UDRC-EURASIP Summer School

Sensing and Tracking

Sensor Fusion – Dr. David Cormack
Senior Systems Engineer, Leonardo

29th June 2021

About Me

- PhD in Signal Processing through the UDRC – finally graduating this Thursday!
- Worked on multi-target tracking with sensor fusion, plus a dive into sensor registration – some more on this later
- Now back at Leonardo as a Senior Systems Engineer developing tracking and fusion solutions for airborne platforms
- I’ve tried to keep a lot of maths out of these slides, and aim at a higher level
- Please jump in and ask questions as we go, especially if anything isn’t clear!

About Leonardo

LEONARDO

PROFILE

Leonardo is a global high-tech company in the Aerospace, Defence and Security sector, delivering advanced solutions based on dual-use technologies, to meet both military requirements and civil applications. As an industrial and technological leader, the Company aims to be an engine for development, contributing to security and progress worldwide.

OUR BUSINESS

Air
A wide range of aircraft, helicopters and avionics for commercial, public services, security and defence applications, as well as advanced solutions for training and simulation and safe air traffic management.

Land
A complete portfolio of platforms and systems to provide Armed Forces with integrated capabilities for ground superiority, in any conventional and asymmetric scenario.

Sea
Integrated solutions for total naval dominance that meet all the requirements of any type of modern ship, from the smallest units through to the largest aircraft carriers, enabling them to carry out any kind of mission.

Space
An integrated offer that covers the complete value chain of space industry, from satellite services for geo-information, communication and navigation, to space manufacturing, as well as equipment and payloads for exploration missions.

Cyber & Security
Integrated solutions for safety and security of territories, critical national infrastructures, citizens and enterprises, with the highest level of protection, situational awareness and information superiority.

Unmanned Systems
End-to-end Unmanned Systems with high interoperability with other platforms and a flexible multi-role capability, to fulfil a wide spectrum of missions.

INNOVATION

10 Leonardo Labs in Italy and US: an R&D corporate infrastructure transversal to all business sectors.

To **enhance** the potential of current technologies and keep current Leonardo's products competitive.

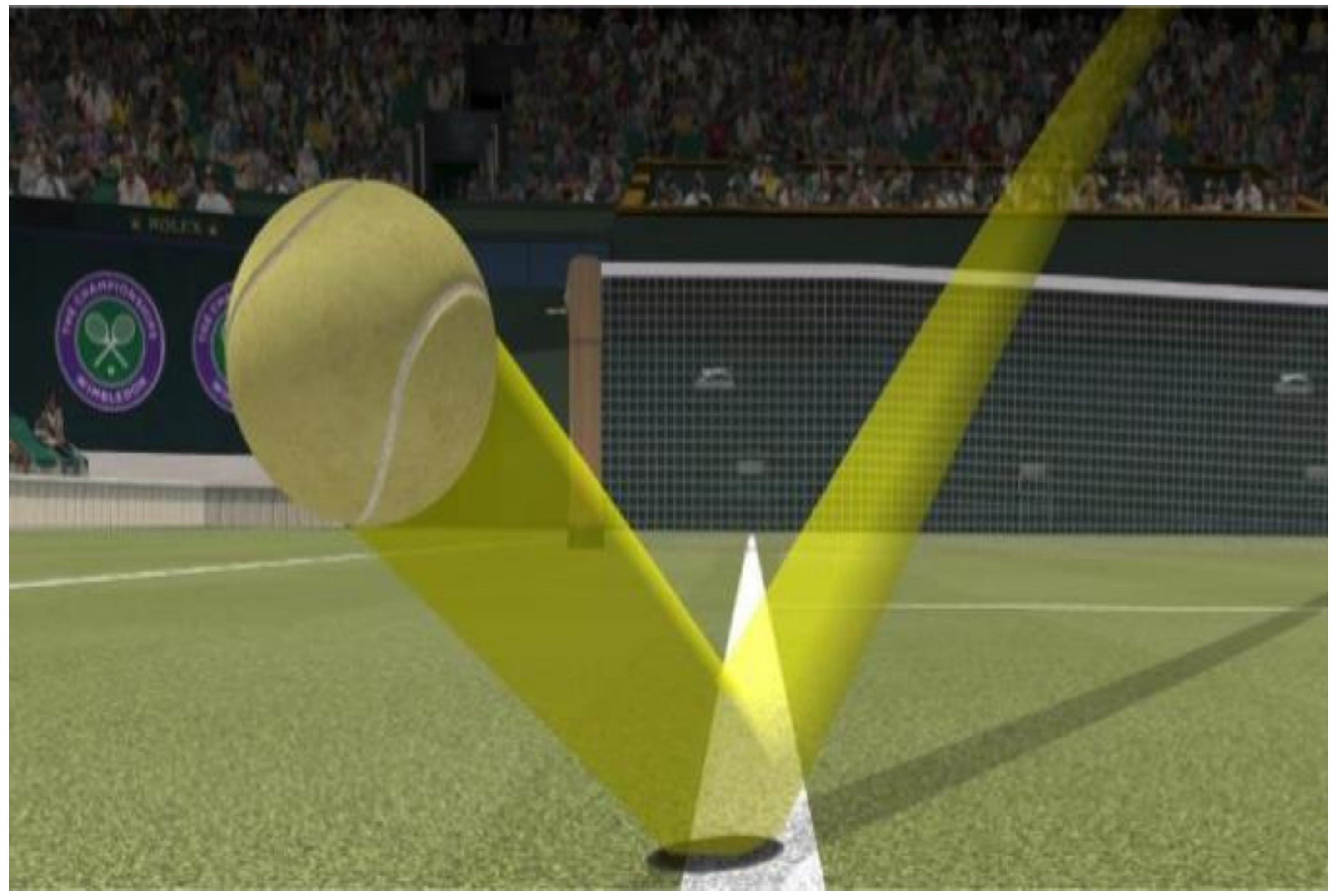
To **explore** emerging and disruptive technologies and anticipate future market demand.

- Future rotorcraft technologies
- Materials technologies
- Future Security & Safety technologies
- Future aircraft technologies
- Space technologies
- Future electronics & Sensing
- Quantum technologies
- High Performance Computing/ Cloud/Big Data
- Applied Artificial Intelligence
- Intelligent Autonomous Systems

- **8,800** people involved in R&D programmes and engineering
- **72** research fellows selected internationally reaching 400 in 2025 for our laboratories
- Cooperation with **70** Universities and Research centres
- **400** technologies in Leonardo's portfolio

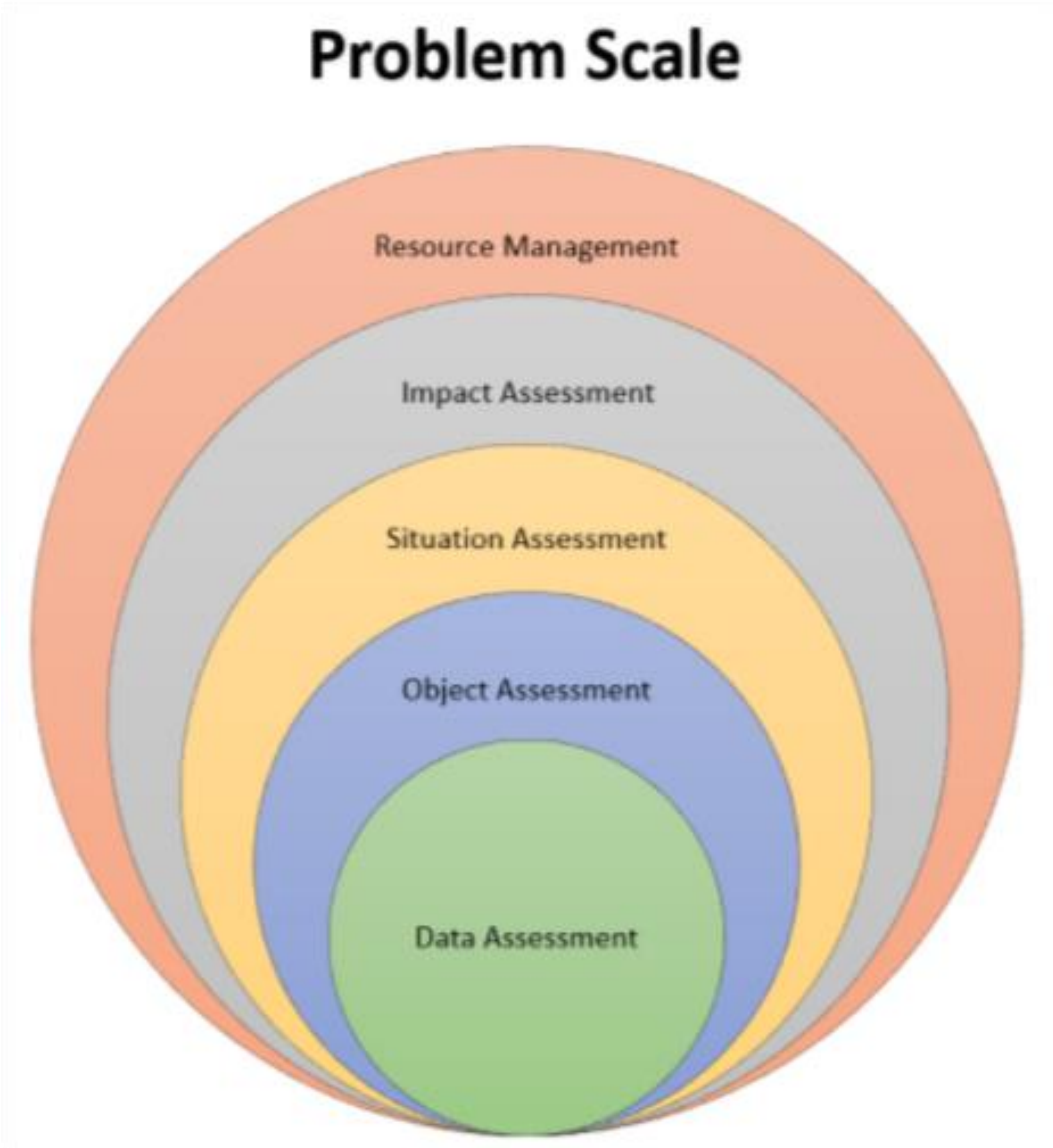
Background

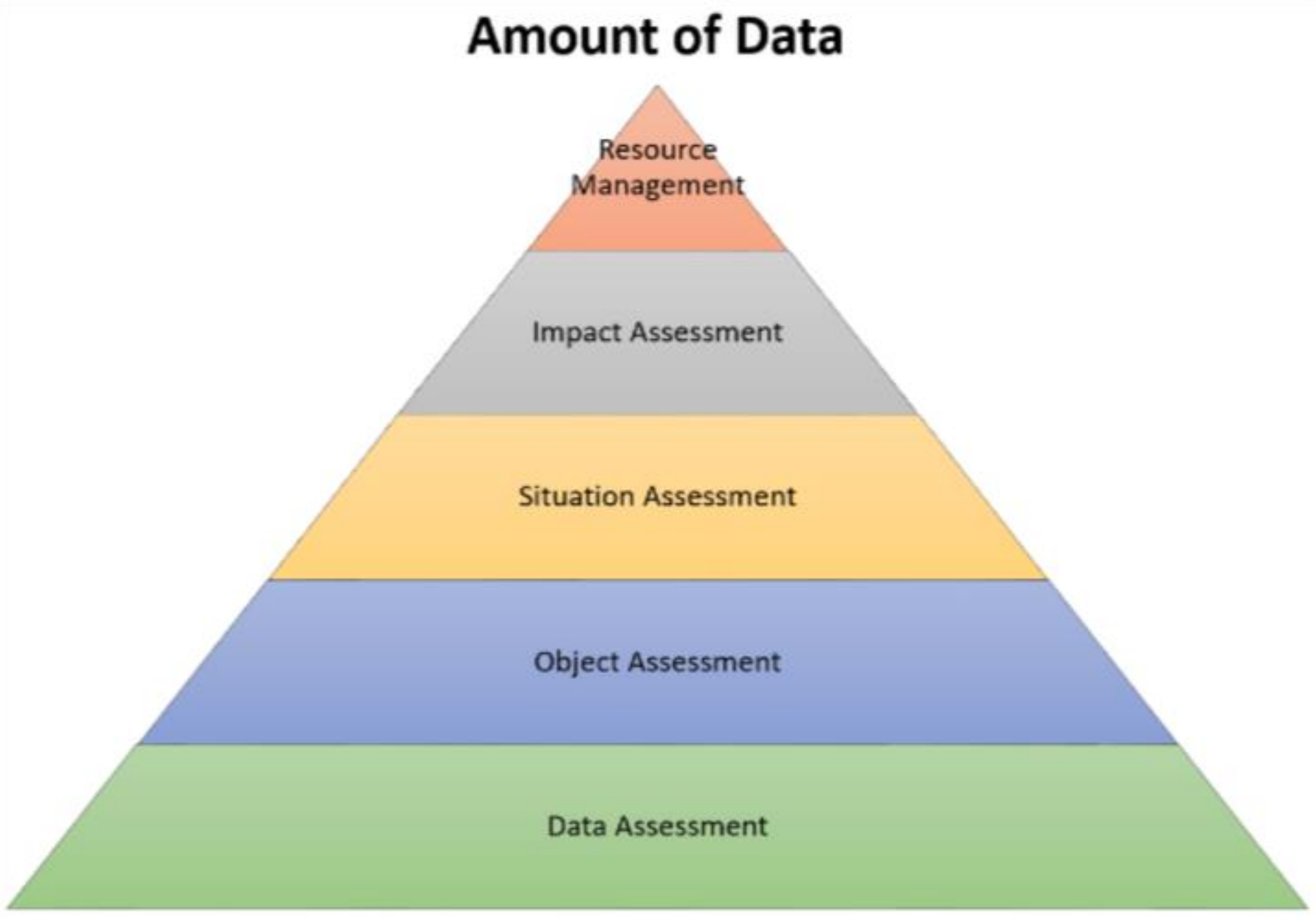
- Modern platforms often carry a range of different sensors onboard, which may carry out a range of different tasks
- These could range from avoiding obstacles in your new self-driving car, or being “able to see further” in your new plane
- Sensor fusion is basically the process of taking data from multiple sources, and combining it in a logical way to improve your awareness picture
- By “improve”, this could mean
    - Tracking targets at a faster rate;
    - Detecting targets further away;
    - Improved confidence in the stuff I observe.



Background

- Sensor fusion is often broken down into a number of different levels, depending on what data you are using, and what it relates to
- These are often referred to as the **JDL levels of fusion** in the literature – Google search will turn up quite a lot on this!
- As we move up through the different levels, the physical scale of the problem increases
- Because we are constantly refining our data using various algorithms along the way, the amount of data will likely get smaller at higher levels
- It would take a lot longer than this session to cover all the levels in detail. For now, we will focus mainly on **Object Assessment** and **Situation Assessment.**





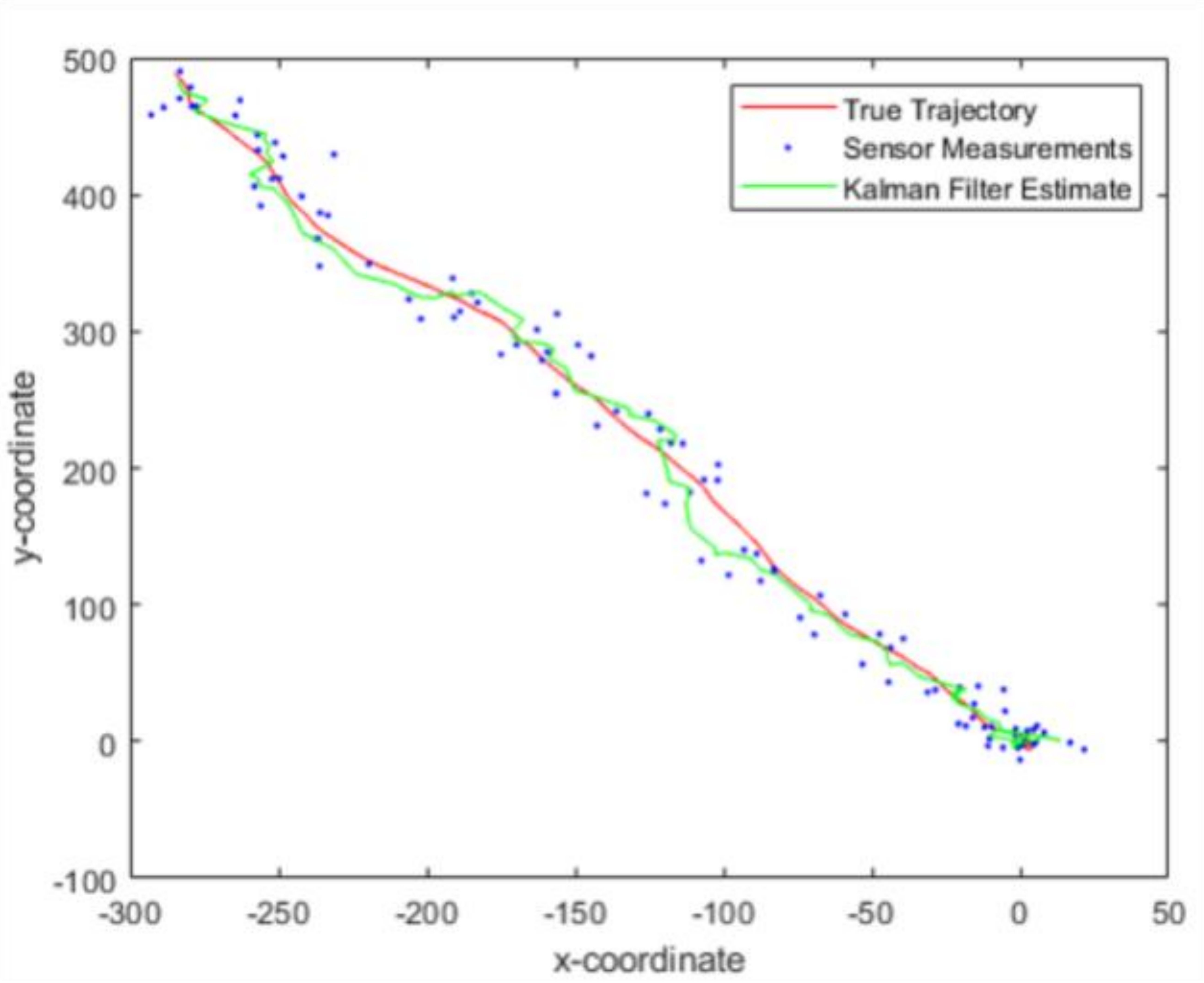
Background

- Sensor fusion often works closely with state estimation and tracking steps to refine the track picture

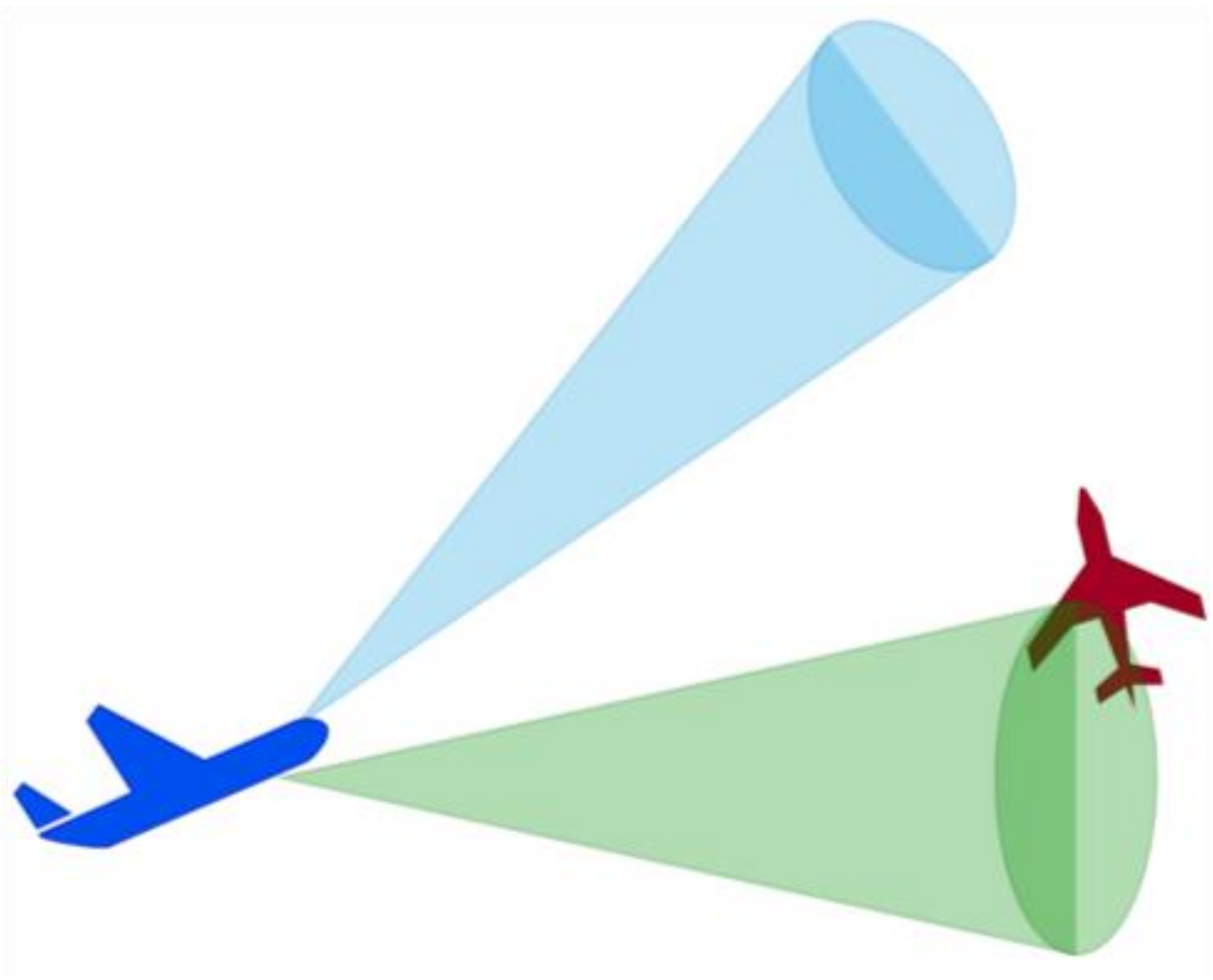
- As we have seen today, there are a plethora of tracking techniques available to us, each with their own relative benefits and drawbacks

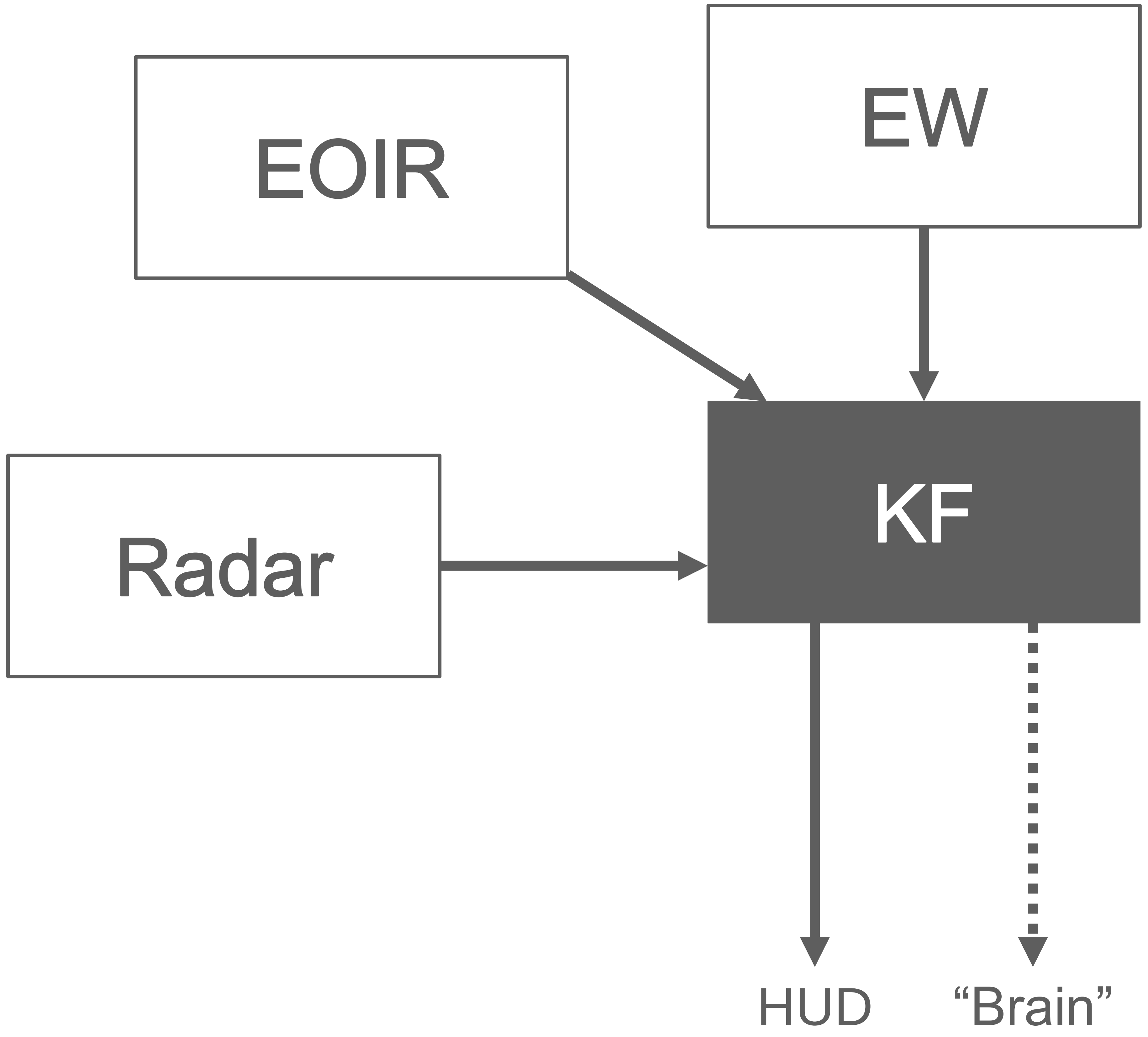
- To keep things simple, let's assume for the purposes of tracking (and possibly the fusion itself), that this is carried out using a **Kalman filter (KF)**

- Let's now take a look at some of the different architectures, starting with the **Object Assessment** level



Centralised Fusion





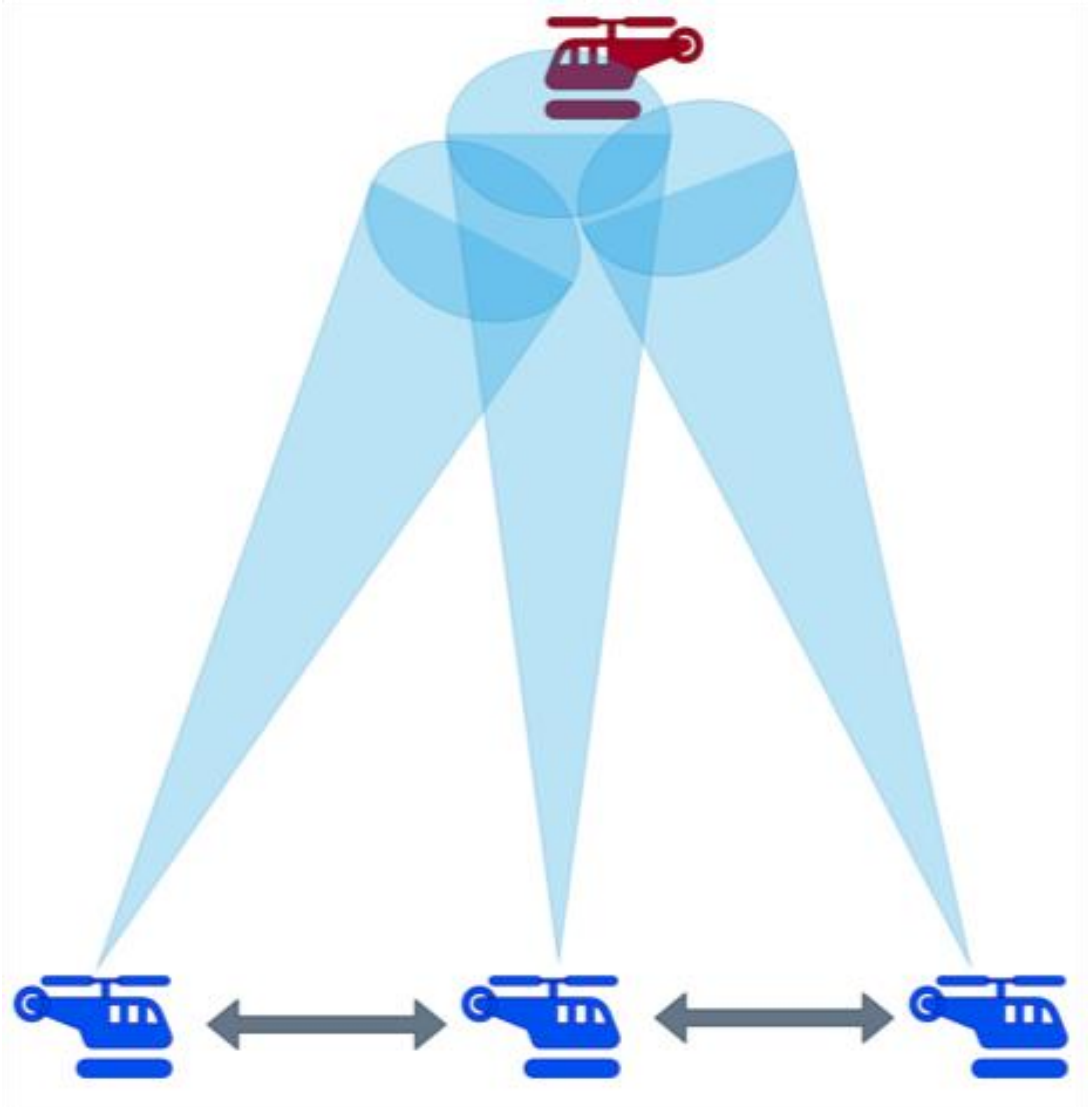
Centralised Fusion

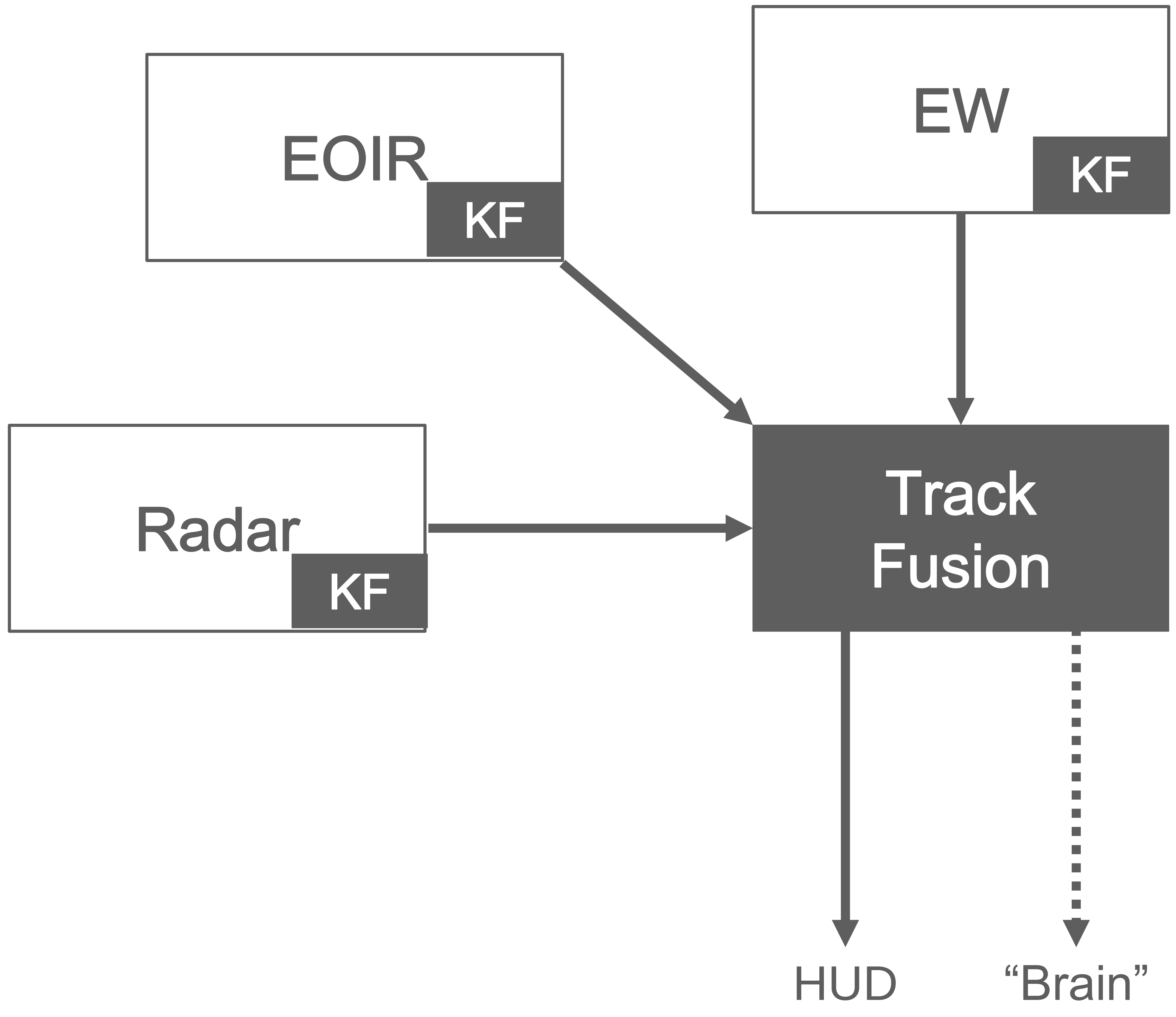
- In centralised fusion, we are typically working at the **Object Assessment** level where an object(s) have been detected by different sensors using appropriate signal/image processing.
- Sensors can then pass their plots/detections to a centralised tracker, which will then form a track on each individual object in the scene.
- Tracks can then be displayed appropriately, and potentially passed on to some form of sensor management to make decisions about what to do, and where to look next.

Pros	Cons
"Optimal" solution	More data being sent
Usually easier to tune parameters	Can be more computation hungry
Single track picture of whole scene	Lack of redundancy
Data stored in one place	
Increase in track confidence	

Live Script 1: Centralised Fusion

Distributed Fusion





Distributed Fusion

- In distributed fusion, we are moving towards the **Situation Assessment** level of the diagram, where objects have already been detected and tracked within an individual sensor or platform, and you are now attempting to combine or fuse data across them.

- Tracks from each individual sensor or platform are brought together in a central location and fused using a different type of algorithm e.g. covariance intersection.

- As with the centralised fusion, we still end up with a track picture that should incorporate information from the whole region.

Pros	Cons
Computation distributed through network	More reliance on underlying assumptions
Single track picture of whole scene	Requires more tuning in each KF
Possibly easier to “retro-fit” to legacy	Often lower track confidence
	Limited redundancy

Live Script 2: Distributed Fusion

(Im)perfect Fusion

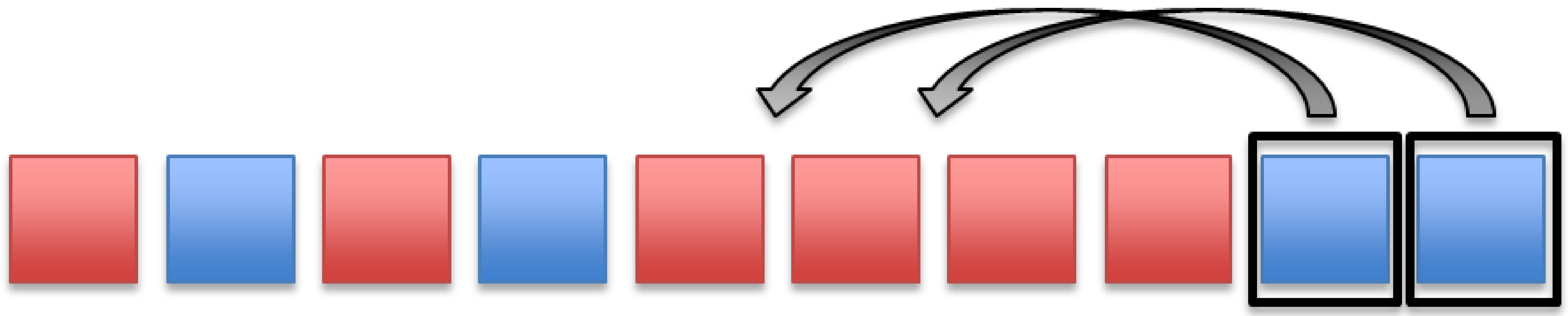
- So far, I have painted a picture that sensor fusion is relatively simple, where we just plug everything into the same tracker, or connect my various tracker outputs to a black box algorithm.
- Alarmingly, quite a lot of the tracking/fusion literature would have you think that this is the case.
- In practice, there are a number of overarching or system-level problems that are only discussed in more select areas of the literature. These problems really come to the forefront when we start discussing using multiple sensors or multiple platforms.
- For convenience, let's group them into three main strands –
    - Timing
    - Communications
    - Navigation

Timing

- When you open up a fusion-related paper on your preferred literature searching tool*, quite often you will see the assumption that the given sensor network is synchronised, and measurements/tracks will be available in all places at all times.

- Each sensor will likely have its own clock or timing hardware built-in, but can we guarantee that each of these clocks is synchronised?

- If the answer to that is no, we could end up with the following. Assuming that the sequence of measurements should be repeating red-blue, we may actually observe this sequence instead



- You can imagine that if we process this sequence in the wrong order as above, we could end up with a bit of a mess where tracks could split, or we start multiple tracks on the same object!

- In the literature, these problems are often referred to as **out-of-sequence data** and/or **network latency.**

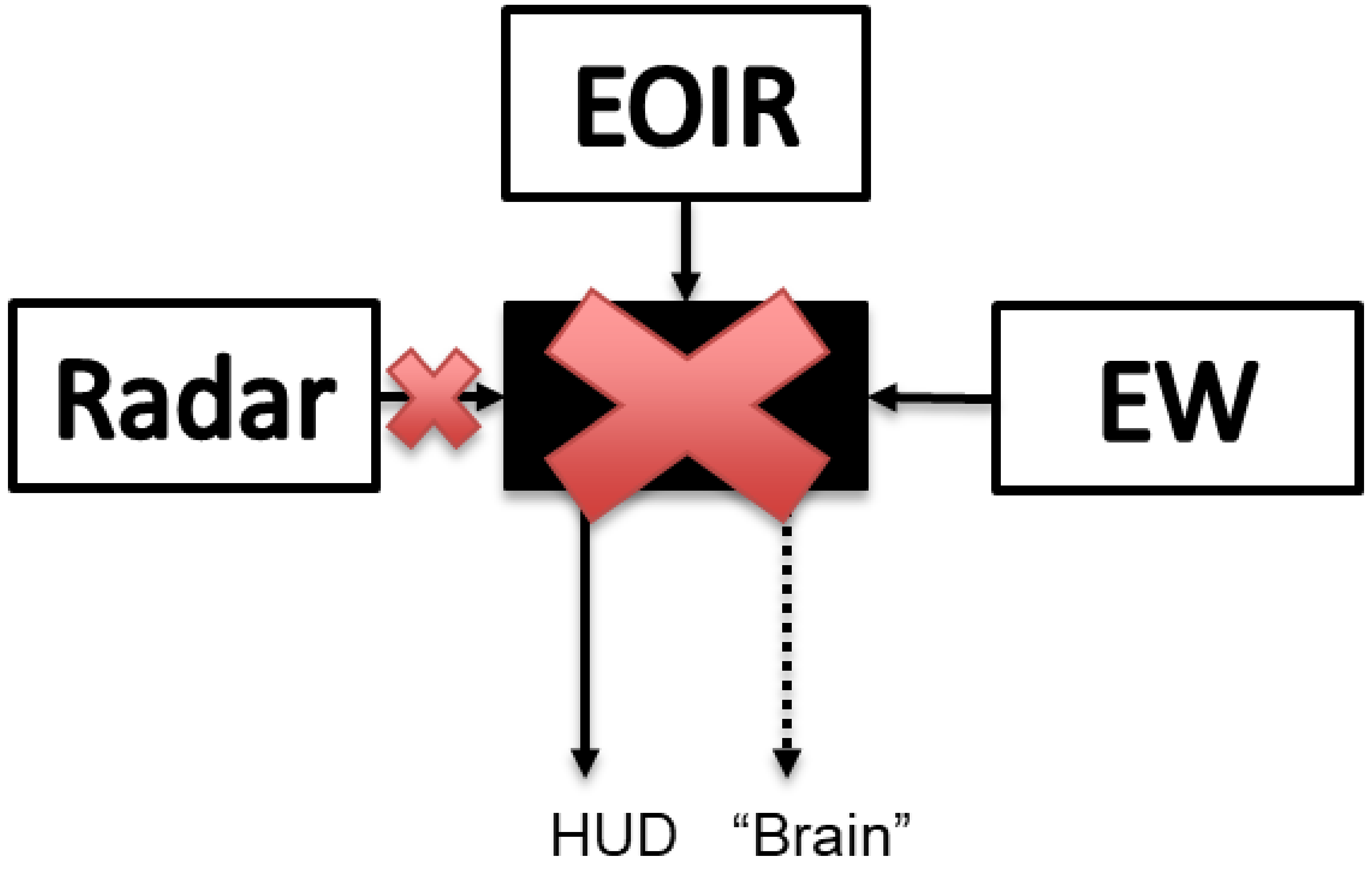
Communications

- Next, in the communications area, we could run into a number of different limitations.

- The first of these could be network bandwidth. When considering centralised measurement fusion, there is likely to be a lot of data being passed around, as typically the number of measurements is greater than the number of tracks. If we have any system “bottlenecks” or pipeline issues, the overall system could slow to a crawl!

- Another main concern is system redundancy. Consider the diagram and situation below. If for any reason we were to lose a link to a specific sensor, or if it is carrying out a different task, another sensor can hopefully take over; this is not a complete disaster.

- However, if we lose the main fusion engine for some reason, you can imagine that this puts us in a more difficult position.



Navigation

- This is a topic I know a bit more about; my PhD was most aligned to this area!
- Trackers often rely on a large amount of accurate navigation data to support a number of tasks:
  - Localising the platform;
  - Reference frame conversions;
  - Bias/registration compensation
- It is clear to say that if this data is not available, or is incorrect/incomplete, we can start running into deeper issues with tracking and fusing data.
- Something like a GPS-denied environment would be an extreme case; but there are some more “boundary” cases where biases or reference frames may slowly change over time, which can be more difficult to spot and correct for.
- Now consider this example -

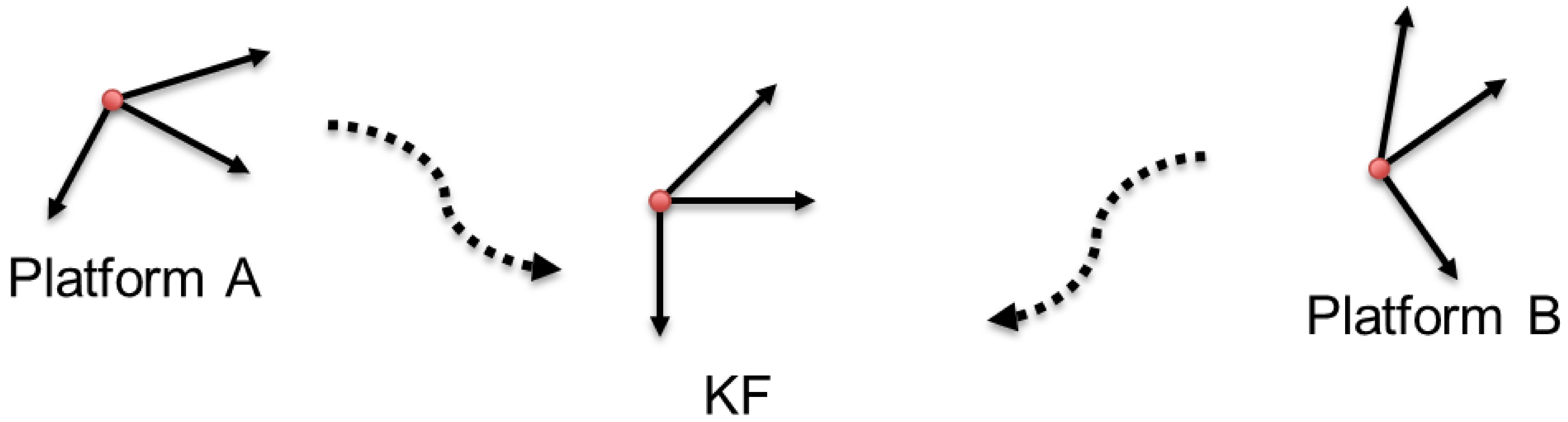
Navigation

- In order to perform fusion, data needs to be referenced to the same frame or point in space.

- For simplicity, I have only shown a rotational difference below, but you can imagine there may also be translational differences. In the literature, this is sometimes referred to as the sensor registration problem.

- Having issues with GPS could make the translation, or localisation, part of this problem much more difficult.

- Also, by having issues with some onboard navigation equipment, e.g. a gyroscope or IMU, could cause issues with the rotational part of the problem.



To conclude…

- I hope this has given you all at least a short introduction into some potential architectures that we could find in parts of a fusion system.
- With the overarching issues we face in practice, fusion systems take careful design which go beyond just “combining or fusing” the data as it were.
- For the types of fusion shown, there are of course lower-level design and implementation issues we will likely face which I haven’t had time to cover here.

Any questions?

Join the community

- Stone Soup code repository
  https://github.com/dstl/Stone-Soup
- Stone Soup documentations:
  https://stonesoup.rtfd.io
- Stone Soup chat room:
  https://gitter.im/dstl/Stone-Soup
- ISIF Open Source Tracking and Estimation Working Group:
  https://isif-ostewg.org