Reading : <https://uploadvr.com/how-vr-tracking-works/>

IMU (Inertial Measurement Unit) : <https://en.wikipedia.org/wiki/Inertial_measurement_unit>

Tracking Implementation : <https://core.ac.uk/download/pdf/326909176.pdf>

<https://x-io.co.uk/gait-tracking-with-x-imu/>

<https://stackoverflow.com/questions/38009496/calculate-the-displacement-of-device-in-unity>

Use Vuforia to track the object while in screen space?

Positional Tracking: <https://xinreality.com/wiki/Positional_tracking>

Integration-Differentiation : <https://www.uml.edu/docs/Integration-Tutorial_tcm18-190090.pdf>

Solution : Intertial + Optical Tracking + Kalman Filter for capturing phone state in AR

Drift : <https://www.researchgate.net/post/How_can_I_avoid_data_drifting_when_integrating_acceleration_signal2>

Reading:

<https://stackoverflow.com/questions/7858759/android-type-linear-acceleration-sensor-what-does-it-show#8006832>

<https://developer.android.com/guide/topics/sensors/sensors_motion>

<https://www.navlab.net/Publications/Introduction_to_Inertial_Navigation_and_Kalman_Filtering.pdf>

<http://www.starlino.com/imu_guide.html>

<https://os.mbed.com/users/aberk/notebook/dead-reckoning/>

<https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.441.7962&rep=rep1&type=pdf>

<https://zju3dv.github.io/rnin-vio.github.io/>

AR CORE / AR KIT for automated device tracking using AR

Image Tracking: In general not available in Hololens without VisualLib (price), Vuforia (basic plan allows).

We can do it ourselves with Tensorflow (?)

API for Tracking and Availability:

<https://docs.unity3d.com/Packages/com.unity.xr.arfoundation@4.1/manual/index.html>

Both of these approaches fail when no visual clarity is present. RNIN-VIO seems to be a really good alternative. Presented here:

<https://zju3dv.github.io/rnin-vio.github.io/>

(Wish I could implement it…)

Or HybVIO:

<https://github.com/SpectacularAI/HybVIO>