Towards Accurate Positioning in Multiuser Augmented Reality on Mobile Devices

Na Wang
George Mason University
Email: nwang4@gmu.edu

Haoliang Wang
Stefano Petrangeli
Viswanathan Swaminathan
Adobe Research
Email: {hawang, petrange, vishy}@adobe.com

Fei Li
Songqing Chen
George Mason University
Email: {fli4, sqchen}@gmu.edu

2022 IEEE International Symposium on Multimedia (ISM) | 978-1-6654-7172-5/22/$31.00 ©2022 IEEE | DOI: 10.1109/ISM55400.2022.00048

***Abstract*—Multiuser Augmented Reality (MuAR) is essential to implementing the vision of Metaverse for its capability to provide immersive and interactive experiences. In such experiences, peer positions are critical to understand each other's intentions and actions so as to guarantee the smooth cooperation among users. However, we find that the explicit peer positions provided by the current practice could be incomplete and/or inaccurate in some situations, which leads to the weakened spatial awareness. To achieve the accurate peer tracking in MuAR, we propose a novel multiple sensors information fusion method, CSA (Coordinate System Alignment), to detect and correct defective relative positions by the current practice. CSA firstly formulates problem of correcting erroneous positions into an overdetermined system, and then finds the solution by applying the simulated annealing algorithm to expedite the search process. The evaluation results show that CSA's ability to reduce errors significantly (58.3% on average) under long-term error duration, especially its advantage in reducing the relative direction errors. The result confirms the potential of CSA to provide reliable peer tracking in MuAR. Meanwhile, it does not impose extra restrictions on users' practice with current mobile devices in experiences.**

***Index Terms*—Augmented Reality, Multiuser AR, spatial awareness, tracking**

I. Introduction

With the pervasive adoption of mobile devices, Augmented Reality (AR) is expected to mainly provide immersive experiences on mobile devices [1]. Furthermore, with the introduction of new AR toolkits, and advances in hardware, AR gradually evolves to support multi-user experience [2], in which multiple devices share a common experience to achieve the cooperation and interaction. To facilitate the interaction in MuAR, it is important for participating devices to share the position, since it is often associated with peers' intentions and actions, and reflects the following consequences. It is particularly important in applications such as rescue and medical operations [3]. For example, firefighters or robots may rely on each other's positions to search for or coordinate their actions when saving trapped lives in difficult environments. Thus, the problem of the peer position computation is critical in such applications. For the single user experience, popular AR frameworks, such as Apple ARKit and Google ARCore [4], [5], provide explicit devices poses, including positions and orientations, computed by the Simultaneous Localization and Mapping (SLAM) [6]. However, the computed positions cannot be directly used to compute relative positions in MuAR, because they are located with respect to independent coordinate systems. In other words, in the single user experience, the coordinate system of the AR 3D world is constructed with respect to the initial pose of independent mobile devices [7]. So the computed information is not relative to each other and cannot be directly used to compute relative positions.

Previous studies employ various technologies to achieve the mobile devices localization [8]–[10]. For example, GPS can be helpful in assisting outdoor applications, for example, the streets or buildings. However, it does not work properly indoors, and its meter-level precision cannot support applications designed for the room-scale experiences. In the case of BLE or WiFi signals, the signal interference and multi-path effects cause significant estimation errors [8], [10]. Moreover, the proposed approaches requires additional hardware set up and time-consuming calibration. However, most AR experience is ad hoc, happening in random locations instead of the pre-calibrated space.

To achieve peer tracking in MuAR, the industry has proposed to utilize Ultra Wideband (UWB) chips, recently introduced to mobile devices. The high-frequency ability of the UWB chip is utilized for the devices communication [11]. To this end, the solution requires participating devices to be in proximity and there is no presence of solid obstacles between devices. The tracking support is helpful. However, the requirements imposed by the solution restrict the movement freedom of users, especially for MuAR, and can conflict with the design of the application developers. More importantly, we find in the previous study that the proposed support could produce incomplete or inaccurate positions in some cases [12].

The unreliability poses a new challenge to MuAR, and potentially hinder the progression of Metaverse. Instead of upgrading mobile devices with the more advanced hardware, we propose a novel economical approach named CSA (Coordinate Systems Alignment), to deal with the unreliability. The approach employs the device positions from the single user and multiuser AR experiences only. It firstly formulates the detection and correction of erroneous updates into an overdetermined system, achieving the alignment of multiple independent coordinate systems in MuAR. Then the approach finds the solution to the overdetermined system by employing

978-1-6654-7172-5/22/$31.00 ©2022 IEEE
DOI 10.1109/ISM55400.2022.00048

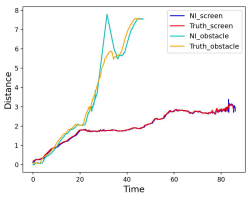
the simulated annealing algorithm, to avoid searching in an exponential space. In this way, the relative positions of peer devices are available by correcting incomplete or inaccurate reported positions. Finally, we collect the dataset about users' movement traces to evaluate the proposed solution. The results show the potential of CSA to reduce position errors by 58.3% on average under both short-term and long-term error duration, especially its advantage in reducing the relative direction errors. Therefore, it is safe to conclude CSA provides a viable solution to the legacy devices equipped with UWB chips running AR applications.

The rest of the paper is organized as follows. Sec. II briefly presents our investigation results about unreliability of the current practice. Sec. III discusses the related challenges and our corresponding solutions for the problem.The effectiveness of proposed solution CSA is evaluated in Sec. IV. The paper is concluded in Sec. V.

II. NI: Measurement Study

Nearby Interaction (NI) is the first industry solution for MuAR to provide peer tracking on iOS mobile devices equipped with UWB chips. The explicit peer position output is composed of two components: relative distance and relative direction. However, the convenience comes at a cost. The framework imposes a series of requirements on users' practice, in terms of activity sphere, device orientation and movement style, to guarantee it performs as expected: *first*, the maximum distance between any two peers is 9 meters; *second*, the screens of peer devices should be kept in the portrait mode; *third*, peer devices should appear within the line of sight of each other and there is no presence of obstacles between them [13].

We perform a series of experiments to investigate the reliability of NI. The key findings of the study are summarized as follows. First, the violation of the line of sight clearance results in inaccurate relative distances, shown in Figure 1, and unavailable direction reports, shown in Figure 2. Second, the influence of the screen orientation, shown in Figure 2, focuses on the availability of direction updates. The complete description of experiment setup and results is available in [12].



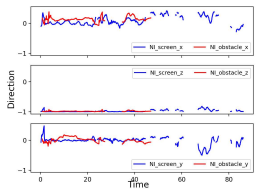


Fig. 1: Impact on the DistanceFig. 2: Impact on the Direction

Next, according to the magnitude of relative distance deviation, we organize the errors into three types: **Type I, transient error,** similar to the glitch, reporting slightly deviated distance; **Type II, persistent error,** as shown from t = 23 to 35 seconds in Figure 2, reporting the relative distances which deviate significantly from the truth; **Type III, moderate error,** is between the other two types. For all three types, the relative direction is missing.

III. Problem Space: Challenges and Solution

The measurement study shows the state-of-the-art support for MuAR is unreliable. Therefore, we aim to find an economical approach by leveraging the existing framework, instead of relying on upgraded hardware. In this section, we dissect the problem, discuss about related challenges, and present our solutions. Since the key is to align coordinate systems for multiple devices in AR experiences, the proposed solution is named as CSA (Coordinate System Alignment).

Challenge 1: Long-term unavailability of position updates. Predicting users' (or devices) positions in volumetric media is a well addressed problem. A natural solution is to predict future positions based on the past trace. However, the method mainly works in the case of short-term prediction horizon; once the horizon exceeds 2 seconds [14], [15], the results are no longer accurate. On the other hand, in our case, the error duration usually lasts much longer than that.

Solution: The reason for the traditional prediction working only in short-term horizon is there may exist drastic changes in users' movement. Thus the past trace only is not always a reliable predictor for the future. To obtain accurate relative positions in MuAR, it is necessary to know, at least partial or indirect, information of users' trace during the error duration. For this reason, we introduce the users' traces in single user AR experience, happening synchronously with MuAR. The trace can be obtained from the ARKit, a framework often used in single user AR applications development [4]. The coordinate system used for the ARKit is shown in Figure 3.

Challenge 2: Problem formulation. The self-position by ARKit is computed in the coordinate system constructed with respect to the initial pose of mobile devices. Therefore, the reported positions cannot be directly used to compute relative positions in MuAR. With the introduction of ARKit, the problem becomes how to use self positions in single user AR for the computation of relative positions in MuAR without the global coordinate system.

Solution: Our solution here is to introduce the alignment vector among different coordinate systems. More specifically, we suppose introduced ARKit positions are denoted as A and B for Device A and Device B. Each record in A and B has the same format $(time, x, y, z)$, representing device's position in the direction of x, y, and z at any specific $time$. Also, the relative positions by NI are denoted as A_{B} representing the relative position of Device A to B. The problem now can be formulated as the computation of alignment vectors Δ between two systems as the Eq. 1 shows.

$$A - (B + \Delta) = A_{\mathrm{B}} \tag{1}$$

Challenge 3: Alignment vector dissection. Once the alignment vector is solved, the ARKit position in single-user AR can be used to compute relative positions. However, solving Δ

is not straightforward. The coordinate system alignment in 3D space often involves multiple types of transforms, including rotation, scale and translation, as shown in Figure 4.

Solution: To describe the transform between two systems, we introduce the rotational, scaling and translational matrices, represented by R, S and T, respectively. Furthermore, since no scaling is involved in the specific problem, the matrix S can be seen as an identity matrix. Then the Eq. 1 can be further formulated as Eq. 2, an overdetermined system about 12 unknown variables from the rotational and translational matrices. The value of n depends on the available correct NI records in the experiment.

$$\begin{bmatrix} x_1^A \\ y_1^A \\ z_1^A \end{bmatrix} - \left(\begin{bmatrix} r_{xx} & r_{xy} & r_{xz} \\ r_{yx} & r_{yy} & r_{yz} \\ r_{zx} & r_{zy} & r_{zz} \end{bmatrix} \cdot \begin{bmatrix} x_1^B \\ y_1^B \\ z_1^B \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} \right) = \begin{bmatrix} x_1^{AB} \\ y_1^{AB} \\ z_1^{AB} \end{bmatrix}$$
$$\vdots$$
$$\begin{bmatrix} x_i^A \\ y_i^A \\ z_i^A \end{bmatrix} - \left(\begin{bmatrix} r_{xx} & r_{xy} & r_{xz} \\ r_{yx} & r_{yy} & r_{yz} \\ r_{zx} & r_{zy} & r_{zz} \end{bmatrix} \cdot \begin{bmatrix} x_i^B \\ y_i^B \\ z_i^B \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} \right) = \begin{bmatrix} x_i^{AB} \\ y_i^{AB} \\ z_i^{AB} \end{bmatrix}$$
$$\vdots$$
$$\begin{bmatrix} x_n^A \\ y_n^A \\ z_n^A \end{bmatrix} - \left(\begin{bmatrix} r_{xx} & r_{xy} & r_{xz} \\ r_{yx} & r_{yy} & r_{yz} \\ r_{zx} & r_{zy} & r_{zz} \end{bmatrix} \cdot \begin{bmatrix} x_n^B \\ y_n^B \\ z_n^B \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} \right) = \begin{bmatrix} x_n^{AB} \\ y_n^{AB} \\ z_n^{AB} \end{bmatrix} \quad (2)$$

Challenge 4: Solution space size. Although solving the overdetermined system is a well studied problem, the system setup is not easy. Because of the enormous quantity of available records, it is challenging to determine which records are used in computation for the better performance. For example, in the experiment violating the line of sight feature only, the total number of NI records is 1,942 for a session lasting 47 seconds. After rejecting erroneous records, 1,919 records remain available, and there would be $2^{1,919} - 1$ possible combinations. For each combination, it takes 2.53 to 16.96 seconds to solve, depending on the number of records used (The running time data comes from our measurement on a laptop with the 2.3 GHz Intel Core i5 processor and 8 GB memory). Therefore, it is not feasible to test all combinations.

Solution: To expedite the search process in the exponential solution space, we propose to formulate the problem of determining which records to be used as an optimization problem as the Eq. 3. M is the number of position records used in the system. P_{A} and P_{B} are positions of two mobile devices , while $\hat{P}_{\mathrm{A}}$ and $\hat{P}_{\mathrm{B}}$ are corresponding truth positions. The optimization goal is to find a solution set for matrices R and T, to minimize the sum of squares of distances between computed positions and corresponding truth. To this end, the simulated annealing algorithm is employed. In our experience, the algorithm provides a good trade-off between accuracy and speed [16].

$$\textbf{Minimize}: \sum_{1}^{M} \left\| [P_A - (R \cdot P_B + T)] - \left(\hat{P}_A - \hat{P}_B \right) \right\|^2 \quad (3)$$

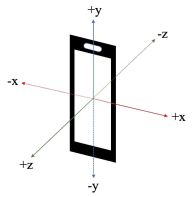


Fig. 3: AR Coordinate System

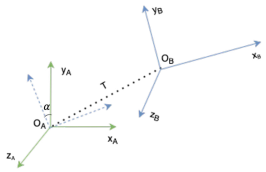


Fig. 4: Transformation

IV. Evaluation

The traces used for the evaluation, including users' movement traces and ground truth locations, are collected in the same way as the motivation study in Sec. II. We adopt the CUWB system for the ground truth collection. More detailed setup information is available in [12]. For the collected truth positions, the cubic spline interpolation is employed.

A. Evaluation Metrics

To evaluate the effectiveness of CSA, we use Mean Squared Error (MSE) between computed values and ground truth defined as Eq. 4. N is the number of defective records for a specific trace. (x, y, z) are the ground truth positions of participating devices, while $(\hat{x}, \hat{y}, \hat{z})$ are positions computed by CSA. The solution is regarded to be more effective with the smaller MSE value. In addition, the effectivity of CSA on the correction of the relative distance and direction is measured in the distance error and direction error, as Eq. 5 and Eq. 6 show.

$$MSE = \frac{1}{N} \sum_{i=1}^{N} (x_i - \hat{x_i})^2 + (y_i - \hat{y_i})^2 + (z_i - \hat{z_i})^2 \quad (4)$$

$$MSE(dist) = \frac{1}{N} \sum_{i=1}^{N} (d_i - \hat{d_i})^2 \quad (5)$$

$$Mean(dir) = \frac{1}{N} \sum_{i=1}^{N} arccos \frac{\mathbf{v}_{AB} \cdot \mathbf{v'}_{AB}}{\|\mathbf{v}_{AB}\| \, \|\mathbf{v'}_{AB}\|} \quad (6)$$

B. Results

Table I shows the error distribution of collected traces. For six cases among them, only Type III error exists. For all other cases, Type II and Type III errors coexist. Type I error is not considered here since they are corrected by replacing with the mean value of neighbors for all methods.

Firstly, the effectivity of CSA is evaluated. For comparisons, we evaluate three other baseline cases: (1) *no prediction (NP)*, in which the mean value of previous 30 position reports is used to replace erroneous relative distance and/or direction;

(2) *dead reckoning (DR)*, employing the user's current velocity to predict the future movement [17]; (3) *autoregression model (AR)*, which uses three collected traces for training and selects the best-performing model.

TABLE I: Error Distribution for 18 Cases

User	1	2	3	4	5	6	7	8	9
TypeII	✗	✗	✓	✓	✗	✓	✓	✓	✓
TypeIII	✓	✓	✓	✓	✓	✓	✓	✓	✓
User	10	11	12	13	14	15	16	17	18
TypeII	✓	✓	✗	✓	✓	✓	✗	✗	✓
TypeIII	✓	✓	✓	✓	✓	✓	✓	✓	✓

Figure 7 illustrates the comparison results of different methods. The figure employs the natural logarithm of MSE values for the y axis. We can observe CSA outperforms all others for all cases. In particular, CSA reduces errors by 58.3% on average compared to DR, a typical method used in the position prediction for the short-term horizon. The results support our previous analysis in Challenge 1 in Sec. III.

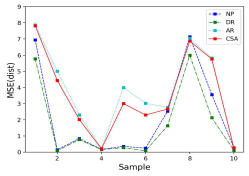


Fig. 5: MSE for the relative distance

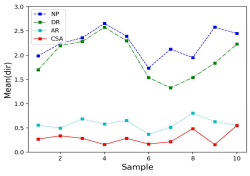


Fig. 6: Mean error for relative direction

Next, we investigate the potential of CSA on the correction of the relative distance and direction errors. As shown in Figure 5, CSA has no advantage in reducing errors of the relative distances. However, it show the best performance in reducing the relative direction error as shown in Figure 6. Considering the relative direction is missing for all types of errors, the result supports our choice of CSA.

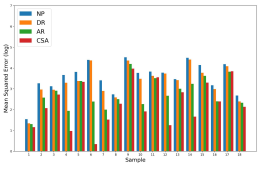


Fig. 7: Performance Comparison

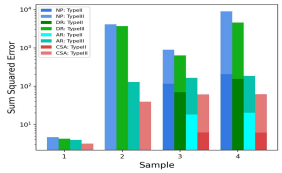


Fig. 8: Comparison in Reducing Different Types of Errors

Lastly, Figure 8 shows the Sum Squared Error (SSE) distribution of two error types for four users. In particular, for the first two cases, only Type III error exists. For the last two case, Type II and Type III errors coexist. And the logarithmic scale of the y-axis is employed. Given that the immediate cause of the persistent error is the presence of obstacles between peer devices in MuAR, and the presence of obstacles, is hard to be avoided, the result demonstrates CSA is the best solution in similar situations.

V. Conclusion

In this work, to fix the tracking unreliability of the current practice on legacy mobile devices in MuAR, we have proposed an approach named CSA, which jointly uses position reports from both single user and multiuser AR experiences, to accomplish multiple coordinate systems alignment and correct erroneous position reported by NI. The evaluation results show CSA's capability of accurate peer tracking in MuAR.

VI. Acknowledgment

We appreciate the constructive comments from the reviewers. This work is supported in part by the NSF grant CNS-2007153 and gift funding from Adobe Research.

References

[1] Y. Siriwardhana, P. Porambage, M. Liyanage, and M. Ylianttila, "A survey on mobile augmented reality with 5g mobile edge computing: architectures, applications, and technical aspects," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 2, pp. 1160–1192, 2021.

[2] Apple. (2018) Swiftshot. [Online]. Available: https://developer.apple.com/documentation/arkit/swiftshot

[3] J. Luksas, K. Quinn, J. L. Gabbard, M. Hasan, J. He, N. Surana, M. Tabbarah, and N. K. Teckchandani, "Search and rescue ar visualization environment (save): Designing an ar application for use with search and rescue personnel," in *2022 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*. IEEE, 2022, pp. 488–492.

[4] Apple. (2017) Arkit. [Online]. Available: https://developer.apple.com/documentation/arkit/

[5] Google. (2018) Arcore. [Online]. Available: https://developers.google.com/ar

[6] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "Orb-slam: a versatile and accurate monocular slam system," *IEEE transactions on robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.

[7] Apple. (2017) Arconfiguration.worldalignment.gravity. [Online]. Available: https://developer.apple.com/documentation/arkit/arconfiguration

[8] F. Palumbo, P. Barsocchi, S. Chessa, and J. C. Augusto, "A stigmergic approach to indoor localization using bluetooth low energy beacons," in *2015 12th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*. IEEE, 2015, pp. 1–6.

[9] M. Scherhäufl, M. Pichler, and A. Stelzer, "Uhf rfid localization based on evaluation of backscattered tag signals," *IEEE Transactions on Instrumentation and Measurement*, vol. 64, no. 11, pp. 2889–2899, 2015.

[10] D. Vasisht, S. Kumar, and D. Katabi, "{Decimeter-Level} localization with a single {WiFi} access point," in *13th USENIX Symposium on Networked Systems Design and Implementation (NSDI 16)*, 2016, pp. 165–178.

[11] I. Oppermann, M. Hämäläinen, and J. Iinatti, *UWB: theory and applications*. John Wiley & Sons, 2004.

[12] N. Wang, H. Wang, S. Petrangeli, V. Swaminathan, F. Li, and S. Chen, "A reality check of positioning in multiuser mobile augmented reality: Measurement and analysis," in *ACM Multimedia Asia*, 2022, pp. 1–5.

[13] Apple. (2020) Nearby interaction. [Online]. Available: https://developer.apple.com/documentation/nearbyinteraction

[14] S. Petrangeli, G. Simon, and V. Swaminathan, "Trajectory-based viewport prediction for 360-degree virtual reality videos," in *2018 IEEE International Conference on Artificial Intelligence and Virtual Reality (AIVR)*. IEEE, 2018, pp. 157–160.

[15] N. Wang, H. Wang, S. Petrangeli, V. Swaminathan, F. Li, and S. Chen, "Towards field-of-view prediction for augmented reality applications on mobile devices," in *Proceedings of the 12th ACM International Workshop on Immersive Mixed and Virtual Environment Systems*, 2020, pp. 13–18.

[16] K. A. Dowsland and J. Thompson, "Simulated annealing," *Handbook of natural computing*, pp. 1623–1655, 2012.

[17] A. Mavlankar and B. Girod, "Video streaming with interactive pan/tilt/zoom," in *High-Quality Visual Experience*. Springer, 2010, pp. 431–455.