI was collected from a commodity access point (AP) (ZTE H3601) transmitting standard IEEE 802.11 beacon traffic, using a single-antenna software-defined radio (SDR) receiver (Microphase AntSDR E200) controlled through the PlutoSDR Industrial I/O (IIO) interface [9], [10] and a VERT2450 antenna. Recordings used the 5 GHz band (channel 36) with 20 MHz bandwidth ($F_s = 20$ MHz) and receiver gain 30 dB. In the baseline layout, the transmitter (Tx) and receiver (Rx) were placed at approximately desk height with a separation of about 2 m. The remaining sessions introduced controlled changes in propagation conditions, namely a moderate receiver displacement and reorientation (*small-shift*), a non-line-of-sight obstruction created by a cabinet placed between Tx and Rx (*NLoS*), and deployment in a different room (*room2*).

TABLE I
Recording sessions and induced deployment shifts.

Session	Setup / induced shift
S1 baseline	Reference configuration with Tx and Rx placed at approximately desk height and separated by about 2 m.
S2 small-shift	Same room as the baseline, with a moderate displacement and reorientation of the receiver relative to the reference setup.
S3 NLoS	Same room, but with a non-line-of-sight obstruction introduced between Tx and Rx.
S4 room2	Tx/Rx deployment moved to a different room, producing the strongest environment shift among the recorded sessions.

The SDR capture chain stored the real and imaginary parts of the packet-level CSI together with packet timestamps. CSI was extracted from successfully processed IEEE 802.11 beacon packets using a MATLAB-based SDR processing workflow, and the resulting stream was saved as continuous packet-level CSI for the 52 active orthogonal frequency-division multiplexing (OFDM) subcarriers. During acquisition, the receiver operated in repeated capture bursts of approximately 0.8 s with a target of 10 beacon packets per burst. Although the nominal router beacon interval was set to 100 time units (TU), the effective temporal spacing used in the analysis was taken from the timestamps of the retained packets in each recorded stream.

Across the recorded streams, the average retained-packet spacing ranged from approximately 0.30 s to 0.44 s. Accordingly, the tested window lengths $W = \{4, 8, 16\}$ correspond to approximate temporal spans of about 1.2–1.8 s, 2.4–3.5 s, and 4.8–7.0 s, while the hop sizes $H = \{1, 2, 4\}$ correspond to update intervals of about 0.30–0.44 s, 0.60–0.88 s, and 1.2–1.8 s, respectively.

Instead of constructing fixed samples during acquisition, each recording was saved as a continuous CSI stream and the final datasets were generated offline by re-windowing the same streams using temporal window lengths $W \in \{4, 8, 16\}$ and hop sizes $H \in \{1, 2, 4\}$. For each session, class, and (W, H) pair, 200 valid windows were retained, resulting in 400 samples per session and 1600 samples for each temporal configuration. In this work, a valid window denotes a window composed of W consecutive retained CSI packets from the same stream, each containing a complete 52-subcarrier CSI estimate. This design isolates the effect of temporal windowing from the acquisition process and enables a fair comparison across configurations, since all derived datasets originate from the same underlying continuous recordings.

B. Channel State Information Preprocessing and Offline Windowing

Following standard CSI extraction and preprocessing practice for Wi-Fi sensing [11], [12], each recording is stored as a

Authorized licensed use limited to: Polytechnic University of Bucharest. Downloaded on October 08,2026 at 12:44:23 UTC from IEEE Xplore. Restrictions apply.

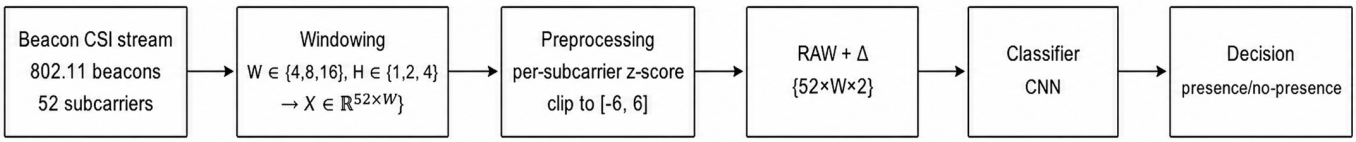


Fig. 1. Overview of the proposed pipeline: continuous CSI stream acquisition, offline re-windowing for different (W, H) pairs, RAW+Δ sample construction, CNN-based classification, and LOSO evaluation.

continuous CSI stream containing the real and imaginary parts of the complex channel response for the 52 active OFDM subcarriers, together with packet timestamps. Let

$$\mathbf{H}(n) \in \mathbb{C}^{52} \tag{1}$$

denote the CSI vector extracted from the n-th received beacon. For each subcarrier, we compute the amplitude magnitude and express it in decibels:

$$X(k,n) = 20\log_{10}\left(|H(k,n)| + \varepsilon\right), \tag{2}$$

where $k \in \{1,\ldots,52\}$ is the subcarrier index, n is the beacon index, and ε is a small constant used for numerical stability. This produces, for each class recording, a continuous amplitude stream

$$\mathbf{X}_{\text{stream}} \in \mathbb{R}^{52\times N},$$

where N is the number of valid captured beacons in that recording.

Starting from this continuous stream, samples are generated offline using sliding temporal windows of length W and hop size H, both expressed in beacons. Each sample is defined as

$$\mathbf{X}_m = \mathbf{X}_{\text{stream}}(:,\, s_m : s_m + W - 1) \in \mathbb{R}^{52\times W}, \tag{3}$$

where s_m denotes the start index of the m-th window. In this work, we generate datasets for all combinations

$$W \in \{4,8,16\}, \qquad H \in \{1,2,4\}.$$

For each session, class, and (W, H) pair, all candidate windows are first enumerated from the saved CSI stream using start indices spaced by the selected hop size. A window is considered valid if it contains W consecutive retained CSI packets from the same stream. Since the continuous stream is built only from successfully processed beacon packets and each retained packet already contains a complete 52-subcarrier CSI estimate, no additional packet-level completeness filtering is applied during offline sample construction. From the available candidates, exactly 200 windows are retained approximately uniformly over the full stream duration.

This offline re-windowing strategy is important for two reasons. First, it ensures that all compared datasets originate from the same underlying CSI recordings, so differences between configurations can be attributed to the temporal parameters (W, H) rather than to repeated acquisitions. Second, because the leave-one-session-out split is performed at the session level, no window from the held-out test session appears during training, which prevents cross-session leakage [5], [6].

The saved packet timestamps are not used to define the windows themselves, which are constructed in beacon-index space from the retained packet sequence. Instead, timestamps are used only to interpret the temporal meaning of the selected W and H values.

C. CNN Input Construction

For each windowed CSI sample $\mathbf{X}_m \in \mathbb{R}^{52\times W}$ obtained as described in Sec. III-B, the CNN input is constructed as a two-channel tensor that combines the CSI amplitude map and its first-order temporal difference. This choice follows the intuition that the raw amplitude preserves the instantaneous channel structure, while the temporal difference emphasizes short-term variations caused by human motion, in line with prior CSI deep-learning workflows [12].

Let

$$\mathbf{X} \in \mathbb{R}^{52\times W}$$

denote one windowed CSI amplitude sample in dB. A temporal-difference map is then computed along the beacon axis as

$$\Delta X(k,t) = \begin{cases} 0, & t = 1, \\ X(k,t) - X(k,t-1), & t = 2,\ldots,W, \end{cases} \tag{4}$$

where k is the subcarrier index and t is the time index inside the window.

To reduce per-subcarrier gain offsets and scale differences, both the raw amplitude map and the temporal-difference map are normalized independently, per subcarrier and within the current window, using z-score normalization. After normalization, both channels are clipped to the range $[-6, 6]$ in order to reduce the influence of outliers. Since these operations are applied independently for each sample, no statistics are shared across sessions or folds.

The final CNN input is obtained by stacking the normalized raw map and the normalized temporal-difference map along the channel dimension:

$$\mathbf{Z}_{\text{CNN}} = \left[\tilde{\mathbf{X}}, \widetilde{\Delta\mathbf{X}}\right] \in \mathbb{R}^{52\times W\times 2}. \tag{5}$$

Thus, for each choice of window length $W \in \{4,8,16\}$, the network receives a tensor of size $52 \times W \times 2$. This representation is used identically for all evaluated hop sizes $H \in \{1,2,4\}$, so that the analysis isolates the effect of temporal windowing rather than changes in feature design.

Authorized licensed use limited to: Polytechnic University of Bucharest. Downloaded on October 08,2026 at 12:44:23 UTC from IEEE Xplore. Restrictions apply.

D. Learning Model

A single lightweight CNN is used in all experiments. The input is the two-channel tensor

$$\mathbf{Z}_{\mathrm{CNN}} \in \mathbb{R}^{52\times W\times 2},$$

where $W \in \{4, 8, 16\}$ is the temporal window length.

The network architecture is fixed for all configurations and is adapted from the MATLAB wireless sensing example in [12]. It consists of:

- an input layer with `Normalization="none"`;
- a 3×3 convolution with 8 filters, batch normalization, ReLU, and 2×2 max pooling;
- a 3×3 convolution with 16 filters, batch normalization, ReLU, and 2×2 max pooling;
- a 3×3 convolution with 32 filters, batch normalization, ReLU, and dropout with probability $p = 0.1$;
- a fully connected layer with 2 outputs, followed by softmax and a classification layer.

Training is performed with Adam using a learning rate of 10^{-3}, mini-batch size 8, and 20 epochs, with shuffling at every epoch. In each LOSO fold, the model is trained from scratch using only the training sessions.

E. Evaluation Protocol and Metrics

We evaluate cross-session generalization using a strict LOSO protocol over the four recording conditions: *baseline*, *small-shift*, *NLoS*, and *room2*.

This protocol is important because windows extracted from the same continuous stream are strongly correlated, especially for small hop sizes. By splitting the data at the session level, the model is always evaluated on an unseen recording condition.

All preprocessing and sample construction steps are applied independently within each recorded stream, and no window from the held-out session is used during training. The same CNN architecture and the same training hyperparameters are used for all folds and all temporal configurations.

For each held-out session, we compute true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN), where the *presence* class is treated as positive. From these values, we report accuracy (Acc), F1-score (F1), and Matthews correlation coefficient (MCC):

$$\mathrm{Acc} = \frac{\mathrm{TP}+\mathrm{TN}}{\mathrm{TP}+\mathrm{TN}+\mathrm{FP}+\mathrm{FN}}, \tag{6}$$

$$\mathrm{Prec} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}}, \qquad \mathrm{Rec} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}}, \tag{7}$$

$$\mathrm{F1} = \frac{2\,\mathrm{Prec}\cdot\mathrm{Rec}}{\mathrm{Prec}+\mathrm{Rec}}, \tag{8}$$

$$\mathrm{MCC} = \frac{\mathrm{TP}\cdot\mathrm{TN}-\mathrm{FP}\cdot\mathrm{FN}}{\sqrt{(\mathrm{TP}+\mathrm{FP})(\mathrm{TP}+\mathrm{FN})(\mathrm{TN}+\mathrm{FP})(\mathrm{TN}+\mathrm{FN})}}. \tag{9}$$

Final performance is reported as mean $\pm$ standard deviation across the four folds.

TABLE II
OVERALL LOSO PERFORMANCE FOR THE NINE TEMPORAL WINDOWING CONFIGURATIONS (MEAN $\pm$ STD OVER 4 FOLDS).

W	H	Acc (%)	F1	MCC
4	1	75.31±10.63	0.767±0.097	0.511±0.210
4	2	76.31±7.80	0.777±0.060	0.530±0.151
4	4	78.69±7.26	0.799±0.056	0.581±0.140
8	1	80.19±12.09	0.809±0.120	0.616±0.232
8	2	82.69±9.98	0.838±0.090	0.663±0.193
8	4	81.94±10.34	0.834±0.087	0.646±0.200
16	1	85.94±12.25	0.878±0.097	0.738±0.217
16	2	86.19±11.85	**0.881±0.096**	**0.747±0.212**
16	4	**86.44±12.10**	0.876±0.109	0.744±0.225

Note: The best value in each metric column is shown in bold.

Table II summarizes the overall LOSO performance for all nine temporal windowing configurations, reported as mean $\pm$ standard deviation over the four held-out sessions. The results show a clear dependence on the window length W. The best mean performance is obtained for $W = 16$, while the differences between hop sizes $H \in \{1, 2, 4\}$ are comparatively small. This suggests that enlarging the temporal context is more beneficial than changing the window stride.

This trend is also visible in Fig. 2, which plots the overall MCC as a function of W for different values of H. MCC increases consistently from $W = 4$ to $W = 16$ for all hop sizes, whereas the separation between the H-curves remains modest.

To examine session-level behavior, Fig. 3 reports the per-session accuracy for the best-performing configuration. The baseline and small-shift sessions achieve the highest accuracies, while the NLoS and room2 sessions remain more difficult. This indicates that the main degradation is caused by stronger propagation changes rather than by temporal windowing alone.

Overall, the results show that longer temporal windows improve cross-session robustness, but severe environment shifts still limit generalization.

IV. DISCUSSION

A. Main Findings

The experiments show that temporal windowing has a clear impact on cross-session Wi-Fi CSI presence detection. Across all tested hop sizes, increasing the window length from $W = 4$ to $W = 16$ leads to a consistent improvement in overall LOSO performance. This trend is visible both in Table II and in Fig. 2, where MCC increases steadily with window length.

By contrast, the influence of hop size is weaker. For a fixed W, changing H from 1 to 4 produces only modest differences in Acc, F1, and MCC. This suggests that, in the tested setup, the dominant design factor is the temporal context included in each sample rather than the exact spacing between adjacent windows.

Authorized licensed use limited to: Polytechnic University of Bucharest. Downloaded on October 08,2026 at 12:44:23 UTC from IEEE Xplore. Restrictions apply.

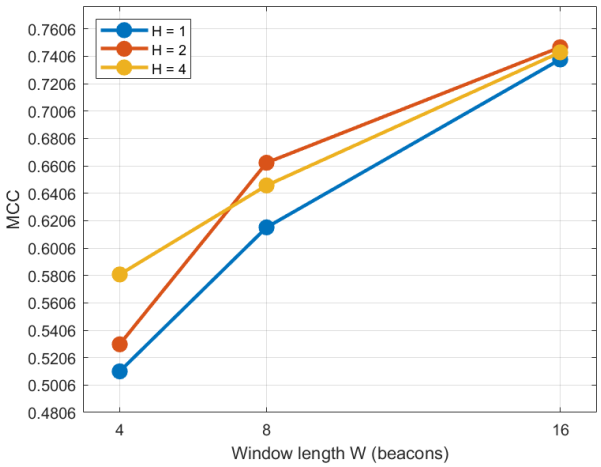


Fig. 2. Overall LOSO performance in terms of MCC for all temporal windowing configurations. The curves show the mean MCC versus window length $W \in \{4, 8, 16\}$ for each hop size $H \in \{1, 2, 4\}$, while the error bars indicate the standard deviation across the held-out sessions.

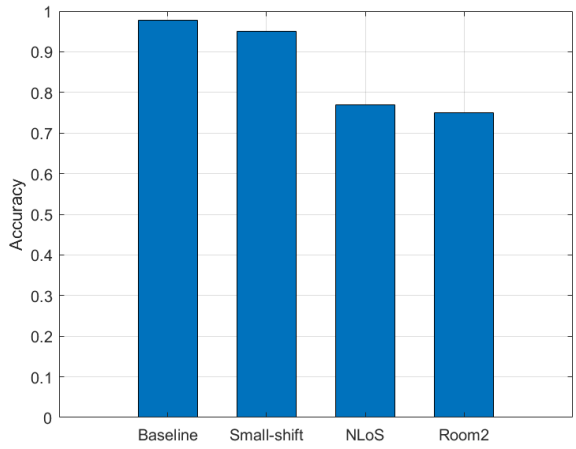


Fig. 3. Per-session LOSO accuracy for the best-performing temporal windowing configuration. The bars correspond to the held-out sessions (*baseline*, *small-shift*, *NLoS*, and *room2*), highlighting that the largest degradation occurs under the more severe environment shifts.

B. Why Longer Windows Help

A longer temporal window gives the model access to a more complete local evolution of the CSI response. In device-free sensing, human presence is not reflected by a single isolated fluctuation, but by a short sequence of channel perturbations distributed over multiple packets [1], [3]. When W is too small, the network may observe only a fragment of that pattern, which makes the classification less stable under session shift.

This explains why $W = 16$ yields the best average performance in our experiments. The larger window likely captures a more informative motion signature and reduces the sensitivity to short-term fluctuations or weak packet-level variations. In practical terms, the results indicate that the model benefits more from richer temporal context than from denser overlap between successive windows.

C. Limited Influence of Hop Size

The effect of hop size is comparatively small. For each fixed window length, the three hop values lead to similar average performance, especially for $W = 8$ and $W = 16$. This indicates that once the temporal context is sufficiently informative, modifying the overlap between adjacent windows does not substantially change the final cross-session behavior.

Still, hop size should not be interpreted as irrelevant. A smaller hop produces more overlapping windows and therefore samples the stream more densely, while a larger hop reduces overlap and yields a sparser set of examples. In the present study, however, this difference is secondary compared with the effect of window length itself.

D. Session-Level Robustness

The per-session results in Fig. 3 show that the baseline and small-shift sessions remain the easiest conditions, while the more difficult cases are the NLoS and room2 sessions. This is consistent with the intuition that moderate placement changes are easier to tolerate than stronger propagation changes caused by obstruction or deployment in a different room [1], [5], [6].

These results confirm that the main challenge is not simply detecting presence in a fixed environment, but maintaining performance when the wireless channel statistics change across sessions. Even under the best temporal configuration, the largest degradation still appears in the sessions associated with stronger environment shift.

E. Limitations and Future Work

This study is intentionally focused and therefore has several limitations. The dataset contains only four recording conditions, a single subject, and one acquisition setup, so the conclusions should be interpreted as a controlled analysis of temporal windowing rather than a universal rule for all CSI sensing systems. In addition, only one CNN pipeline and one input representation (RAW+Δ) are considered. Consequently, the observed preference for longer windows may not transfer unchanged to other subjects, motion patterns, or model families; in particular, a stronger architecture could respond differently to the tested windowing choices.

Future work should extend the analysis to more rooms, more users, and more diverse motion patterns. It would also be useful to test whether the same trends remain valid for other model architectures or under lightweight calibration in a new environment, as suggested by prior work on transfer learning and robustness in wireless sensing [8], [13]. Finally, a broader sweep over larger values of W could clarify whether the observed gain saturates beyond $W = 16$ or whether additional temporal context continues to improve robustness.

Authorized licensed use limited to: Polytechnic University of Bucharest. Downloaded on October 08,2026 at 12:44:23 UTC from IEEE Xplore. Restrictions apply.

V. Conclusion

This work analyzed the effect of temporal windowing on Wi-Fi CSI-based presence detection under strict LOSO evaluation. By re-windowing the same continuous CSI streams with different (W, H) pairs and evaluating all configurations using the same lightweight CNN, we showed that window length has a stronger impact than hop size on cross-session performance. In particular, longer windows consistently improved robustness, with the best results obtained for $W = 16$.

The experiments also showed that NLoS and cross-room conditions remain the most difficult deployment scenarios. Overall, the study indicates that temporal windowing is a meaningful design choice in CSI sensing systems and should be selected using session-independent evaluation rather than convenience alone.

VI. Acknowledgment

This work was supported by the European Union – NextGenerationEU, through the National Recovery and Resilience Plan (PNRR), Component 9, Investment 4, under project SENSE, THINK @ POLITEHNICA BUCUREȘTI (SENTHIPoli) No. 6.PI/I4/C9.

The content of this material does not necessarily represent the official position of the European Union or the Government of Romania

References

[1] Y. Ma, G. Zhou, and S. Wang, "Wifi sensing with channel state information: A survey," *ACM Computing Surveys*, 2019.

[2] Y. Wang, J. Liu, Y. Chen, M. Gruteser, J. Yang, and H. Liu, "E-eyes: Device-free location-oriented activity identification using fine-grained wifi signatures," in *Proceedings of the 20th Annual International Conference on Mobile Computing and Networking (MobiCom '14)*, 2014.

[3] W. Wang, A. X. Liu, M. Shahzad, K. Ling, and S. Lu, "Understanding and modeling of wifi signal based human activity recognition," in *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking (MobiCom '15)*, 2015.

[4] S. Palipana, P. Agrawal, and D. Pesch, "Channel state information based human presence detection using non-linear techniques," in *Proceedings of the 3rd ACM International Conference on Systems for Energy-Efficient Built Environments (BuildSys '16)*, 2016.

[5] J. Zhang, Z. Tang, M. Li, D. Fang, P. Nurmi, and Z. Wang, "Crosssense: Towards cross-site and large-scale wifi sensing," in *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking (MobiCom '18)*, 2018.

[6] Y. Zheng, Y. Zhang, K. Qian, G. Zhang, Y. Liu, C. Wu, and Z. Yang, "Widar3.0: Zero-effort cross-domain gesture recognition with wi-fi," in *Proceedings of the 17th Annual International Conference on Mobile Systems, Applications, and Services (MobiSys '19)*, 2019, accessed 2026-02-09. [Online]. Available: https://kunqian.info/publications/widar3.0.pdf

[7] Z. Yang, Y. Zhang, K. Qian, and C. Wu, "SLNet: A spectrogram learning neural network for deep wireless sensing," in *Proceedings of the 20th USENIX Symposium on Networked Systems Design and Implementation (NSDI 23)*, 2023, accessed 2026-02-09. [Online]. Available: https://www.usenix.org/conference/nsdi23/presentation/yang-zheng

[8] W. Hou and C. Wu, "RFBoosT: Understanding and boosting deep wifi sensing via physical data augmentation," *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol. (IMWUT)*, vol. 8, no. 2, May 2024.

[9] MicroPhase Technologies, "Antsdr: Documentation and software (e200 family)," https://github.com/MicroPhase/antsdr_doc, accessed 2026-02-09.

[10] Analog Devices, "libiio documentation," https://analogdevicesinc.github.io/libiio/, accessed 2026-02-09.

[11] D. Halperin, W. Hu, A. Sheth, and D. Wetherall, "Tool release: Gathering 802.11n traces with channel state information," *ACM SIGCOMM Computer Communication Review*, 2011.

[12] MathWorks, "Detect human presence using wireless sensing with deep learning," https://www.mathworks.com/help/wlan/ug/detect-human-presence-using-wireless-sensing-with-deep-learning.html, MATLAB Documentation (WLAN Toolbox), accessed 2026-02-09.

[13] J. Yang, Y. Xu, H. Cao, H. Zou, and L. Xie, "Deep learning and transfer learning for device-free human activity recognition: A survey," *Journal of Automation and Intelligence*, vol. 1, no. 1, p. 100007, Dec. 2022.

Authorized licensed use limited to: Polytechnic University of Bucharest. Downloaded on October 08,2026 at 12:44:23 UTC from IEEE Xplore. Restrictions apply.