EECE5698
Networked XR Systems

Lecture Outline for Today

- Sensing and tracking modalities, and their limitations
- RF tracking
- IMU tracking
- Multi-modal Fusion for tracking

Tracking in XR - Recap

- What is Tracking?
  - The process of continuously determining the position and orientation of a user's device or body parts within a given space, such as hands, face, or eyes.

Tracking in XR - Recap

- Why do we need Tracking?
  - Essential for creating an immersive and interactive experience, as it allows the virtual environment to respond dynamically to the user’s movements.
  - E.g., hand tracking in AVP eliminates the need for controllers

Sensing and Tracking Modalities

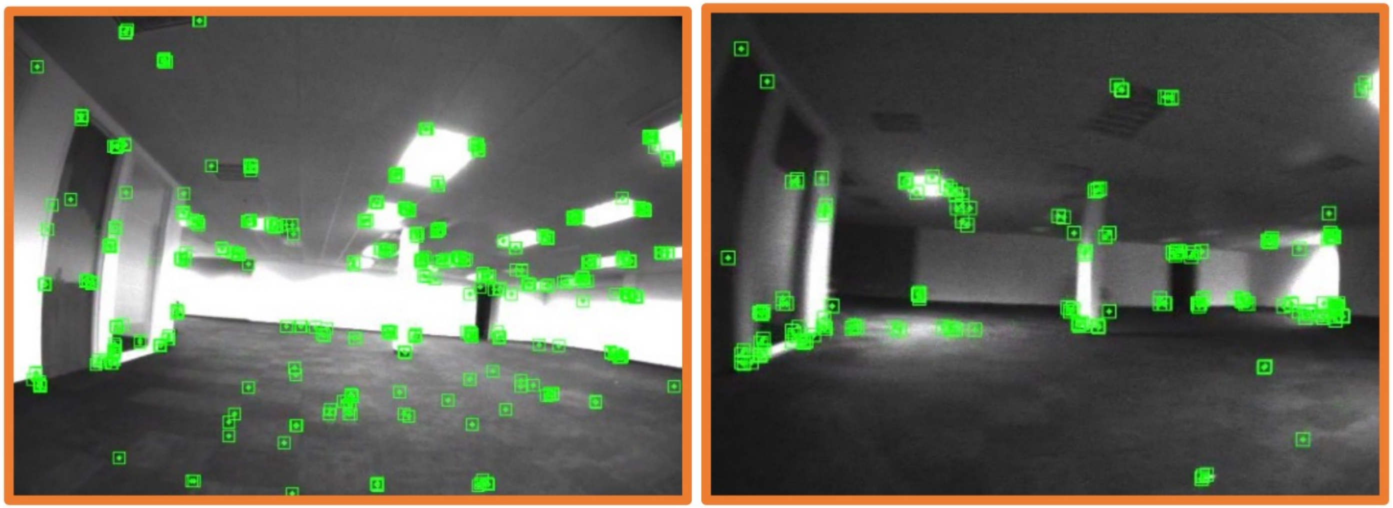
- Cameras
- IMUs
- Lidar
- RF

Sensor and Tracking Idea

- Camera
  - Image Capture: The camera captures sequential images or video frames of the environment or specific objects within it.
  - Feature Detection: Identify and extract specific features from the captured images. These features could be visual markers, predefined shapes, or unique patterns that the system can recognize and track.
  - Motion Analysis: The system analyzes the changes in position and orientation of the detected features across different frames to determine the movement of the camera or the objects within its view.

Sensor and Tracking Idea - Camera

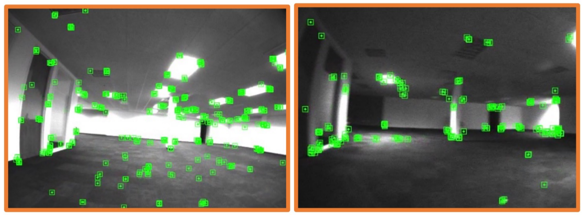
- Step1: Capture images
  - Mono or Stereo or multiple cameras
- Step2: Feature Extraction
  - Features are detected in the first frame, and then matched in the second frame.



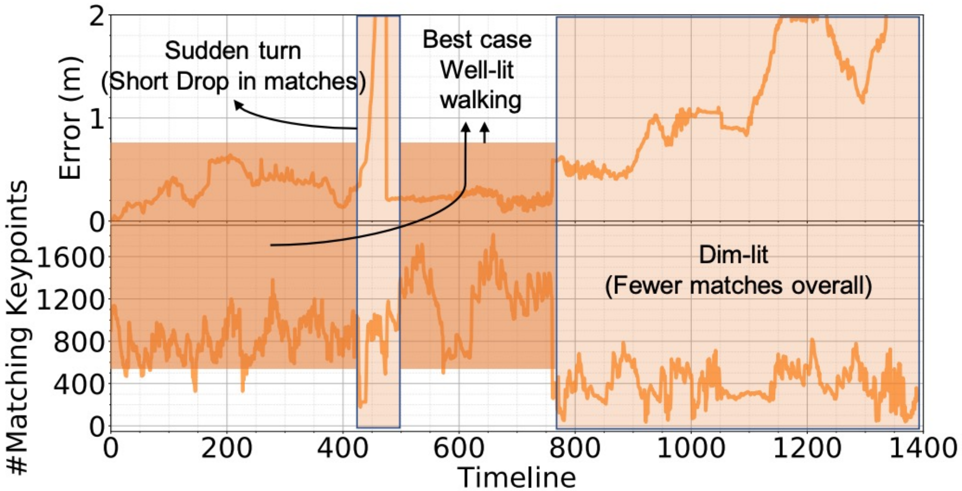
(a) Well-lit (568 matches) (b) Dim-lit (252 matches)

Sensor and Tracking Idea - Camera

- Step1: Capture images
  - Mono or Stereo or multiple cameras
- Step2: Feature Extraction
  - Features are detected in the first frame, and then matched in the second frame



(a) Well-lit (568 matches) (b) Dim-lit (252 matches)



(c) Matches vs. Error

Sensor and Tracking Idea - Camera

- Step3: Optical flow estimation

Get rid of outliers

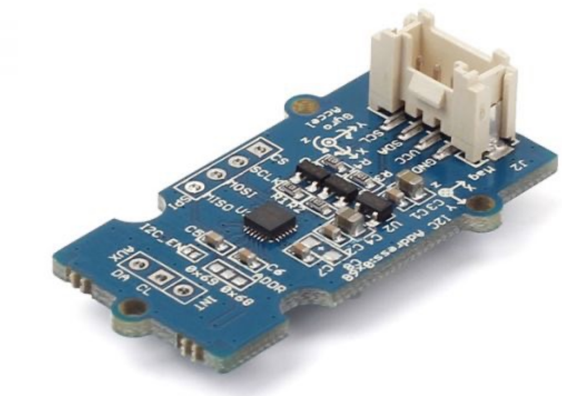
Sensor and Tracking Idea - Camera

- Step4: Estimate camera motion from optical flow

  - The optical flow field illustrates how features diverge from a single point, the *focus of expansion*. The focus of expansion can be detected from the optical flow field, indicating the direction of the motion of the camera, and thus providing an estimate of the camera motion.

Sensor and Tracking Idea

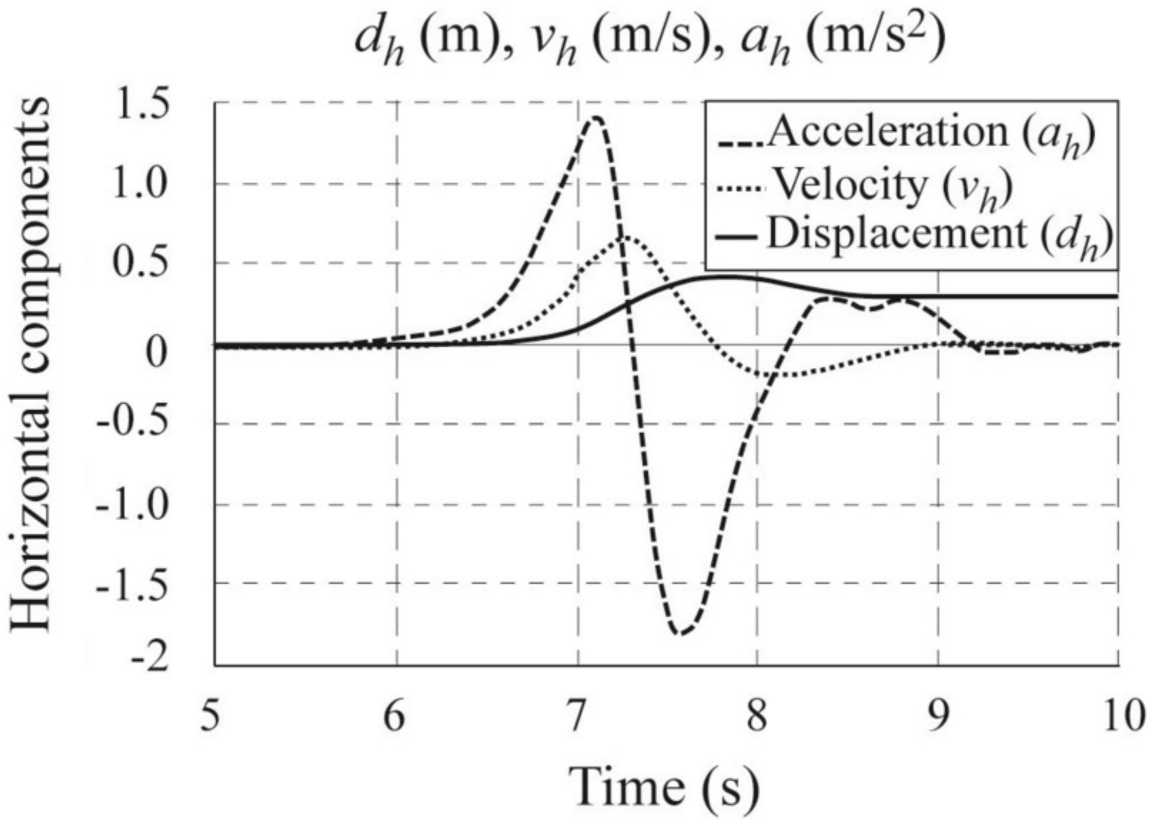
- IMU



Accelerometer, Gyroscope, Magnetometer

Sensor and Tracking Idea

- IMU



Sensor and Tracking Idea

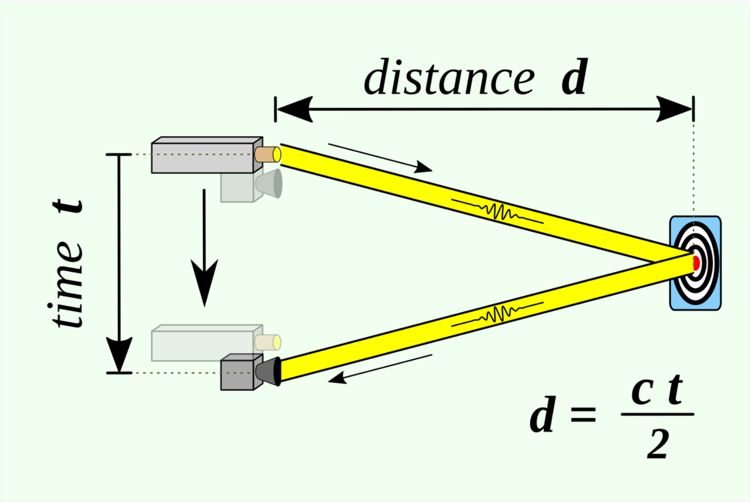
- IMU
  - Calculating velocity and distance from acceleration (a_x, a_y, a_z)

$$v(t) = v(t_0) + \int_{t_0}^{t} a(u)du.$$

$$s(t) = s(t_0) + \int_{t_0}^{t} v(u)du.$$

Sensor and Tracking Idea

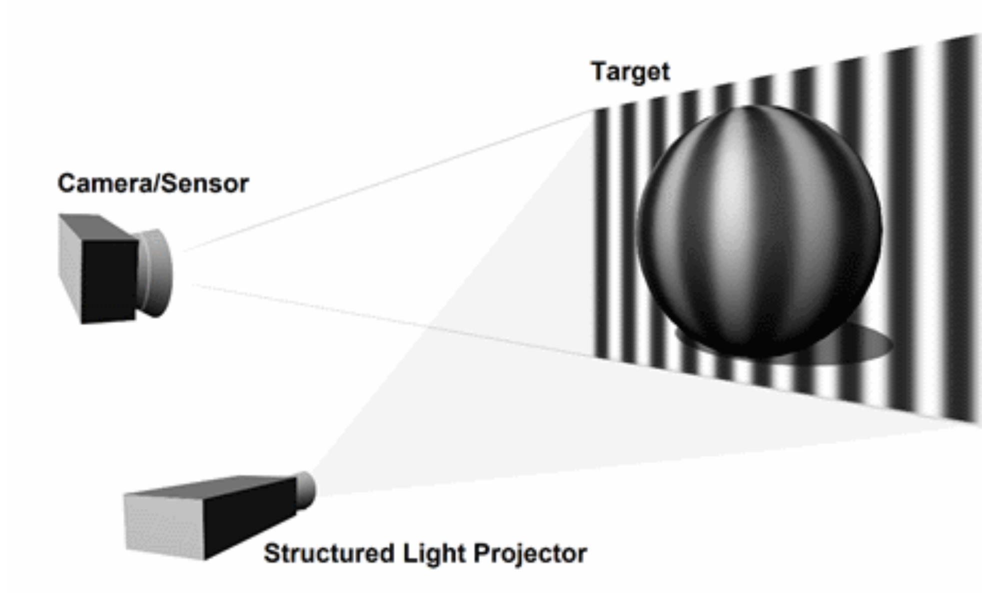
- Lidar



Time of flight

Sensor and Tracking Idea

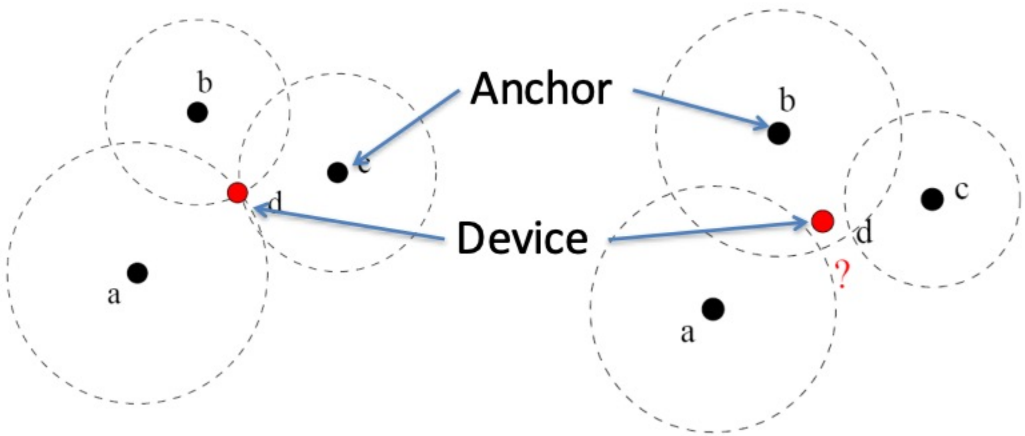
- Lidar



Structured light projection

RF-based Tracking

- Range based tracking
  - Convert received signal strength (RSS) or signal timing to a distance estimate with respect to anchor nodes with known locations.
  - Problem: distance estimates may be erroneous, and the circles may not intersect at a single point.



RF-based Tracking

How to estimate location when the circles do not intersect?

Idea: localize at a point that presents the minimum error to the circles by some reasonable error measure.

k anchors at positions (x_i, y_i)

Assume node to be localized has actual location at (x_0, y_0)

Distance estimate between node 0 and anchor *i* is r_i

Error:

$$f_i = r_i - \sqrt{(x_i - x_0)^2 + (y_i - y_0)^2}$$

RF-based Tracking

Linearization and Min Mean Square Estimate

- Ideally, we would like the error to be 0

$$f_i = r_i - \sqrt{(x_i - x_0)^2 + (y_i - y_0)^2} = 0$$

- Re-arrange:

$$(x_0{}^2 + y_0{}^2) + x_0(-2x_i) + y_0(-2y_i) - r_i^2 = -x_i^2 - y_i^2$$

- Subtract the last equation from the previous ones to get rid of quadratic terms.

$$2x_0(x_k - x_i) + 2y_0(y_k - y_i) = r_i^2 - r_k{}^2 - x_i^2 - y_i^2 + x_k^2 + y_k^2$$

- Note that this is linear.

https://en.wikipedia.org/wiki/Ordinary_least_squares

Tracking Metrics

- Typical metrics of importance
  - Accuracy
  - Latency
  - Tracking drift
  - Tracking jitter
  - Update rate
  - Reliability

Sensor Limitations

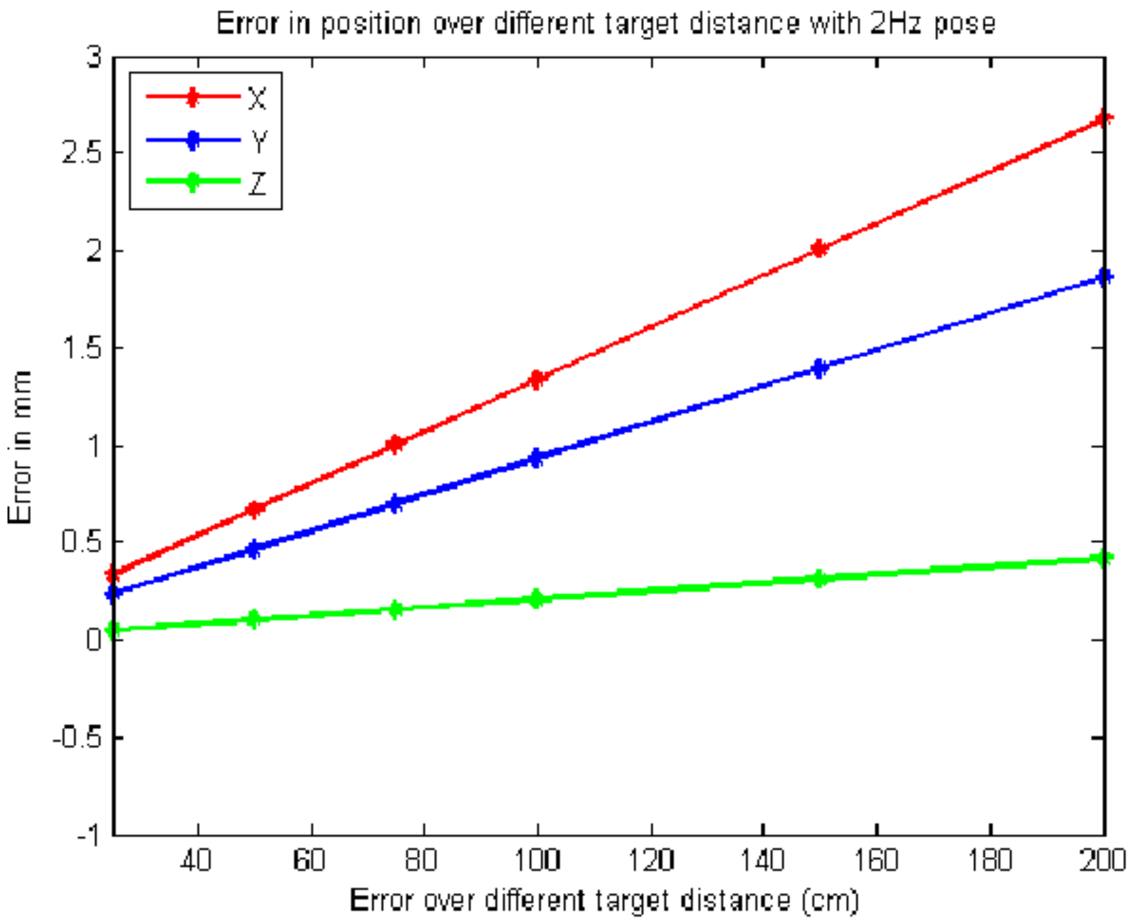
- Camera
  - Heavily depends on the environment
    - Lighting conditions
    - Geometry of the objects in the environment
    - Uniform surfaces or color
    - Moving objects
    - Fails when too close to objects; camera view occluded

Sensor Limitations

- IMU
  - Drift: One of the most significant limitations of IMUs is drift. Over time, small errors in the measurement of acceleration and angular velocity accumulate, leading to increasing errors in calculated positions and orientations. This drift is especially problematic in applications requiring long-term accuracy without external correction signals.
  - Noise: The raw signals from an IMU’s sensors typically contain a significant amount of noise. Effective use of IMU data often requires sophisticated filtering techniques to reduce this noise and improve the accuracy of the derived motion data.
  - Magnetic Interference: Magnetometers are susceptible to magnetic interference from nearby ferrous materials and electromagnetic fields, which can distort the readings.
  - Integration Errors: The process of integrating acceleration to calculate velocity and position introduces cumulative errors, which grow over time. Without external reference points for correction, these errors can lead to significant inaccuracies in position data.

Sensor Limitations

- IMU



Sensor Limitations

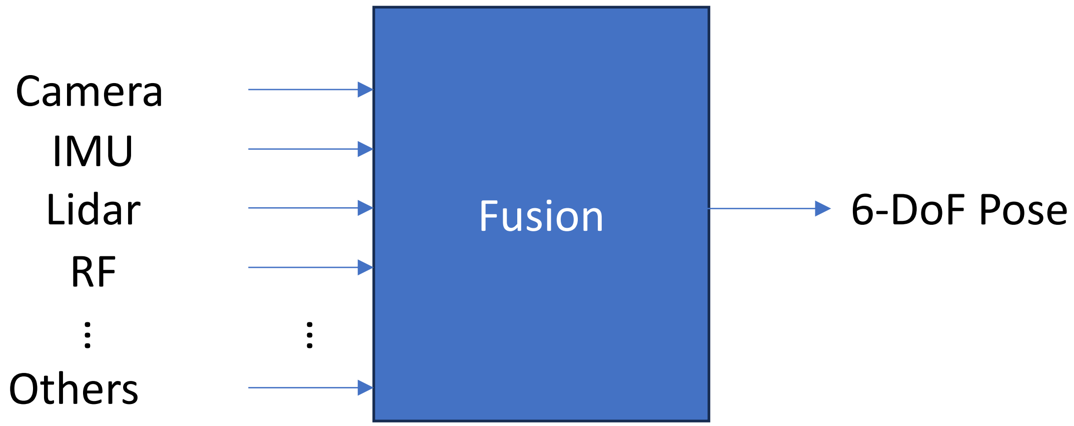
- **Lidar**
  - Range Limitations: The effective range of lidar depends on the power of the laser and the reflectivity of the target surface. While recent advancements have improved range, lidar is still generally less effective at longer distances compared to some radar systems.
  - Limited Performance Under Direct Sunlight: The effectiveness of lidar can be compromised in bright conditions. Sunlight can interfere with the sensor's ability to detect its own laser pulses, particularly when objects are far away.
  - Field of View: Some lidar systems have a limited field of view compared to cameras. This limitation means that lidar may need to be used in conjunction with other types of sensors to provide comprehensive coverage of an environment.
  - Surface Reflectivity Issues: Lidar relies on light bouncing back from surfaces to measure distances. Surfaces that are very dark, or non-reflective, might not return enough light, leading to gaps in data or inaccuracies.

Sensor Limitations

- RF
  - Multipath Propagation: In environments with many reflective surfaces, like metal or glass, RF signals can bounce and create multiple paths that reach the receiver at slightly different times. This phenomenon, known as multipath propagation, can lead to errors in determining the position of an object or user.
  - Line-of-Sight Requirements: While RF signals can penetrate some obstacles, their effectiveness is greatly reduced when there is no clear line of sight between the transmitter and receiver. Obstacles can attenuate (weaken) or block RF signals, impacting accuracy and reliability.
  - Spatial Resolution: The accuracy of RF-based tracking can be limited by the wavelength of the radio signals used. Generally, RF systems provide less spatial resolution compared to optical or ultrasonic systems, which can be a limitation for applications requiring precise positioning.
  - Scalability and Infrastructure: Deploying RF-based localization systems often requires extensive infrastructure, such as multiple antennas or nodes placed throughout the environment. This setup can be costly and complex to scale, especially in large or dynamic areas.

Multi-modal Sensor Fusion

- Can we combine all of these sensing modalities to get the best of all



Multi-modal Sensor Fusion

- Traditional Approaches
  - Step1: Get position estimates & Certainty estimates for each sensor modality
  - Step2: Pass the estimates to an algorithm e.g., Kalman filter or particle filter for final position

Multi-modal Sensor Fusion

- E.g., IMU + Camera fusion with Kalman Filter
    - The Kalman Filter operates in two basic steps:
        - **Prediction** and **Update**.

Multi-modal Sensor Fusion

- Kalman Filter Prediction Step
  - This step projects forward the current state estimate to the next time step using a model of the system's dynamics. For example, if you are tracking a moving vehicle, you might predict its next position based on its current position and velocity.
  - Along with the state, the uncertainty associated with the state (often represented as a covariance matrix) is also projected forward. This predicted uncertainty is increased because the prediction step usually adds uncertainty.

Multi-modal Sensor Fusion

- Kalman Filter Update Step
  - When new sensor data is received, the Kalman Filter updates the predicted state based on this new data. This involves a comparison of the actual measurement with the predicted measurement.
  - The difference between the actual measurement and the predicted measurement is called the "innovation" or "residual". This innovation is then used to update the predicted state and reduce its uncertainty.

Multi-modal Sensor Fusion

- The math behind the Kalman Filter involves a few key equations, primarily linear algebra. Here's a simplified breakdown:
  - **State Estimation Equation:**

    $$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k(y_k - H_k\hat{x}_{k|k-1})$$

    Here, $\hat{x}_{k|k}$ is the updated estimate of the state, $\hat{x}_{k|k-1}$ is the predicted state, K_k is the Kalman gain, y_k is the actual measurement, and H_k is the measurement matrix.
  - **Kalman Gain Equation:**

    $$K_k = P_{k|k-1}H_k^T(H_kP_{k|k-1}H_k^T + R_k)^{-1}$$

    $P_{k|k-1}$ is the predicted covariance, H_k^T is the transpose of the measurement matrix, and R_k is the measurement noise covariance.
  - **Covariance Update Equation:**

    $$P_{k|k} = (I - K_kH_k)P_{k|k-1}$$

    I is the identity matrix.

Multi-modal Sensor Fusion

- IMU+Camera Case Study - Step 1: System Model
  - You need to define your system's state; let’s say it includes the robot's position, velocity, and orientation.
  - The state at time k can be predicted based on the state at time k−1 using the equations of motion, factoring in the outputs from the IMU.
- Step 2: Prediction
  - Using the Kalman Filter, you start by predicting the next state of the robot using the last known state and the system model. This prediction also includes the propagation of the state’s uncertainty (covariance).
  - The IMU data (acceleration and angular velocity) is used to predict how much the robot's position and orientation have changed since the last update.

Multi-modal Sensor Fusion

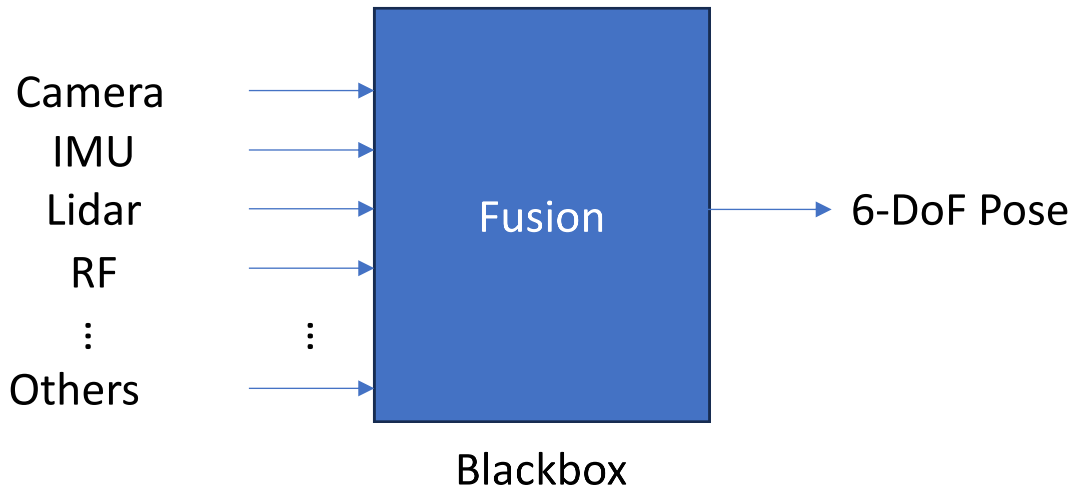
- IMU+Camera Case Study - Step 3: Measurement Update
  - The camera captures an image and uses visual cues to estimate the robot’s current position and orientation. This could involve detecting known features or landmarks and comparing their observed locations with known locations.

Multi-modal Sensor Fusion

- IMU+Camera Case Study - Step 4: Update with Kalman Filter
  - With the new measurement from the camera, the Kalman Filter computes the Kalman Gain, which determines how much weight to give to the prediction vs. the measurement.
  - The robot's state is then updated by combining the prediction with the new measurement, weighted by the Kalman Gain. This update adjusts the predicted state to be closer to the measurement, reducing error introduced by the IMU's drift.
- Step 5: Covariance Update
  - After updating the state, the Kalman Filter also updates the estimation of the covariance, reflecting the decreased uncertainty due to the incorporation of the measurement.

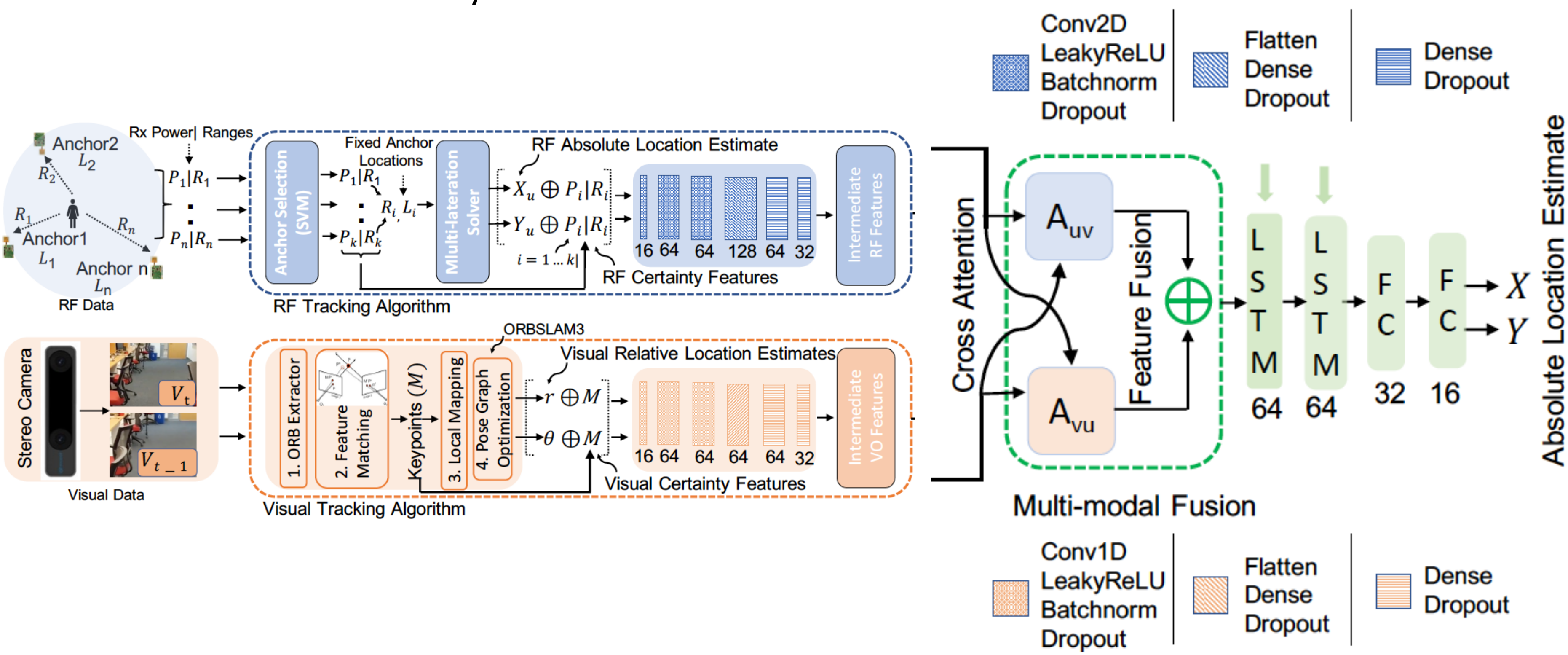
Multi-modal Sensor Fusion

- Recent advances in sensor fusion – machine learning

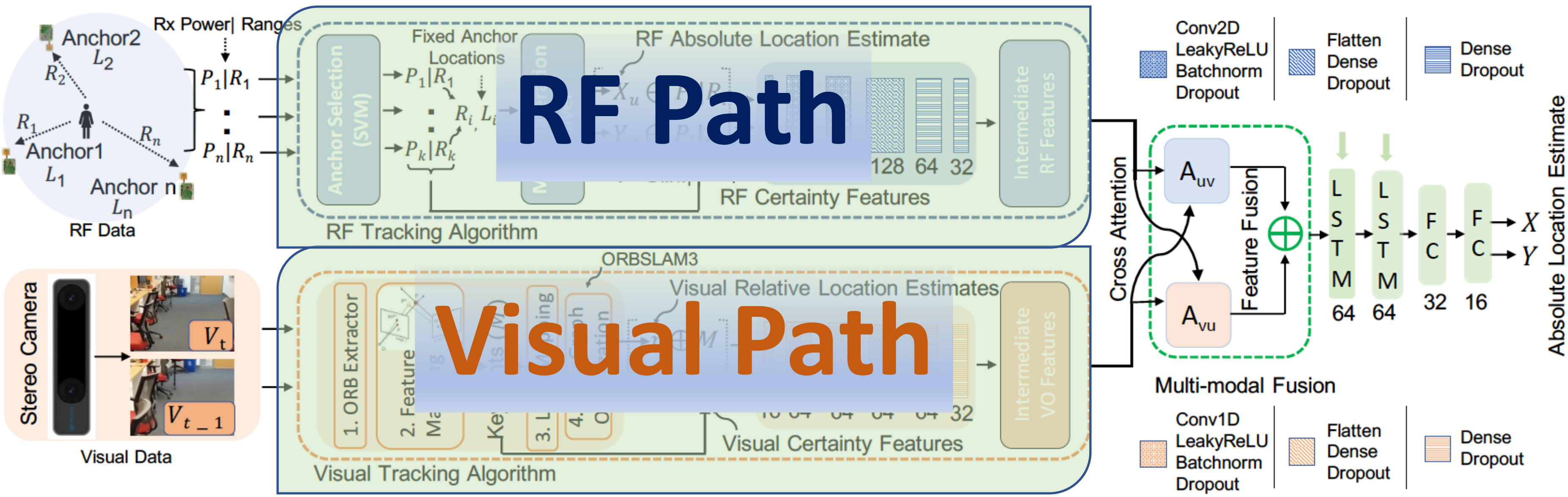


Multi-modal Sensor Fusion

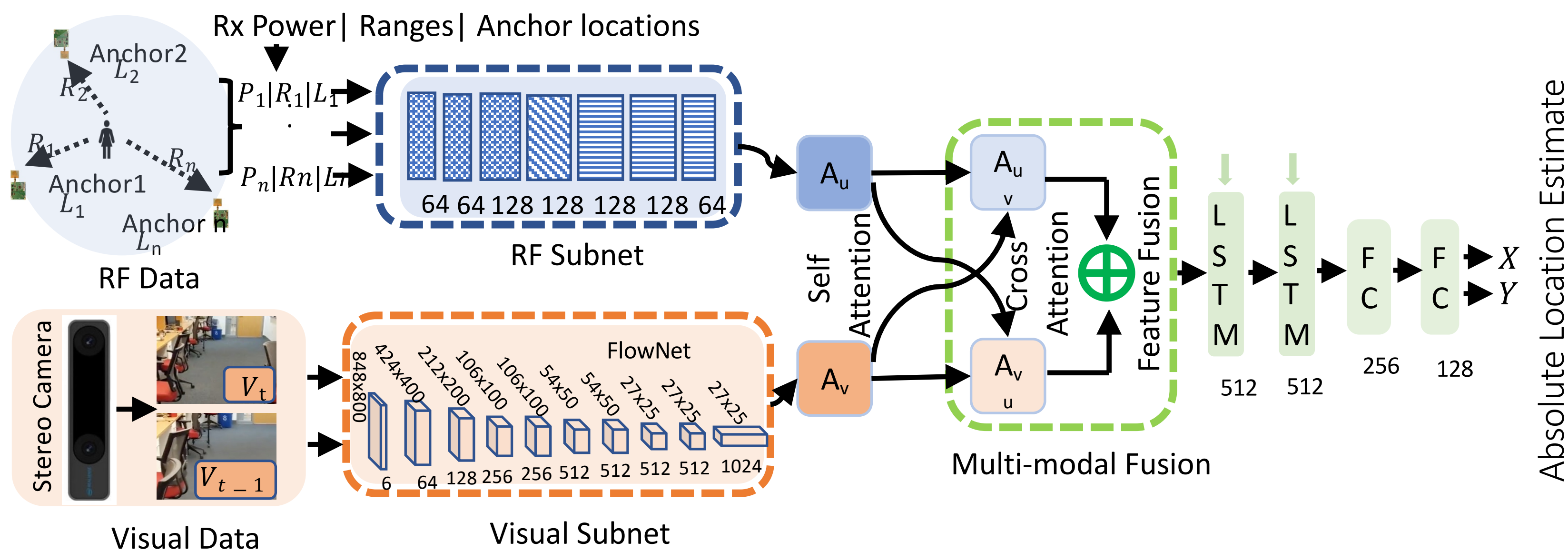
RF+Camera Case Study



Multi-modal Sensor Fusion



Blackbox Solution



Summary of the Lecture

- Sensing and tracking modalities, and their limitations
- RF tracking
- IMU tracking
- Multi-modal fusion for tracking

Next up: deep learning advances in XR content