




MotionTrace: IMU-based Trajectory Prediction for Smartphone AR Interactions

Rahul Islam 
Stevens Institute of Technology
Hoboken, USA

Vasco Miguel Liang Xu 
University of Chicago
Chicago, USA

Karan Ahuja 
Northwestern University
Evanston, USA

Abstract—Smartphone powered mobile Augmented Reality (AR) applications require precise and accurate 6DoF localization of the phone within a 3D space. This typically requires camera-based localization systems that are power intensive and prone to motion blurs, thereby limiting their usage. In such cases, IMU-based methods are a promising alternative for low power localization sensing. In this paper, we present MotionTrace, a method for predicting a user’s handheld phone trajectory using a smartphone’s inertial sensor. We evaluated MotionTrace over future hand positions at 50, 100, 200, 400, and 800ms time horizons using the large motion capture (AMASS) and smartphone-based full-body pose estimation (Pose-on-the-Go) datasets. Our results show that MotionTrace can estimate the future phone position of the user with an average MSE between 0.11 - 143.62 mm across different time horizons.

I. INTRODUCTION

Augmented Reality (AR) blends digital elements with the real world, creating interactive experiences in sectors like healthcare, education, and entertainment. With tools such as Apple’s ARKit and Google’s ARCore, AR technology is readily available on everyday devices like smartphones and tablets. By 2024, AR has reached a billion users. AR experiences enhance the user’s field of view (FOV) by integrating digital content, enabling interaction with digital elements overlaid on physical surroundings. However, the widespread adoption of AR is hindered by the need for high bandwidth and continuous tracking for realistic experiences, which requires trajectory prediction for FOV. High-quality AR experiences require significant data transfers, including complex meshes and textures, to make digital objects appear realistic within the user’s FOV. This results in extensive rendering and possible delays, leading to subpar user experiences due to prolonged loading times before an AR experience can start.

Field-of-View (FOV)-dependent streaming, initially developed for 360-degree video platforms, optimizes video delivery by adapting the streaming quality to the user’s current and projected FOV, significantly reducing startup latency and data transmission volume [1], [2]. This concept has been extended to augmented reality (AR) applications with trajectory predictions, where techniques selectively enhance the resolution and detail of AR content within or likely to fall within the user’s immediate view, improving performance and user experience [3]. However, AR presents unique challenges due to its environments’ complex and dynamic nature, where multiple objects often occupy the user’s FOV simultaneously,

necessitating a more comprehensive approach to FOV with trajectory prediction and rendering immediately relevant AR content with higher quality. Continuous camera-based tracking in AR can degrade smartphone performance due to high power consumption and processing demands, potentially leading to overheating and reduced battery life, which in turn negatively impacts user experience by limiting the device’s operational duration and responsiveness. Despite initial successes, the scope of trajectory prediction in AR remains under-explored, offering avenues for significant improvements and innovations.

FOV prediction in AR, distinguished from 360-degree video streaming by AR’s support for six degrees of freedom (6DOF) with real-time trajectory information of user and interactions with digitally superimposed objects on the real world, remains an unresolved problem requiring tailored solutions [1], [4], [5]. We propose MotionTrace, a novel approach that utilizes the readily accessible Inertial Measurement Unit (IMU) data from smartphones—leveraging historical data on hand position combined with orientation and acceleration—to accurately predict a user’s trajectory in AR, focusing solely on translation across three degrees of freedom. This method not only addresses the complexity added by AR’s interactive digital objects and user movements but also highlights the superiority of using IMU over camera sensors. IMU-based tracking is less resource-intensive, making it more suitable for continuous operation without substantial power drain, thus ideal for long-term applications where battery conservation is critical. Additionally, the IMU proves reliable where camera-based systems falter, such as in poorly lit or visually occluded environments. This makes IMUs exceptionally beneficial for persistent sensing in diverse conditions. Our extensive evaluation of this method uses large motion capture (AMASS) and smartphone-based full-body pose estimation (Pose-on-the-Go) datasets, effectively predicting future hand positions at intervals up to 800ms at 30fps, showcasing the practical applicability and advantages of IMU data in dynamic, interactive AR settings [5].

II. RELATED WORK

A. Trajectory Prediction in Augmented Reality

In bandwidth-demanding AR applications, several methods have been developed to optimize the streaming of AR content, enhancing user experience and reducing bandwidth. Noh et al. [6] and Park et al. [7] describe cloud-assisted systems and

3D tiling techniques, respectively, that select optimal levels of detail and tiles based on bandwidth and user proximity. Crucial to these technologies is the accurate prediction of a user’s trajectory, which is enhanced by algorithms like the Trace Match & Merge [5], using historical AR data to predict future trajectory, and the ACE Dataset approach [3], which analyzes user movements and digital object locations to predict user focus areas. These predictive models are essential for maintaining high visual fidelity and surpass traditional trajectory prediction methods such as dead reckoning and linear regression [1].

By predicting which parts of a scene a user is likely to focus on, these systems can pre-load high-fidelity graphics in those areas, thereby reducing latency and enhancing the overall user experience. Such techniques underscore the importance of predictive accuracy in the development of advanced AR platforms, providing a direct link to the necessity of our research in improving trajectory prediction through MotionTrace.

B. Inertial Sensors in Augmented and Virtual Reality

Inertial sensors are pivotal in augmenting user interaction within AR and VR technologies through enhanced movement tracking and prediction capabilities. The HOOV system demonstrates this by using wrist-worn inertial sensors to track hand positions outside the user’s visual field, improving interaction with virtual objects and spatial awareness [8]. This concept is further developed in studies like [9] and [10], where inertial sensors facilitate real-time human arm movement prediction for effective human-robot collaboration and enhance 3D hand trajectory forecasting in VR by integrating inertial data with visual inputs, respectively. Extending these applications, [11], and [12] explore the use of inertial sensors in generating collision-free robot trajectories, predicting complex upper limb movements, and improving gesture recognition algorithms in VR, significantly enhancing both safety and efficiency in human-robot interactions.

Moreover, [13] explores the continuous 3D hand trajectory prediction in VR, highlighting the potential of kinematics-based models to predict user interactions with virtual objects, thus preventing collisions and enhancing user experience. Finally, [5] discuss the use of motion tracking and prediction in enhancing the streaming of content in AR applications, where accurate prediction of the user’s field of view can significantly optimize bandwidth usage and reduce latency.

III. IMPLEMENTATION

Our focus is on the inertial sensor present in the smartphone. We operate under the assumption that IMU data from the smartphone is always available. Utilizing historical IMU data, we predict the future position of the hand holding the smartphone. The IMU is known to consume less power than the camera sensor, making it more suitable for continuous sensing without causing significant resource consumption on the device.

A. Model

For the learning architecture, we use a two-layer Bidirectional LSTM with exogenous input, inspired by prior works [14]. For the available IMU, our system uses historical data of orientation (represented as a quaternion) as well as acceleration as input, both in a global coordinate frame of reference. We then flatten the historical data of hand positions concatenated with historical IMU data of n sequence length to create an model input vector of size $n \times 10$: 3 historical hand position, 4 orientation, and 3 acceleration. We then create an exogenous input (dimension=7) to model of orientation and an acceleration at timestep $n + 1$, as we assumed IMU data is always available. With these input model predict future hand position at time step $n + 1$.

The LSTM layer is central to handling the sequential data, with its bidirectional configuration allowing the model to learn dependencies from both forward and backward sequences. This LSTM layer consists of 2 layers and a hidden dimension of 256 for each direction. Thus, the bidirectional setup effectively doubles the LSTM output features to 512 per sequence. After processing through the LSTM, the output at the last time step, representing the most recent and relevant features from both directions of the sequence, is concatenated with the exogenous sensor data, resulting in a combined input vector of 519 dimensions (512 from LSTM and 7 from the exogenous data). This combined data is first passed through a fully connected layer with a dimension transformation from 519 to 256, integrating the features using a ReLU activation function for non-linear processing. A dropout layer with a rate of 0.2 follows to prevent overfitting by randomly dropping units during the training phase. Finally, the output is passed through another fully connected layer, which reduces the dimension from 256 to 3, corresponding to the three dimensions of the hand position.

B. IMU Dataset Synthesis

We required a significant volume of data to train our future hand position model. We employ the CMU [15], BMLrub [16], and HDM05 [17] subsets from the AMASS [18] dataset for the training and testing of our hand position model. The AMASS dataset aggregates various optical marker-based MoCap datasets and standardizes them into 3D human meshes through the SMPL [19] model parameters, creating a comprehensive human motion database. It’s important to note that AMASS has been used in several prior studies [14], [20] as the foundation for creating synthetic datasets.

We utilize the synthetic data creation method as presented in TransPose [14] and DIP [20]. Essentially, we affix virtual IMUs to particular vertices in the SMPL mesh at the right wrist and generate synthetic acceleration data from neighboring frames in the global reference frame. For the creation of synthetic orientation data, we compute joint rotations in relation to the global frame by compounding local rotations from the joint towards the pelvis (root), adhering to the SMPL kinematic chain.

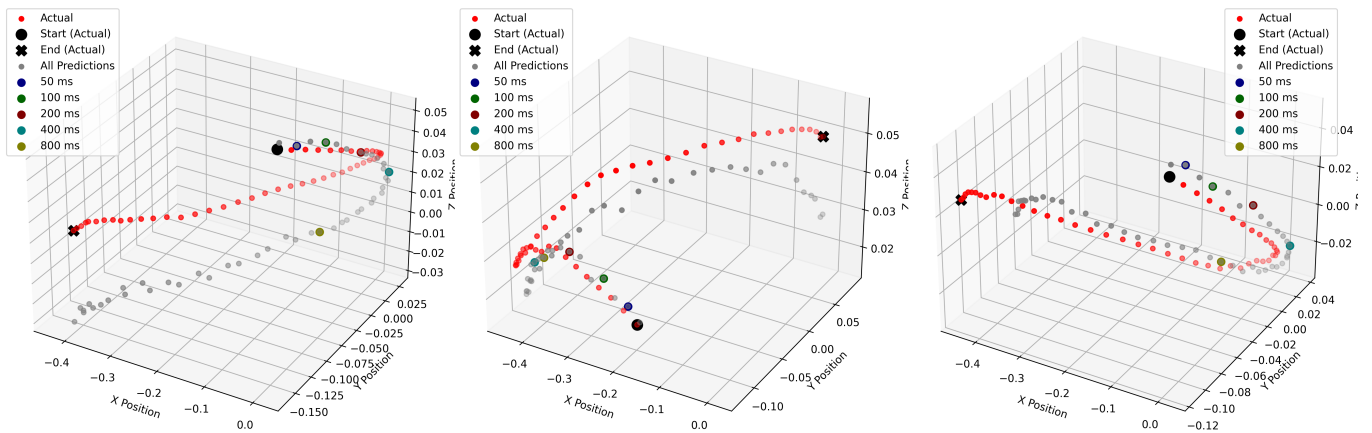
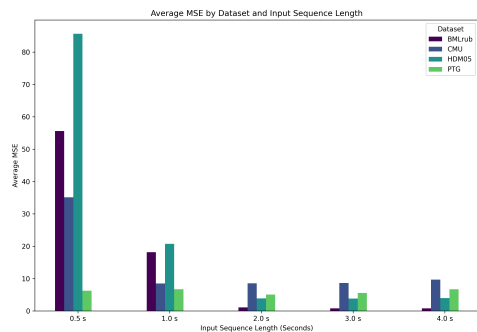
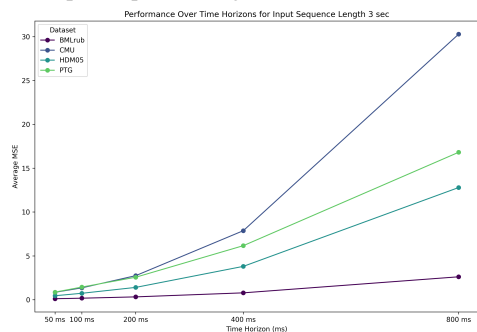


Fig. 1: Samples of predictions by our model at 50, 100, 200, 400, and 800 ms.



(a) Average MSE across time horizons by dataset and input sequence length



(b) Performance over time horizons for input sequence length 3 sec

Fig. 2: Comparison of average MAE across different datasets and time horizons.

C. Training

The model is trained end-to-end using the PyTorch and PyTorch Lightning deep learning frameworks. The batch size is set to 64 during training, and the Adam optimizer is utilized to update the weights with a learning rate of 0.00001. This process is guided by a learning rate scheduler set to plateau. Training uses non-overlapping 3-second windows (or 180 sequence length) of paired IMU and translation data. The

model is trained to predict future hand positions using the mean squared error (MSE) loss. The entire training, lasting 100 epochs or approximately 34 hours, is conducted on an NVIDIA Tesla V100.

IV. EVALUATION

A. Dataset

We evaluate our method subset (CMU [15] (9.19hr), BMLrub [16] (1.7hr), and HDM05 [17] (2.4hr)) of AMASS dataset and on smartphone based full body pose estimate data - Pose-on-the-Go (PTG) [21]. It's worth noting that PTG is collected in the real world. Evaluating our method using PTG further demonstrates its applicability in real-world scenarios. We synthesis IMU and hand position translation on AMASS to train and test our model (See Section III-B). Furthermore the PTG dataset already have hand position (smartphone position) and orientation data. We compute the acceleration with the help of hand position translation and timestamp in the dataset for further training and evaluation.

B. Evaluation Results

We evaluated four different datasets in total. To further assess the generalization capability of our proposed method, we conducted a 4-fold cross-dataset evaluation. This involved training on three subsets and testing on the remaining subset in a round-robin fashion. Table I displays the experimental results of various methods tested on the CMU, BMLrub, HDM05, and PTG datasets. To evaluate our model for future hand position prediction, we try non-overlapping 0.5, 1, 2, 3, and 4-second windows of paired IMU and translation data, i.e., n (sequence length) and IMU data at $n + 1$ time step, to predict hand position at $n + 1$ time step, we assume that IMU data is always available. We use the translation predicted in $n + 1$ time step as input to predict future position at $n + 2$, and so on. We report the MSE score for each dataset at a prediction horizon of 50, 100, 200, 400, and 800 ms at 30fps.

Our results (Fig 2) show that all prediction errors stopped reducing after 3 seconds of input data. The errors increase (Table

TABLE I: Results of cross-dataset evaluation for input sequence length 3 sec. The table shows average MSE in mm.

Time Horizon (ms)	CMU	BMLrub	HDM05	PTG
50	0.84	0.11	0.45	5.36
100	1.35	0.17	0.73	9.32
200	2.74	0.32	1.40	19.66
400	7.87	0.78	3.80	52.74
800	30.28	2.61	12.79	143.62

I and Fig 2b) as the prediction time horizon increases across all datasets. This is expected as longer prediction times generally introduce more uncertainty into the estimation process. The increasing trend in error rates at longer time horizons suggests a limit to the predictability of movement using MotionTrace, especially for datasets with complex movement dynamics like CMU and PTG. For effective AR applications, focusing on shorter prediction windows or improving the model’s ability to handle complex movements might be necessary.

V. DISCUSSION AND CONCLUSION

Trajectory prediction is essential for enhancing the interactivity and immersion of smartphone AR experiences. To this end, we propose MotionTrace, a method to continuously estimate a user’s hand position in 3D-space. This enables phone position localization for trajectory prediction using inertial sensors. We evaluated our method on a large motion capture (AMASS) and smartphone-based full body pose estimation (Pose-on-the-Go) dataset. Our method was able to predict future hand positions with an average MSE ranging from 0.11 - 143.62 mm across different datasets for time horizons between 50 - 800 ms. We also found that 3 seconds of historical inertial sensor data is sufficient for making this prediction. However, our results showed that errors increase as the prediction time horizon lengthens, leading to larger errors across all datasets. The best results were found within a prediction time horizon of 50 - 400 ms.

Our method, when used in conjunction with other methods proposed in previous literature [3], [5], [7], [22], can provide incremental utility. We base our work on the premise that inertial sensor data is always available on the phone. It’s also presumed to consume less power than the camera sensor, making it more suitable for continuous sensing without significantly straining the device’s resources. This presents a significant advantage over previous methods [3], [5] that rely on continuous historical streaming to predict the trajectory of FOV.

REFERENCES

[1] Y. Bao, H. Wu, T. Zhang, A. A. Ramli, and X. Liu, “Shooting a moving target: Motion-prediction-based transmission for 360-degree videos,” in *2016 IEEE International Conference on Big Data (Big Data)*. IEEE, 2016, pp. 1161–1170.

[2] L. Sun, F. Duanmu, Y. Liu, Y. Wang, Y. Ye, H. Shi, and D. Dai, “Multi-path multi-tier 360-degree video streaming in 5g networks,” in *Proceedings of the 9th ACM multimedia systems conference*, 2018, pp. 162–173.

[3] 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.

[4] S. Petrangeli, G. Simon, and V. Swaminathan, “Trajectory-based viewport prediction for 360-degree virtual reality videos,” in *2018 IEEE International Conference on AIVR*. IEEE, 2018, pp. 157–160.

[5] A. Viola, S. Sharma, P. Bishnoi, M. Gadelha, S. Petrangeli, H. Wang, and V. Swaminathan, “Trace match & merge: Long-term field-of-view prediction for ar applications,” in *2021 IEEE International Conference on Artificial Intelligence and Virtual Reality (AIVR)*. IEEE, 2021, pp. 1–9.

[6] H. Noh and H. Song, “Cloud-assisted augmented reality streaming service system: Architecture design and implementation,” in *2020 IEEE International Conference on VCIP*. IEEE, 2020, pp. 363–366.

[7] J. Park, P. A. Chou, and J.-N. Hwang, “Volumetric media streaming for augmented reality,” in *2018 IEEE GLOBECOM*. IEEE, 2018, pp. 1–6.

[8] P. Strelji, R. Armani, Y. F. Cheng, and C. Holz, “Hoov: Hand out-of-view tracking for proprioceptive interaction using inertial sensing,” in *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, 2023, pp. 1–16.

[9] N. D. Kahanowich and A. Sintov, “Learning human-arm reaching motion using imu in human-robot collaboration,” *arXiv preprint arXiv:2308.13936*, 2023.

[10] W. Bao, L. Chen, L. Zeng, Z. Li, Y. Xu, J. Yuan, and Y. Kong, “Uncertainty-aware state space transformer for egocentric 3d hand trajectory forecasting,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 13 702–13 711.

[11] Y. Wang, Y. Sheng, J. Wang, and W. Zhang, “Optimal collision-free robot trajectory generation based on time series prediction of human motion,” *IEEE Robotics and Automation Letters*, vol. 3, no. 1, pp. 226–233, 2017.

[12] T. Zhang, Z. Hu, A. Gupta, C.-H. Wu, H. Benko, and T. R. Jonker, “Rids: Implicit detection of a selection gesture using hand motion dynamics during freehand pointing in virtual reality,” in *Proceedings of the 35th Annual ACM Symposium on UIST*, 2022, pp. 1–12.

[13] N. M. Gamage, D. Ishtaweera, M. Weigel, and A. Withana, “So predictable! continuous 3d hand trajectory prediction in virtual reality,” in *The 34th Annual ACM Symposium on UIST*, 2021, pp. 332–343.

[14] X. Yi, Y. Zhou, and F. Xu, “Transpose: Real-time 3d human translation and pose estimation with six inertial sensors,” *ACM Transactions on Graphics (TOG)*, vol. 40, no. 4, pp. 1–13, 2021.

[15] “Mocap cmu,” Carnegie Mellon University - CMU Graphics Lab, 2004. [Online]. Available: <http://mocap.cs.cmu.edu/>

[16] N. F. Troje, “Decomposing biological motion: A framework for analysis and synthesis of human gait patterns,” *Journal of vision*, vol. 2, no. 5, pp. 2–2, 2002.

[17] M. Müller, T. Röder, M. Clausen, B. Eberhardt, B. Krüger, and A. Weber, “Documentation mocap database hdm05,” *Computer Graphics Technical Report CG-2007-2*, Universität Bonn, vol. 7, p. 11, 2007.

[18] N. Mahmood, N. Ghorbani, N. F. Troje, G. Pons-Moll, and M. J. Black, “Amass: Archive of motion capture as surface shapes,” in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 5442–5451.

[19] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black, “Smpl: A skinned multi-person linear model,” in *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*, 2023, pp. 851–866.

[20] Y. Huang, M. Kaufmann, E. Aksan, M. J. Black, O. Hilliges, and G. Pons-Moll, “Deep inertial poser: Learning to reconstruct human pose from sparse inertial measurements in real time,” *ACM Transactions on Graphics (TOG)*, vol. 37, no. 6, pp. 1–15, 2018.

[21] K. Ahuja, S. Mayer, M. Goel, and C. Harrison, “Pose-on-the-go: Approximating user pose with smartphone sensor fusion and inverse kinematics,” in *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, 2021, pp. 1–12.

[22] J. Van Der Hoof, T. Wauters, F. De Turck, C. Timmerer, and H. Hellwagner, “Towards 6dof http adaptive streaming through point cloud compression,” in *Proceedings of the 27th ACM International Conference on Multimedia*, 2019, pp. 2405–2413.