Tracking & Calibration Case Studies from AR/VR

Dr Pradipta Biswas, PhD (Cantab)
Assistant Professor
Indian Institute of Science
https://cambum.net/

Contents

TRACKING

- Concept
- Different tracking technologies
- Examples of tracking
  - AR Marker trackinng
  - Optitrack
  - Xsens IMU
  - HTC Vive Pro Eye
  - Oculus Rift
- Case Studies on comparing tracking technologies
  - Object tracking
  - Head tracking

CALIBRATION

Concept

Examples of Calibration

- LR based calibration for HRI
- NN based calibration for Auto UI
- NN based calibration for HMDS

Tracking

Concept

Tracking Technology

Measurement Principle

- Signal Strength
- Signal Direction
- Time of Flight

Geometric Property

- Trilateration
- Triangulation

Sensor Synchronization

Signal sources

- Active (transmitter / reflector)
- Passive

DoF

- 3 for position
- 3 for orientation

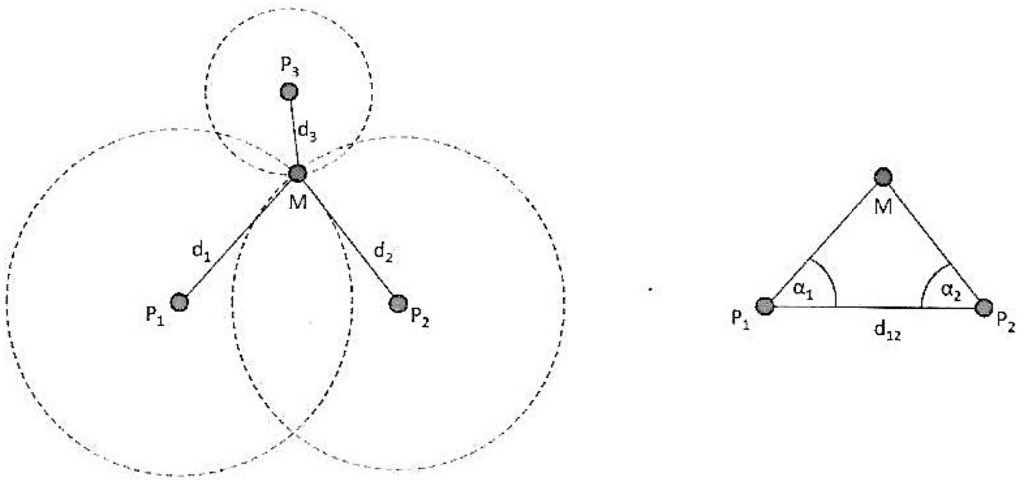


Figure 3.4 (left) Trilateration computes the position of a point M from the known distances d_1, d_2, d_3 to the points P_1, P_2, P_3 by intersection of three spheres. (right) Triangulation determines the position of a point M from two angles α_1 at P_1 and α_2 at P_2 and one known distance d_{12} between P_1 and P_2.

Different Tracking Technologies

Imaging Based

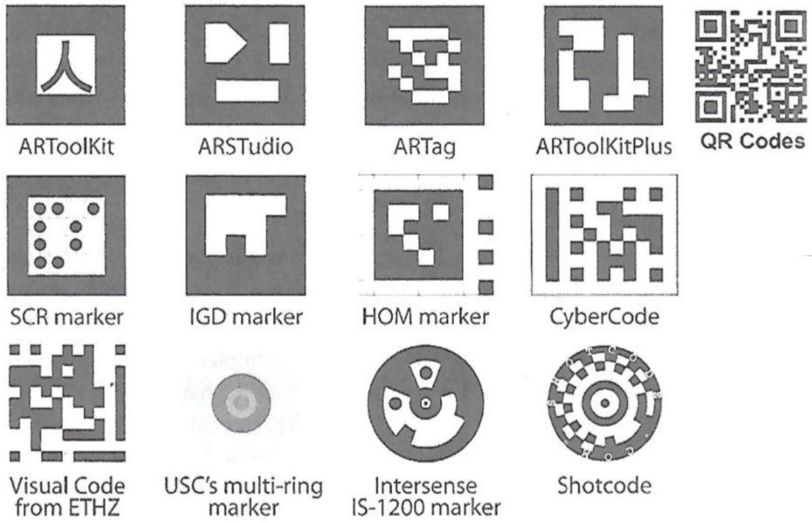
- Frequency of light
- Presence of markers

Inertial Measurement Units

Ultrasound

AR marker tracking

- Circle and Square shapes are easier to detect
- Circle maps to one point, so needs to be grouped
- All corners of square shapes need to be detected
- Patterns inside fiducial markers give orientation and scaling information
- AR renders virtual object after pose estimation of marker



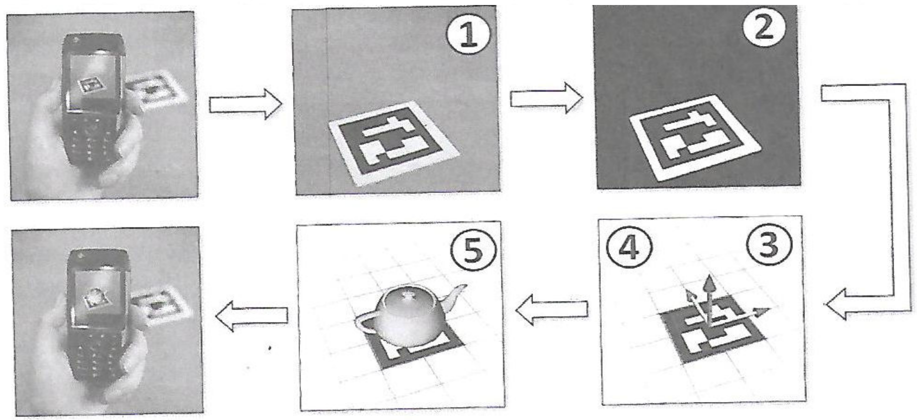
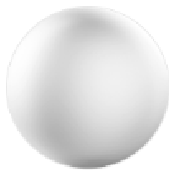


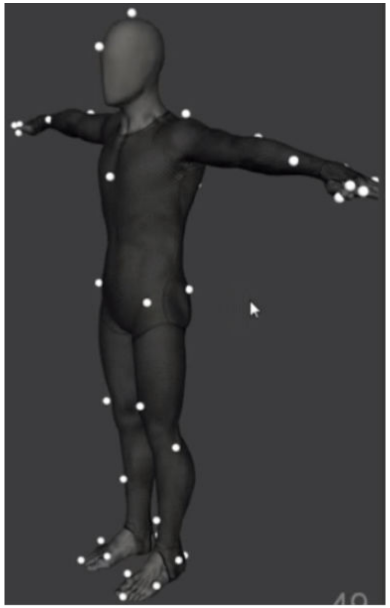
Figure 4.1 The tracking pipeline for square fiducial markers begins with thresholding the image, followed by quadrilateral fitting and pose estimation. With the recovered pose, AR rendering can be performed. Courtesy of Daniel Wagner.

OptiTrack – Motion Capture System

- IR Camera (850 nm, 120 FPS) – Emits and captures reflections
- 3M 7610 Retroreflective IR Marker – For effective IR reflection
- Precise positioning of markers - On the object to track
- Multiple Cameras (8/12/16) can be used to cover the tracking volume
- Unreflective / Black background – Desirable
- Size of marker – Resolution of camera – Tracking Volume – are important



Markers



OptiTrack Motion Capture Systems

Calibration - Tracking volume

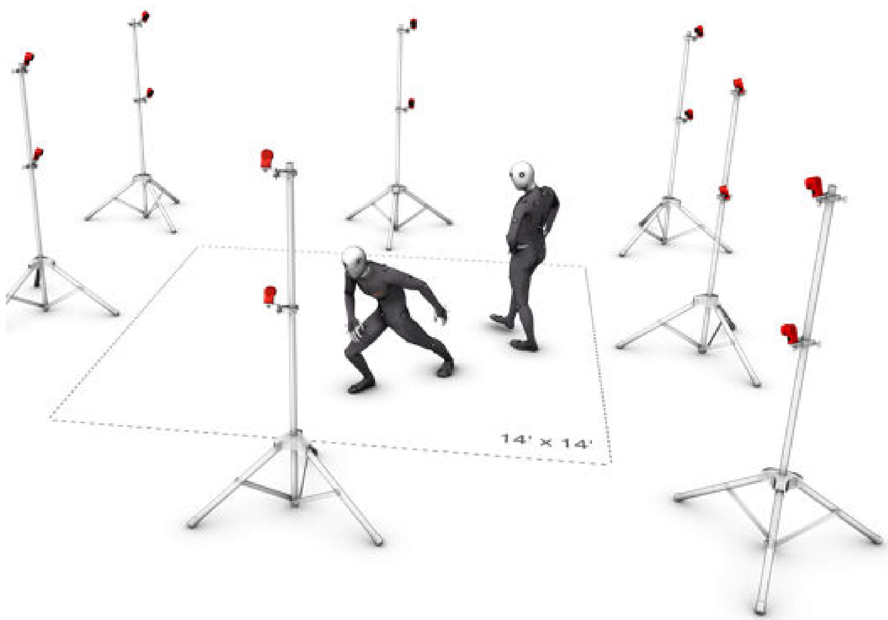
Advanced Image processing algorithms on IR images – 3D localization of markers

Position & Orientation of Rigid Bodies in 3D space & Joint angles and positions in a skeleton

Control and accessing motion capture data – Motive Software/Camera SDK

High accuracy and precision can be obtained subjected to calibration results

Application areas – Movement Sciences, Virtual Reality, Robotics, Drone Tracking, Animation (Film Making), Ergonomic Studies



IMU based Motion Tracking

Inertial Measurement Units (IMU) are based on inertia.

Ranges from a few square mm (Micro Electro-Mechanical Systems – MEMS) to 50-70 cm diameter (Ring laser gyroscopes)

Unlike Vision based OptiTrack, IMUs are self contained systems

- No reference position in space

Measures linear and angular motion using

- Accelerometers – Measures acceleration in x, y, z axes – Uses seismic mass and Cantilever Beam
- Gyroscopes – Angular velocity around x, y, z axes – Uses vibrating element and Foucault Pendulum

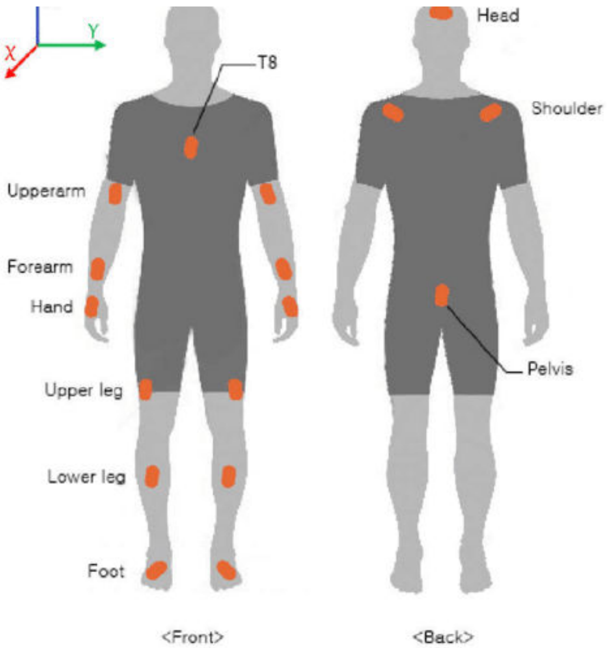
MEMS IMU - Accurate measurements are challenging

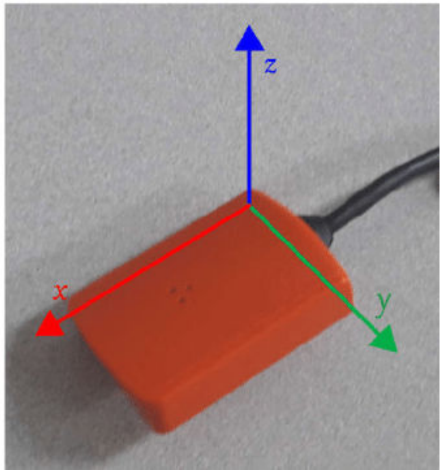
- Double integration of accelerometer data – Position
- Integration of gyroscope data – Orientation
- Both are erroneous due to integration constants (Static and Dynamic Errors)
- Hence, we rely on *Magnetometer* – measures magnetic field in x, y, z axes

Factors to obtain high fidelity data

- High sampling rate, Precise sensors
- Calibration of device
- Signal Processing pipeline with Sensor fusion (Kalman Filtering etc.,)

IMU based Motion Tracking





Common problems in IMU
- Absolute spatial position cannot be obtained
- Drift in position and orientation data

Recent approaches relies on an array of IMUs to overcome this problems

Xsens Motion Tracking relies on placing 17 IMUs on human to perform accurate motion tracking of various joints

Unlimited tracking range – unlike Optitrack

HTC Vive Pro Eye

- **Positional Tracking (6 DoF) – SteamVR ‘Lighthouse’**

Base stations emit synced InfraRed(IR) pulses and laser lines that are then picked up by the headset and controllers.

Headset contains an array of IR photodiodes connected to a chip. This chip measures the time between the IR flash pulse and being hit by the laser sweep for each axis. Thus, position of tracked device is determined.

Specifications	
Screen:	Dual OLED 3.5" diagonal
Resolution:	1440 x 1600 pixels per eye (2880 x 1600 pixels combined)
Refresh rate:	90 Hz
Field of view:	110 degrees
Sensors:	SteamVR Tracking, G-sensor, gyroscope, proximity, IPD sensor, eye tracking

https://enterprise.vive.com/ca/product/vive-pro-eye-office/

I³D

HTC Vive Pro Eye

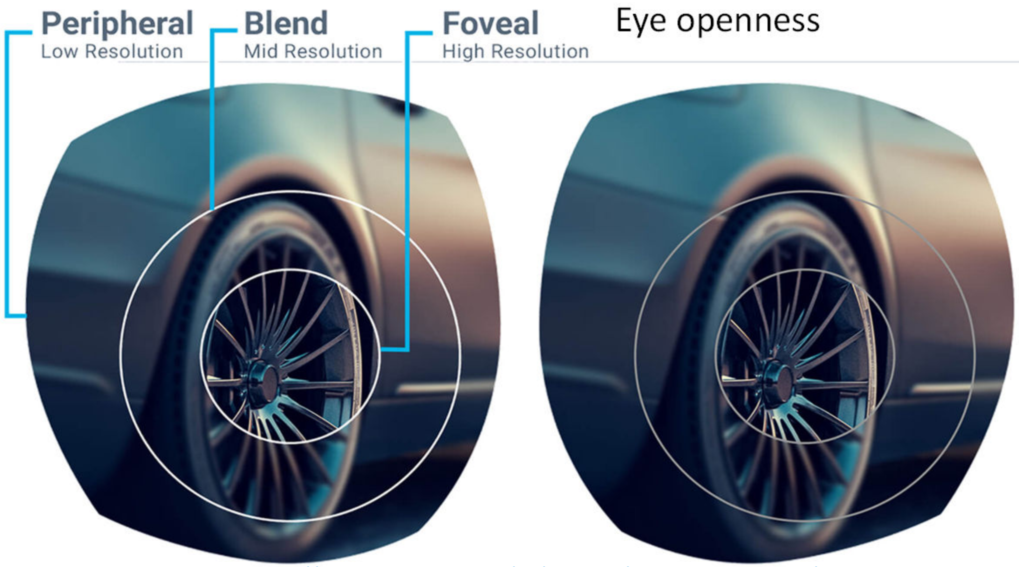
- **Eye Tracking – Tobii Eye Tracking**

Inside of the Head Mounted Display, the eye gaze tracker has illuminator and eye tracking cameras.
The illuminators create a pattern of near infrared light on the eyes. The camera captures high resolution images of the user's eyes. The tracker uses dark pupil tracking with Pupil Center and *Corneal Reflection eye tracking*.

- **Foveated Rendering (NVIDIA Turing based GPUs)**

It is a rendering technique which uses eye-gaze of user to render an area inside VR in higher resolution. This also reduces load on Graphics Processor Unit (GPU) significantly.

Specifications	
Gaze data output frequency (binocular):	120Hz
Accuracy*:	0.5°–1.1°
Calibration:	5-point
Trackable field of view**:	110°
Data output (eye information):	Gaze origin Gaze direction Pupil position Pupil size Eye openness



https://enterprise.vive.com/ca/product/vive-pro-eye-office/

I3D

Oculus Rift

Specifications	
Screen:	Dual OLED
Resolution:	1080x1200 per eye
Refresh rate:	90 Hz
Field of view:	110 degrees
Sensors:	Accelerometer, Gyroscope, Magnetometer, 360-degree positional tracking

Positional Tracking (6 DoF) – 'Constellation'

Headset and Controllers(Oculus Touch) have a pre-defined pattern(constellation) of infrared LEDs.

Base Stations, which are cameras with filters to see only IR light, send images to Computer. Each frame is processed by the computer to identify the position of each IR LED, and thus the relative position of each object.

Constellation

Base Station

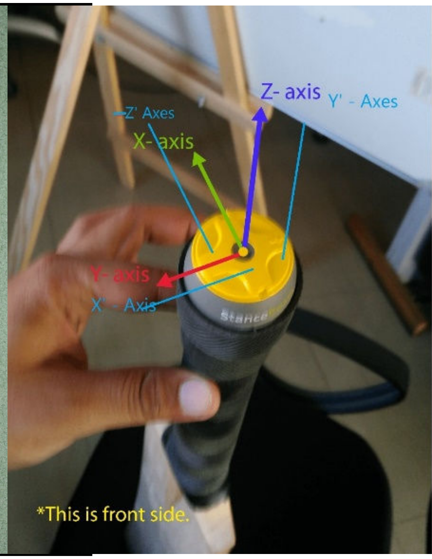
https://assets.fatllama.com/images/medium/oculus-rift-touch-bundle-51613181.jpg

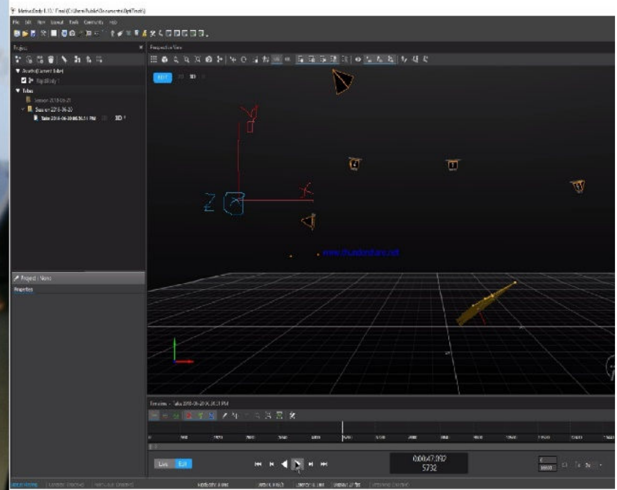
Other Tracking Technologies

- OpenFace Facial Feature Point Tracking (refer lecture on Alternative Input Modalities)
- Microsoft Kinect (refer lecture on Alternative Input Modalities)
- LeapMotion (refer lecture on Alternative Input Modalities)
- Object tracking through CNN (refer lecture on CNN)
- UltraLeap - Ultrasound based tracking (refer lecture on Haptics)

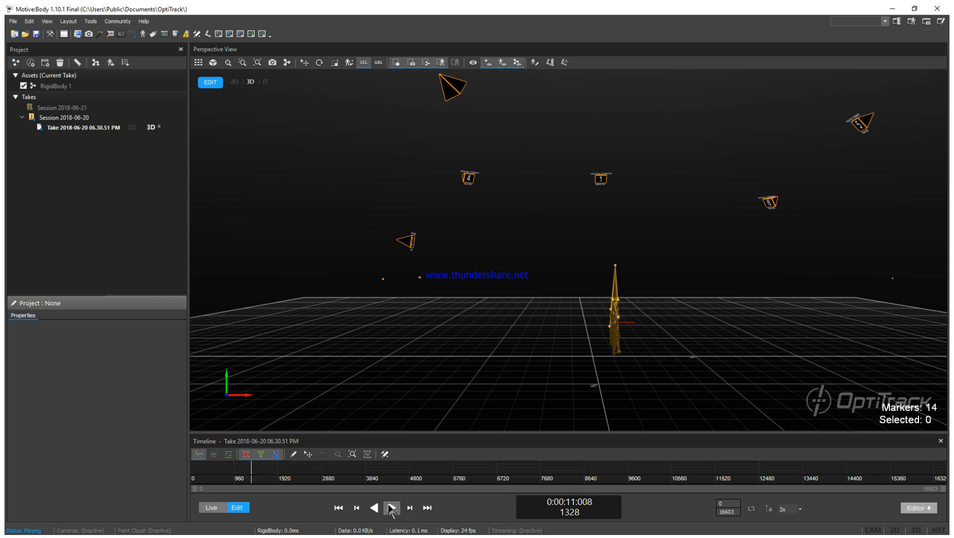
Comparison among different tracking technologies

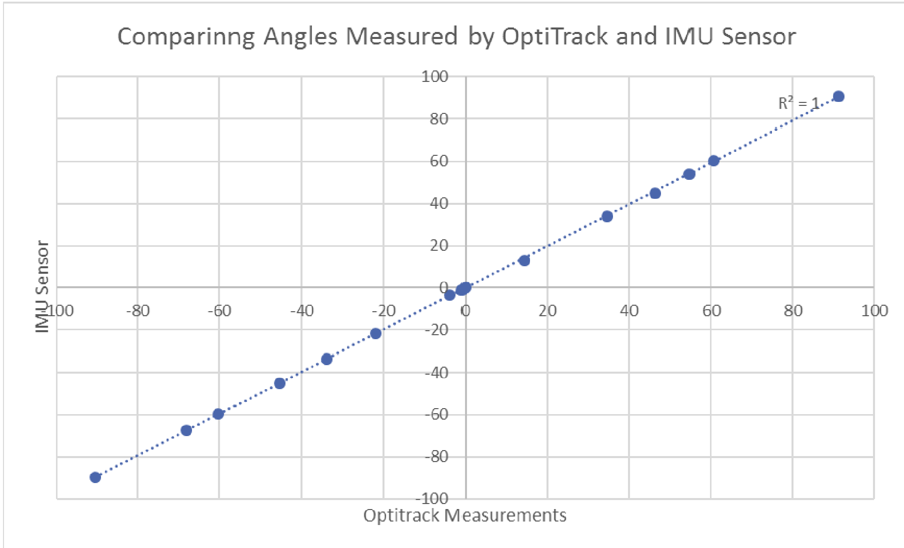
Object Movement



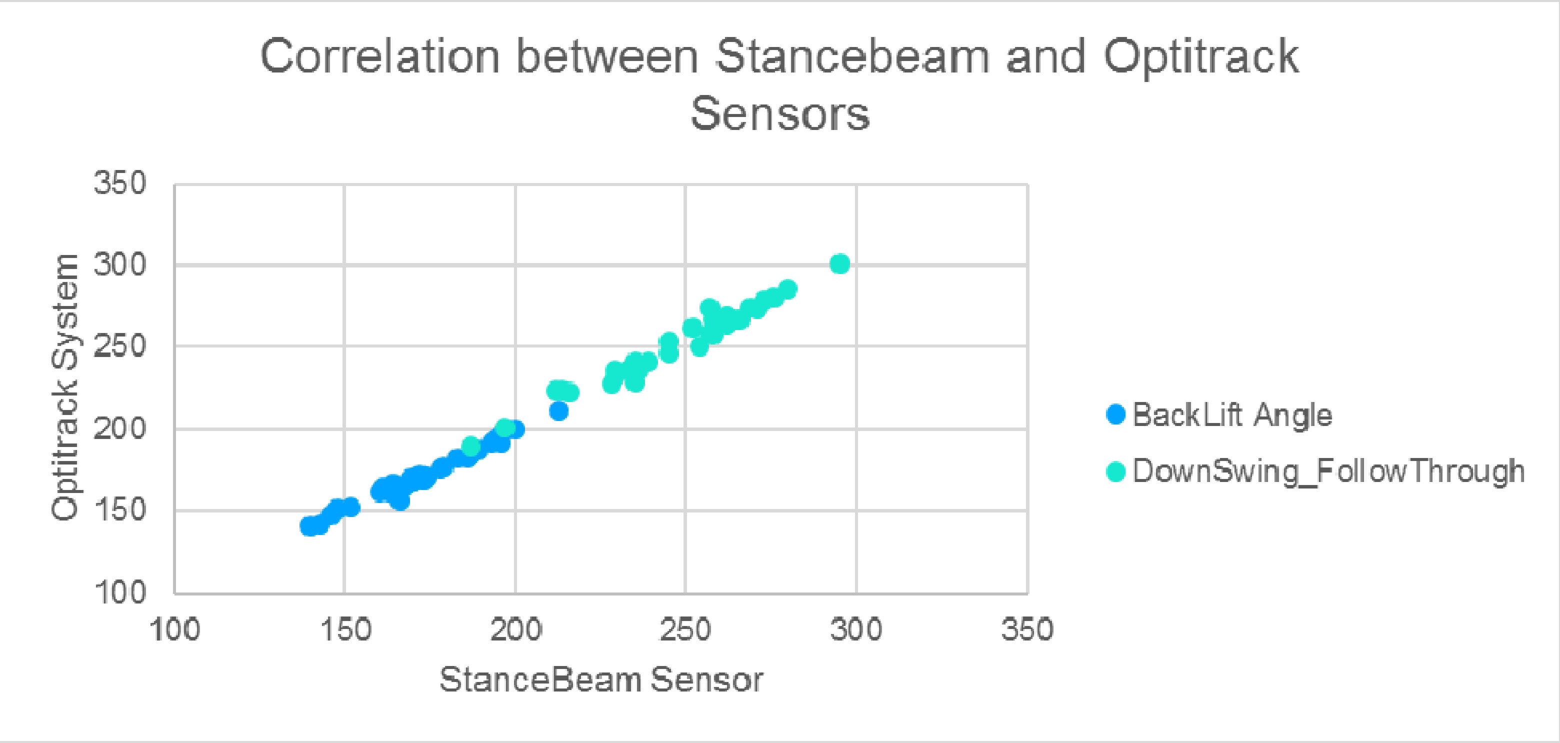


Static Angle Measurement





Correlation Measurement



Head Movement

Since the sampling rate of IMU and OptiTrack is different, we performed time sampling and computed the average value for 1 second interval.

We computed correlation value between IMU and OptiTrack using these time average values.

High correlation between IMU and OptiTrack for both yaw (0.85) and pitch (0.77) measurements

Positive, but low correlation (0.4472) for roll measurements

Calibration

Concept

Camera, Screen and other sensors have different coordinate spaces

Virtual object has to be matched with user input and real object

Mapping between coordinate spaces

Similar to Model or View Transformation used in Computer Vision

Different than camera, sensor or display calibration

Calibration AR Robotics

Measured coordinates of 9 points in the coordinate space of Robot and screen

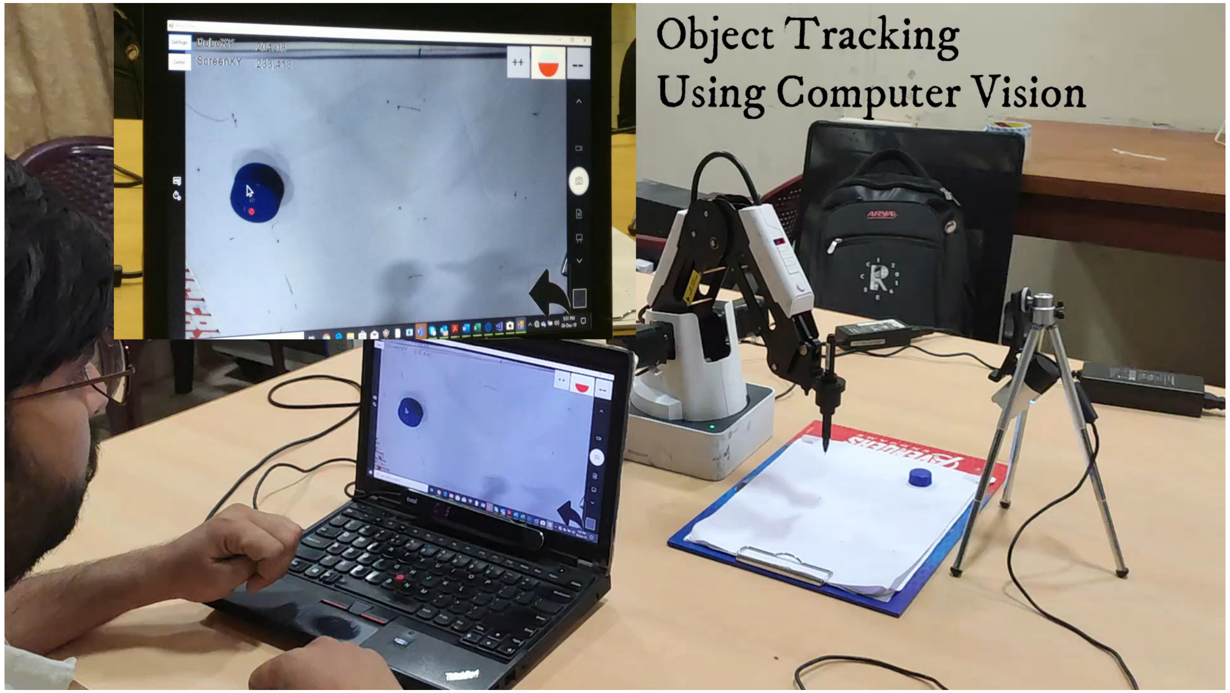
The nine points were distributed throughout the working area of the robot

Used Linear Regression to map screen coordinates to robot coordinates

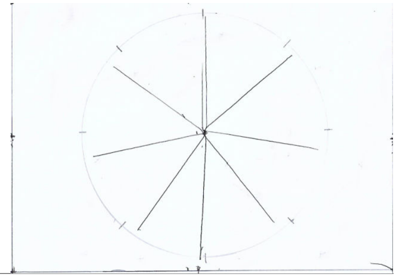
Used correlation and RMS error criteria as goodness of fit

Did not map z-axis

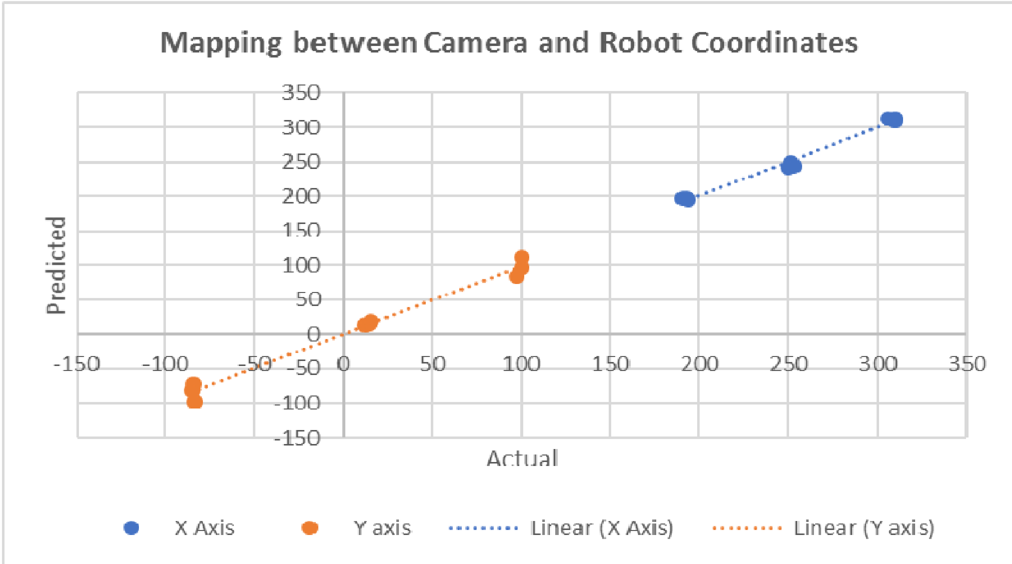
Software takes screen coordinates from user input, converts them to robot coordinate and instruct robot to reach at designated point



I3D



Calibration AR Robotics



Calibration AutoUI

Existing eye trackers are developed for desktop computing environment where

- Tracker is attached below display
- Display is a flat screen

We used eye tracker to track eyes on windshield

Display was away from eye tracker

Display surface was not flat like a computer screen

Calibration - AutoUI

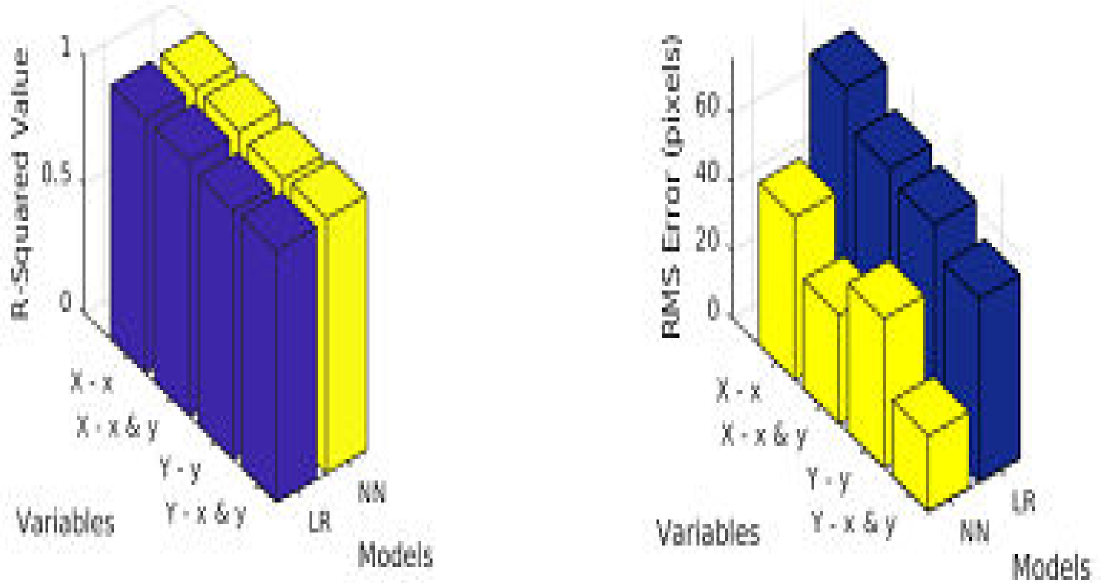
Compared ML systems to convert eye gaze coordinates to screen coordinates on windshield

Set up Linear Regression and Backpropagation Neural Network Models for

- Predicting x-coordinate in screen from x coordinate recorded by gaze tracker
- Predicting x-coordinate in screen from x and y coordinates recorded by gaze tracker
- Predicting y-coordinate in screen from y coordinate recorded by gaze tracker
- Predicting y-coordinate in screen from x and y coordinates recorded by gaze tracker

Compared R^2 and RMS error

Neural Network model worked better than Linear Regression

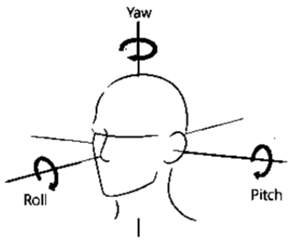


R^2 and RMS error for screen mounted tracker

Combining Head and Eye Gaze Movements

$$T_z(\alpha)=\begin{pmatrix}\cos\alpha & -\sin\alpha & 0\\ \sin\alpha & \cos\alpha & 0\\ 0 & 0 & 1\end{pmatrix} \quad T_y(\beta)=\begin{pmatrix}\cos\beta & 0 & \sin\beta\\ 0 & 1 & 0\\ -\sin\beta & 0 & \cos\beta\end{pmatrix} \quad T_x(\gamma)=\begin{pmatrix}1 & 0 & 0\\ 0 & \cos\gamma & -\sin\gamma\\ 0 & \sin\gamma & \cos\gamma\end{pmatrix}$$

$$T = T_z(\alpha)T_y(\beta)T_x(\gamma)$$



$$eyeL_{hc} = TeyeL \qquad eyeR_{hc} = TeyeR$$

$$eye_{hc} = (eyeL_{hc}\ eyeR_{hc})^T$$

Gaze direction vectors from both eyes (*eyeL & eyeR)*, with their origins at their respective pupil centers are obtained from eye gaze tracker

9-axis IMU to measure yaw (α), pitch (β) and roll (Y) of the user's head

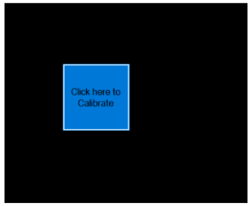
Initial head position is the reference co-ordinate axes and measured head orientation accordingly

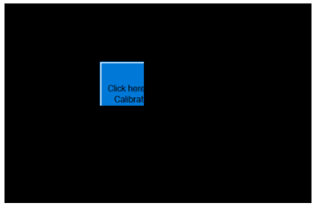
We performed intrinsic 3D transformation [2] for gaze direction vectors to obtain head compensated gaze vectors ($eyeL_{hc}$ & $eyeR_{hc}$)

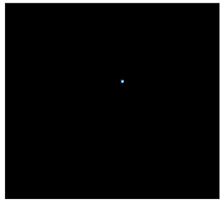
Calibration HMDS

9- Squares appear at designed positions on screen

Attentive Calibration: User is asked to focus on each square ; The size of the square reduces in response to user's focus







User can use either head or/and gaze to focus on squares; Hence a single calibration routine is enough to obtain head and gaze movements

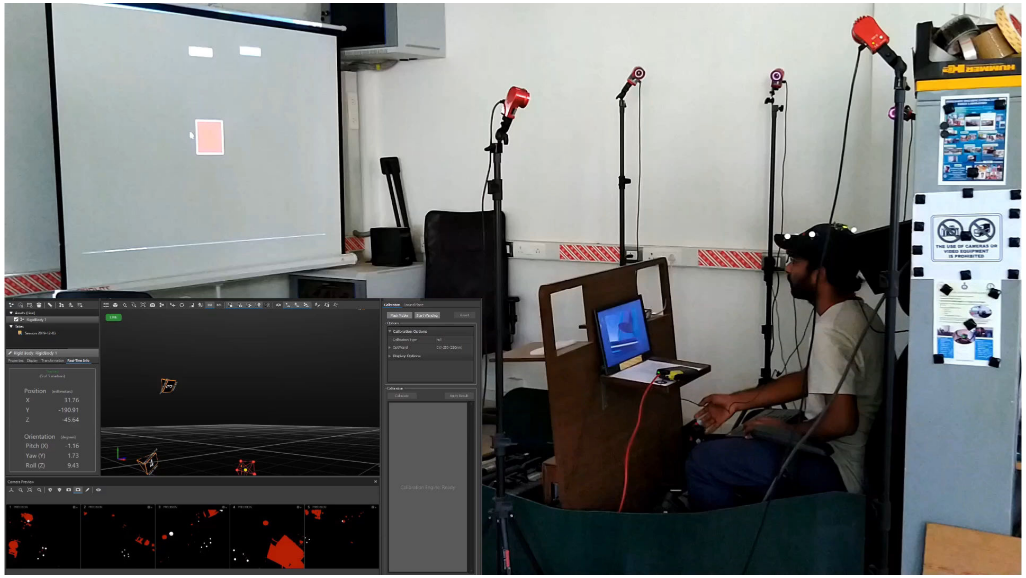
eye_{hc} vectors are collected for each square position at it's minimum size

The mapping function with eye_{hc} as input and corresponding screen co-ordinates as the output is learnt by training a 2 hidden layer neural network

Loss function: *Mean Squared Error*; Optimizer: *Adam*; Libraries: Tensorflow.NET and Keras.NET

I³D

Demonstration



We did not cover

Computer Vision algorithms for tracking

- Needs basic knowledge of Computer Graphics, 3D geometry and computer vision
- Can be described in a separate lecture

AI / ML based tracking algorithms

- Particle filter
- SLAM
- Covered in separate courses

Mobile tracking like GPS

Tracking not commonly used for AR/VR applications like RFID tracking

Take Away Points

- Introducing different tracking technologies
- Case studies on commercial AR/VR tracking systems
- Case studies on comparing tracking accuracy
- Mapping virtual and physical coordinates
- Case studies from Robotics, HUD and HMDS