An Asynchronous Heterogeneous Fusion Framework for RF Localization Ensembles

Alexander I-Chi Lai and Ruey-Beei Wu

Abstract – This study proposed a real-time deep fusion framework for RF positioning ensembles by merging heterogeneous sensor sources and positioning algorithms to create a better combination for a localization solution. The current incarnation of the proposed fusion framework uses video localization, Wi-Fi positioning, motion filters, and a dynamic matching and weighting mechanism in an asynchronous manner. Real-world experiments show that the resulting ensemble reduces the mean error to 0.83 m in real time and 0.62 m for off-line use.

1. Introduction

In terms of localization, ensemble solutions are frequently used to achieve better positioning accuracy than is possible using any single-constituent positioning approach [1–4]. An ensemble fusing multiple sources, algorithms, and weighting functions should be no worse than simply using any single element of the positioning ensemble. A positioning ensemble can be realized by a fusion framework in either the sensor-input, positioning algorithms, or output-filter stages, or any feasible combination of these.

Multiple sensor readings for more than one positioning algorithms are crucial to positioning efficacy because the different solutions complement each other. Specifically, observe the following:

- Not every signal or sensor reading for positioning is available at all scenes. Notably, global navigation satellite system signals are unavailable indoors.
- Different signals for positioning perform well in specific circumstances but poorly otherwise. For example, millimeter-wave signals give a high resolution only at short range.
- Positioning algorithms perform diversely in different environments. Proximity algorithms are used for point presence within a short range of specific landmarks, in contrast to scene analysis or multilateration [5–11].

Fusion approaches [1–4] use different sensors or algorithms to ensure that the overall accuracy is no worse than the accuracy of any composition in the ensemble. Constructing an effective fusion framework is technically challenging because different positional methods can give different estimations of the actual location for any tracked target object. This difference between the estimated positions and the lack of a universal target-identification mechanism means that a successful fusion framework must identify from many inbound estimated positions which combinations of the estimated information are possible and effective. An incorrect combination jeopardizes the entire localization task. Moreover:

- The frequency with which information is updated and the amount of carried information varies for different types of positioning signal. An optical video frame that is captured using a typical security camera updates in tens of milliseconds but the received signal strength for Wi-Fi signals requires much longer (hundreds to thousands of milliseconds) to stabilize. Therefore, the sensor inputs and the related processing algorithms are intrinsically asynchronous.
- Some sensors, such as the inertial measurement unit (IMU) in many mobile devices, give only partial or motion information and no positional information, so these sensor readings do not directly combine other positional information unless there is pre- or postprocessing.
- The relative weights of positioning solutions must be adaptable to allow automatic localization in real time. Most previous fusion frameworks use specific rigid, preprogrammed mechanisms to combine the outcomes from each constituent positioning algorithm in the ensemble, so positioning performance is poor in complex indoor environments.

This study proposes a reconfigurable heterogeneous fusion framework to formulate flexible and expandable ensembles of localization solutions. Unlike previous ones, this framework adaptively adjusts the relative weights and the data-flow paths for various sensor inputs, positioning algorithms, and output filtering functions in an asynchronous manner.

2. The Fusion Framework

The proposed solution, following the unified fusion framework model [1], comprises three cascaded stages, as shown in Figure 1: a sensor-input processing stage, an algorithmic stage, and the final weight-adjusting output stage. During the sensor-input processing stage, the characteristics of each perceived signal or sensor source are unified in the information domain by

Manuscript received 27 December 2021.

Alexander I-Chi Lai and Ruey-Beei Wu are with the Department of Electrical Engineering, National Taiwan University, No. 1, Section 4, Roosevelt Rd., Da'an District, Taipei City, 10617, Taiwan; e-mail: alexiuslai@ntu.edu.tw, rbwu@ntu.edu.tw.

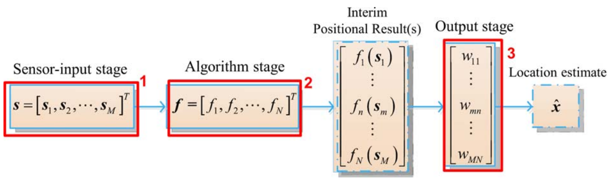

Figure 1. The unified fusion framework model to formulate an RF localization ensemble.

extracting the information entropy to calculate the local information gains. During the algorithmic stage, each positioning-algorithm module uses specific unified sensor inputs to estimate the location with a confidence level. In the final output stage, the outcomes of all positioning-algorithm modules are subject to a weighted combination process to give a final estimation of the position.

The design of the proposed fusion framework is shown in Figure 2. Heterogeneous sensor information and positioning algorithms are merged. On the leftmost side, outputs from multiple positioning algorithms and additional sensor data are fed into the enhanced motion filters.

The filters calculate the interim positions for the next stage—that is, the matching mechanism—to determine which are associated with the same object under tracking. This process is repeated several times to achieve an optimal result using the feedback loop, and gives the best matches for positions that are determined using different algorithms. These interim locations and possible paths are cached in the system memory to allow fast access. When the latest fused location is determined, it is concatenated to the long-term path that can be visualized and saved for further off-line use.

In the final output stage, we crafted a loss function called *MatchingLoss* which is an aggregated measure for the position of the current device under targeting (DUT) that is derived using its deviation from the positions of alternative DUTs over a certain time period and its deviation from the positions of the current DUT at all previous times. This is used to dynamically adjust the data flows and the corresponding weights for

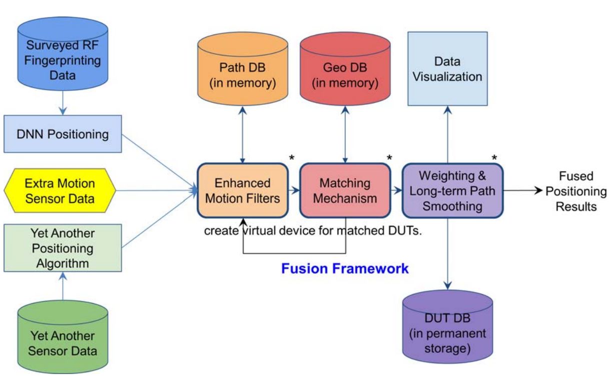

Figure 2. The fusion framework to merge heterogeneous sensor information and positioning algorithms.

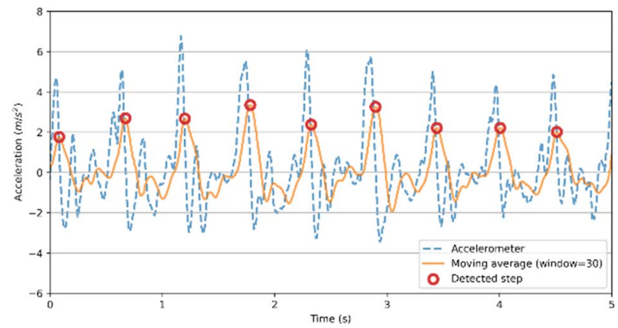

Figure 3. A filtered IMU and PDR are used to estimate average step length and frequency.

combinations of path histories that could belong to the same target, instead of using the weighted average.

To address asynchronous challenges, the proposed fusion framework used a message queue (MQ) to ensure reliable and efficient delivery of information between stages and constituent modules by Apache RabbitMQ [12]. The MQ links enhanced Kalman and particle filters that use pedestrian dead reckoning (PDR), a deep neural network (DNN)-based infrastructure-less positioning mechanism that is detailed in a previous study of ours [13, 14], a multiobject optical video positioning module [15] with Deep *SORT* (Simple Online and Realtime Tracking) [16], and the *MatchingLoss* function for weighting and the output.

3. Enhanced Motion Filtering

Most modern mobile devices are equipped with an IMU comprising an accelerometer, a gyroscope, and a magnetometer to give additional motion information about tracked targets. The filtered IMU readings and PDR are used to estimate the average step length and the stepping frequency, as shown in Figure 3. The filtered acceleration (orange lines in Figure 3) and the stumping of steps (red circles in Figure 3) are used to determine whether the measurements achieve a threshold value to detect a walking status. The gyroscope and magnetometer give the heading of the tracked target. Using motion information from the IMU, the RF positioning results achieve better continuity, so trajectories can be matched.

To combine the motion-status information and positioning estimation, filters smooth the paths using the IMU readings. The fusion framework incorporates sophisticated filters such as Kalman and particle filters, with PDR and other enhancements to complement the particle filter, including the following:

- Angle detection. Cosine similarity is applied to determine whether the progression direction for the positioning algorithm(s) is aligned with the IMU+PDR. For small angular offsets below (heuristically set) 60°, the filtered IMU+PDR heading is used in conjunction with the progression direction given by the positioning algo-

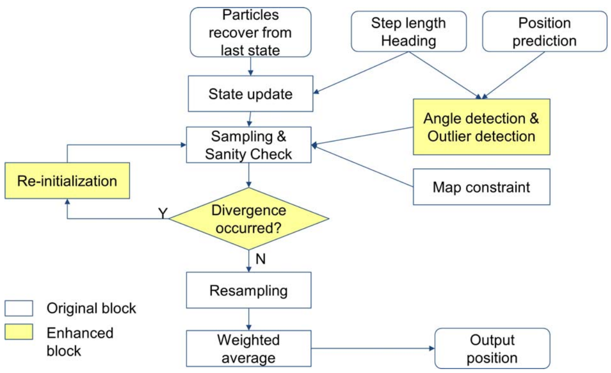

Figure 4. Operational flowchart for the enhanced particle filter.

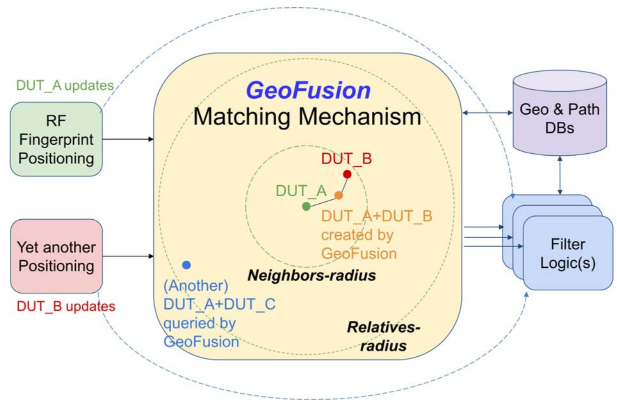

Figure 5. The *GeoFusion* matching mechanism.

rithm(s). Otherwise, the framework resets the particle filter and recalculates the combined direction.

- Step-length detection. Similar to angle detection, a default threshold (70 cm) is used to determine whether the estimated moving distance aligns with the filtered IMU+PDR results, or the filter is reset.
- Divergence detection [17]. The particle-filtered IMU estimation and the positional algorithm's estimation align if both are accurate. That is, if the spreading areas of the particles have little intersection with the positioning algorithm's estimation, the framework resets the particle filter to recalculate the intersection. The particle-filtered IMU estimation and the positional algorithm's estimation are modeled using two random variables to calculate their Mahalanobis distance for the intersection and the outliers.

Refer to Figure 4 for the operational flowchart of the enhanced particle filter with these enhancements.

4. Matching of Heterogeneous Positioning

The localization fusion framework must match different location estimations for the same target that are derived using heterogeneous positioning algorithms, without identifying the target a priori. If two location estimations move together in roughly the same direction, it is very likely that they refer to the same target.

In this study, we accordingly devised a matching mechanism called *GeoFusion* to examine the geographical proximities between location estimations from the positioning algorithms. The operation of *GeoFusion* is shown in Figure 5. Two different positioning algorithms—say an RF fingerprint positioning (green) and another localization method (red)—give respective positional estimations DUT_A and DUT_B. Two proximity thresholds are then defined: a smaller "neighbors-radius" and a larger "relatives-radius," based on the characteristics of the participant positioning algorithms.

If DUT_A and DUT_B move together within the neighbors-radius, the fusion framework composes a new virtual object called DUT_A+DUT_B, with the unified location estimation as the weighted average of the locations of DUT_A and DUT_B. Otherwise, if they separate to the threshold of the relatives-radius, the framework distinguishes between the two estimated positions and decomposes the created virtual object. For the virtual object, the Kalman filter is used to fuse the two positioning results using the predetermined constant variance value.

During such composition and decomposition processes, the interim locations and final paths for the tracked targets, actual or virtual, may pass through additional filter modules, and are respectively recorded into the in-memory geolocation ("Geo") and path ("Path") databases for further optimization. One example is an off-line optimization mechanism replaying the path forward and backward through the enhanced particle filter. The final resultant paths, synthesized by inverse variance weighting of both replays, are much smoother because the two replays suppress different trajectory errors.

5. System Realization and Results

To validate the effectiveness of the fusion framework, a reconfigurable and scalable system called AIPS (AIoT-Based Infrastructure-less Positioning System) was implemented. This is an overhauled architecture from our previous study [14]. The Python-based AIPS fusion framework, running on Google Kubernetes, was equipped with a JavaScript Web graphical user interface (GUI) that enables visualization and analysis of positioning information using fingerprints, 3D heat maps, information gains, and path replays.

Figure 6 shows a screenshot of the AIPS front-end GUI, with a floor map of one test site in the middle section and two different signal fingerprints at some selected locations of interest in the right portion. In the 3D heat maps in Figure 7, the received-signal strength indicator (in orange-purple) is not propor-

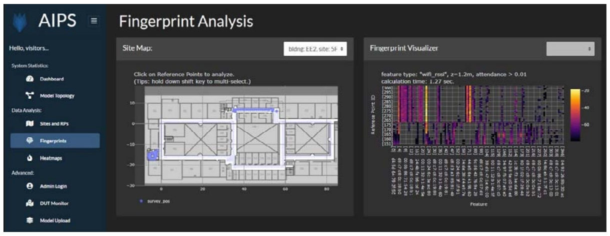

Figure 6. A screenshot of the AIPS localization system.

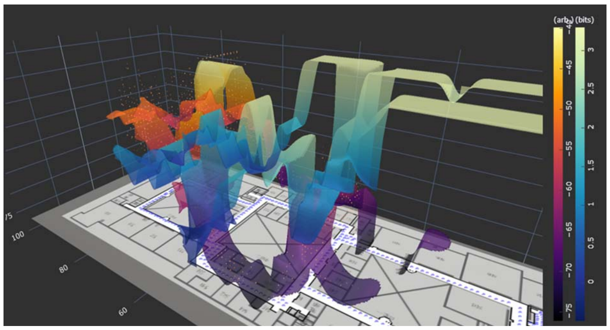

Figure 7. A 3D heat-map view of the fingerprints.

tional to the amount of contributed information (in yellow-blue), indicating that information entropy may yield a better resolution in fingerprint-based localization.

Figure 8a illustrates the real-world experimental setup on the fifth floor of the EE2 Building at National Taiwan University, where a person with a Google Pixel 4 smartphone remained in step with a site-surveying robot [14], traversing from the start to stop positions. The ground truth was constructed by the onboard lidar of the robot. Two cameras were mounted on the walls of the corridor. A custom-made Android app to capture Wi-Fi fingerprints ran on the Pixel 4 and communicated with AIPS. The raw Wi-Fi fingerprint positioning results using a DNN are shown in Figure 8b as the baseline. The line segments that connect each dot on the robot's trace denote the error distances. The particle-filtered IMU+PDR Wi-Fi results are shown in Figure

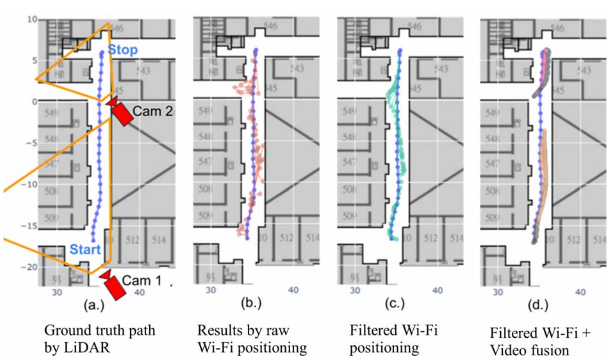

Figure 8. Experimental setup and comparison of different fusion schemes.

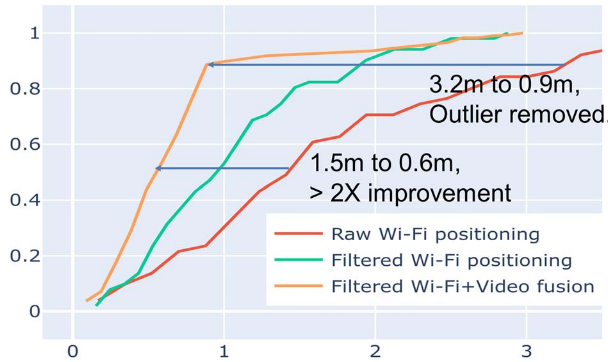

Figure 9. The accuracy of different fusion schemes.

8c, and the fusion results of particle-filtered IMU+PDR Wi-Fi with video positioning are shown in Figure 8d. The two other routes in different colors in Figure 8d show the camera-only initial tracking results for the same person. Without fusion with Wi-Fi positioning, it is difficult to determine whether the target is the same person because the viewing angles vary.

Note that in the experiment setup, the video positioning algorithm samples the DUT location three to nine times more frequently than the Wi-Fi RF positioning. Moreover, the much larger Wi-Fi RF positioning error, reaching an average of 1.08 m [14], dominated the fused positional error. That is, in the extreme scenarios, the Wi-Fi (and fusion) positional error may accumulate three to nine times due to the stale Wi-Fi localization estimation, while the video positioning has already updated three to nine times. Thus in *GeoFusion* we set the neighbors-radius and relatives-radius to, respectively, (1.08 m) $\times$ 3 and (1.08 m) $\times$ 9, rounded to the closest integers.

A quantitative comparison is shown in Figure 9. Using motion filtering and heterogeneous fusing of Wi-Fi and video positioning results, the mean errors decrease to 0.83 m in real time and 0.62 m for offline use, achieving approximately a twofold improvement over the baseline DNN results.

6. Conclusions

The proposed fusion framework achieved better localization results by merging deep-learning-based fingerprint positioning with enhanced motion filters and additional optical video positioning mechanisms. This strategy is capable of unifying more localization methodologies, such as ranging in time-sensitive networking for the upcoming IEEE 802.11be (Wi-Fi 7) [18, 19]. The dynamic weight-adjustment mechanisms that are used to optimize the fusion of multiple localization solutions deserve further study.

7. Acknowledgement

The authors are indebted to Steven Wu and his colleagues at DFI Inc., New Taipei, Taiwan, for financial support and technical guidance.

8. References

1. X. Guo, N. Ansari, F. Hu, Y. Shao, N. R. Elikplim, et al., "A Survey on Fusion-Based Indoor Positioning," *IEEE Communications Surveys & Tutorials*, **22**, 1, 2020, pp. 566-594.
2. J. Torres-Sospedra, A. Moreira, G. M. Mendoza-Silva, M. J. Nicolau, M. Matey-Sanz, et al., "Exploiting Different Combinations of Complementary Sensor's Data for Fingerprint-Based Indoor Positioning in Industrial Environments," 2019 International Conference on Indoor Positioning and Indoor Navigation (IPIN), Pisa, Italy, September 30–October 3 2019, pp. 1-8.
3. L.-H. Chen, E. H.-K. Wu, M.-H. Jin, and G.-H. Chen, "Intelligent Fusion of Wi-Fi and Inertial Sensor-Based Positioning Systems for Indoor Pedestrian Navigation," *IEEE Sensors Journal*, **14**, 11, November 2014, pp. 4034-4042.
4. D. Han, S.-h. Jung, and S. Lee, "A Sensor Fusion Method for Wi-Fi-Based Indoor Positioning," *ICT Express*, **21**, 2, June 2016, pp. 71-74.
5. F. Zafari, A. Gkelias, and K. K. Leung, "A Survey of Indoor Localization Systems and Technologies," *IEEE Communications Surveys and Tutorials*, **21**, 3, 2019, pp. 2568-2599.
6. A. Khalajmehrabadi, N. Gatsis, and D. Akopian, "Modern WLAN Fingerprinting Indoor Positioning Methods and Deployment Challenges," *IEEE Communications Surveys & Tutorials*, **19**, 3, 2017, pp. 1974-2002.
7. A. Yassin, Y. Nasser, M. Awad, A. Al-Dubai, R. Liu, et al., "Recent Advances in Indoor Localization: A Survey on Theoretical Approaches and Applications," *IEEE Communications Surveys & Tutorials*, **19**, 2, 2017, pp. 1327-1346.
8. S. He and S.-H. G. Chan, "Wi-Fi Fingerprint-Based Indoor Positioning: Recent Advances and Comparisons," *IEEE Communications Surveys & Tutorials*, **18**, 1, 2016, pp. 466-490.
9. G. M. Mendoza-Silva, J. Torres-Sospedra, and J. Huerta, "A Meta-Review of Indoor Positioning Systems," *Sensors*, **19**, 20, October 2019, pp. 4507-4551.
10. H. Liu, H. Darabi, P. Banerjee, and J. Liu, "Survey of Wireless Indoor Positioning Techniques and Systems," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, **37**, 6, November 2007, pp. 1067-1080.
11. D. Lymberopoulos and J. Liu, "The Microsoft Indoor Localization Competition: Experiences and Lessons Learned," *IEEE Signal Processing Magazine*, **34**, 5, September 2017, pp. 125-140.
12. Apache Camel, "RabbitMQ," https://camel.apache.org/components/3.13.x/rabbitmq-component.html (Accessed 25 February 2022).
13. A. I-C. Lai and R.-B. Wu, "Infrastructure-less Intelligent RF Spatial Positioning," XXIIIrd URSI General Assembly and Scientific Symposium (GASS), Rome, Italy, August 29–September 5, 2020, p. A16-03.
14. A. I-C. Lai, C.-Y. Chen, and R.-B. Wu, "A Mutual-Information Based Infrastructure-less Radio Frequency Positioning System via Deep Learning," *URSI Radio Science Letters*, **2**, 2020, doi: 10.46620/20-0024.
15. G. Ciaparrone, F. L. Sánchez, S. Tabik, L. Troiano, R. Tagliaferri, et al., "Deep Learning in Video Multi-Object Tracking: A Survey," *Neurocomputing*, **381**, March 2020, pp. 61-88.
16. N. Wojke, A. Bewley, and D. Paulus, "Simple Online and Realtime Tracking with a Deep Association Metric," 2017 IEEE International Conference on Image Processing (ICIP), Beijing, China, September 17–20, 2017, pp. 3645-3649.
17. S. Xu, R. Chen, Y. Yu, G. Guo, and L. Huang, "Locating Smartphones Indoors Using Built-In Sensors and Wi-Fi Ranging With an Enhanced Particle Filter," *IEEE Access*, **7**, 2019, pp. 95140-95153.
18. E. Khorov, I. Levitsky, and I. F. Akyildiz, "Current Status and Directions of IEEE 802.11be, the Future Wi-Fi 7," *IEEE Access*, **8**, 2020, pp. 88664–88688.
19. T. Adame, M. Carrascosa-Zamacois, and B. Bellalta, "Time-Sensitive Networking in IEEE 802.11be: On the Way to Low-Latency WiFi 7," *Sensors*, **21**, 15, July 2021, pp. 4954-4973.