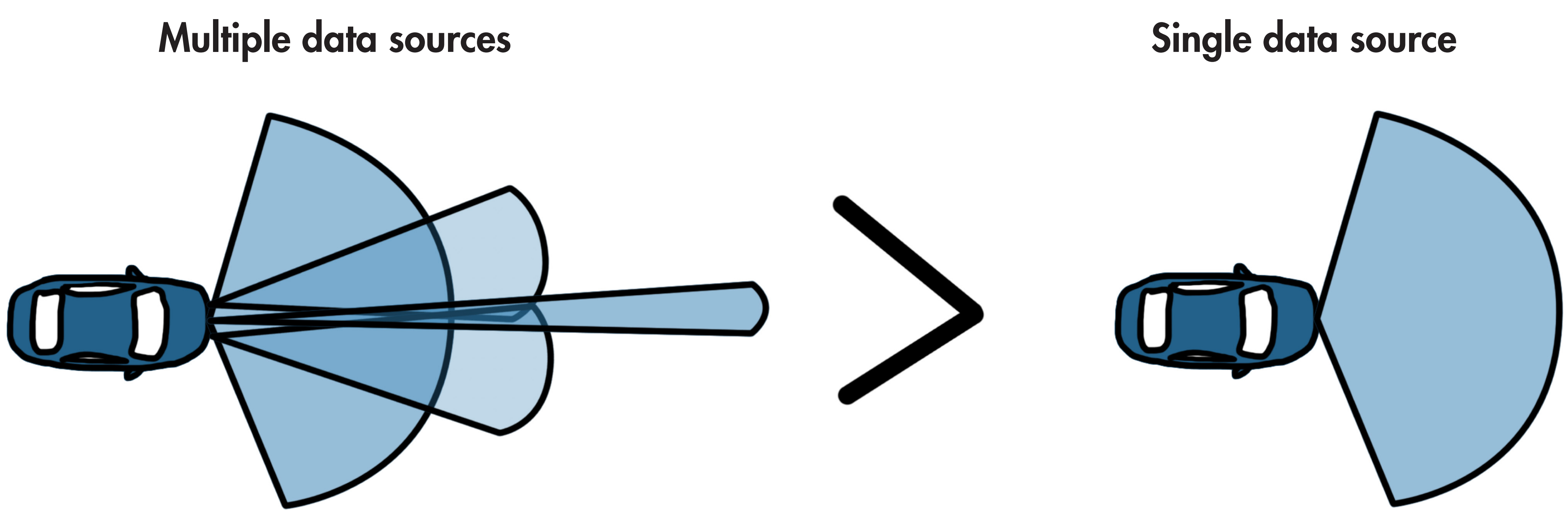
Sensor Fusion and Tracking for Autonomous Systems: An Overview

MathWorks®

What Is Sensor Fusion?

All autonomous systems have at least one thing in common: they use sensors and mathematical models to estimate the state of the system and the environment in which it is operating. Sensors and models are called *data sources* because both can provide information that an autonomous system can use.

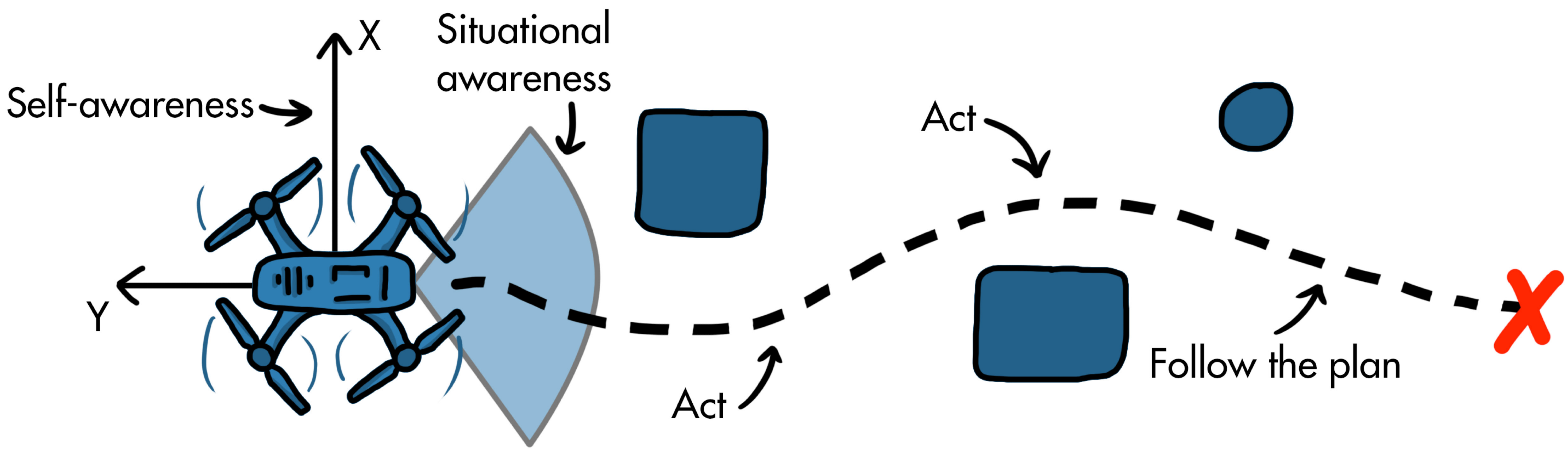
Sensor fusion is the process of combining two or more data sources in a way that generates an estimated state that is more accurate, more consistent, and more dependable than it would be with just a single data source.



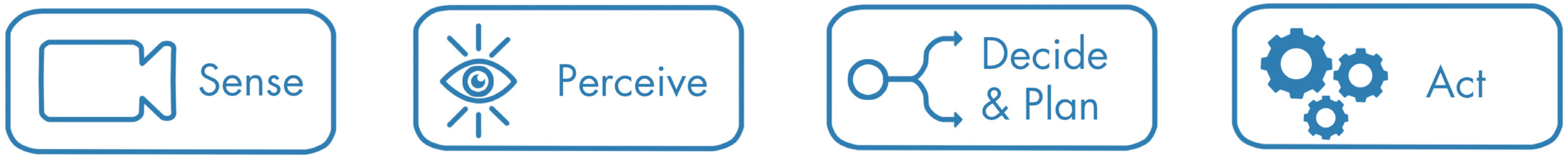
To understand the benefits of sensor fusion, we start by looking at the four general capabilities that an autonomous system needs to have.

Autonomous Systems Capabilities

The goal of an autonomous system is to operate within an environment without human interaction. The system needs to be able to understand itself and the world around it in order to determine which path to take and what the right commands are to get the system to follow that path.



We can divide these autonomous capabilities into four main areas: sense, perceive, plan, and act.



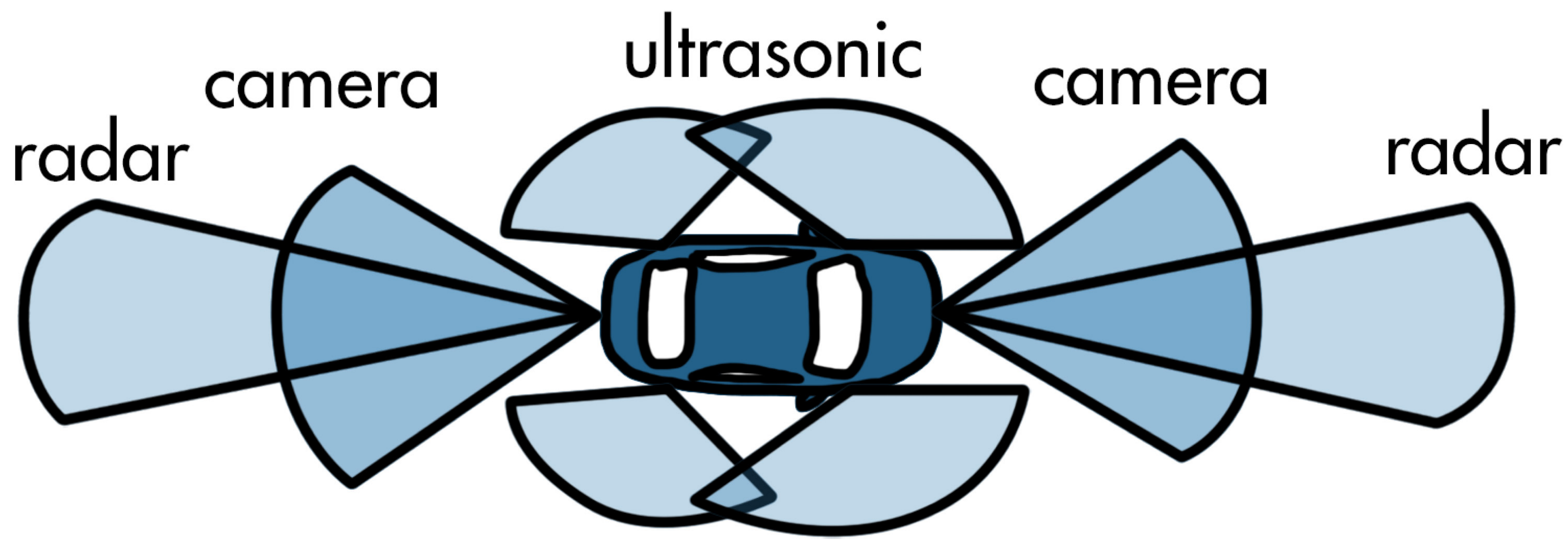
Sense

The sense step is where sensors are used to collect information about both the state of the system and the state of the external world. For example, an automated driving system may measure its own state—for example, determining its position using GPS—but it also may measure the state of the environment with externally facing sensors such as radar, lidar, and vision cameras.

The specific sensor suite is application dependent, but the goal of this step is always the same: collect enough information to enable the system to correctly interpret its situation and ultimately act on it.

Of course, simply gathering raw sensor data isn't immediately useful. The system also has to make sense of the data. This is the role of the next step, perceive.

Example of an automated driving system sensor suite



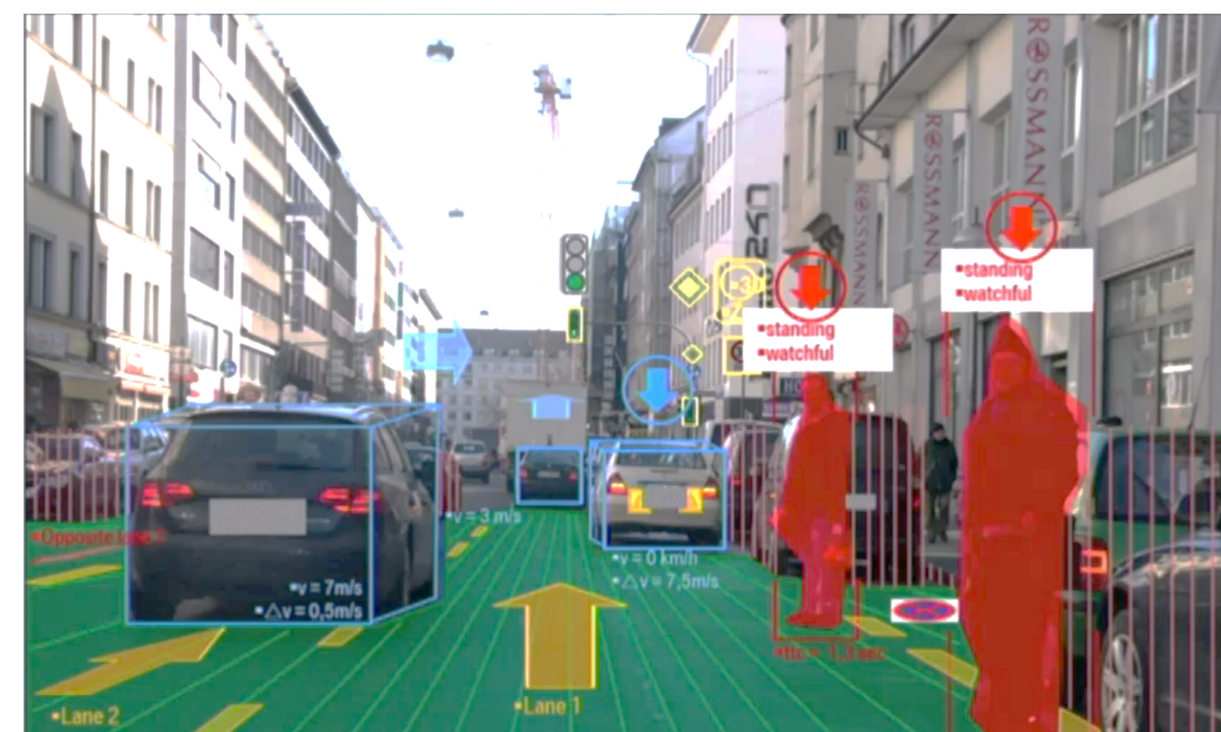
Perceive

Perhaps the easiest way to understand the perceive step is to think about a raw image from a vehicle's camera sensor. An image might have several million pixels, each with 8 bits or more of information in three different color bands. This is a lot of data! In order to make use of this massive array of numbers, an algorithm or a human needs to pull out something of value. In this way, perception is more than acquiring information; it's the act of interpreting it into a useful quantity.

For example, the image on the left is the raw output from a camera. An autonomous car has to ultimately interpret these blobs of pixels as a road with lane lines and recognize that there are other cars and pedestrians sharing the space. It also has to recognize the road signs that give additional information about the situation.

Sense

Perceive

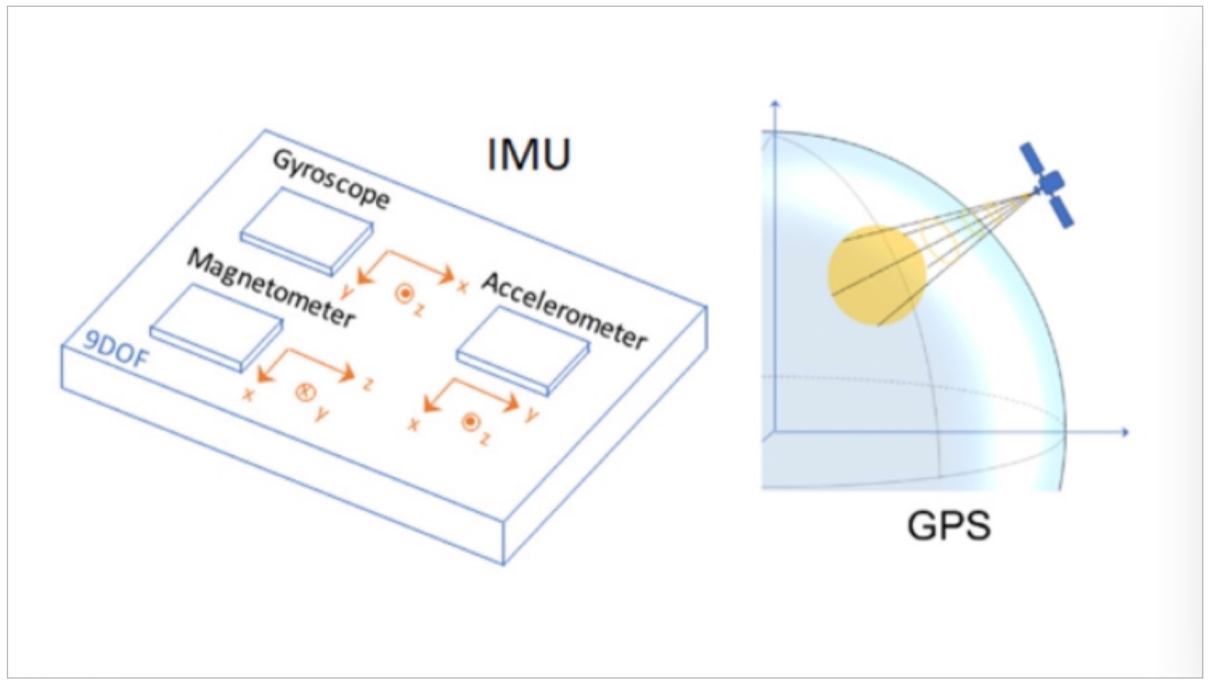


Perceive *continued*

Perception has two different but equally important responsibilities. It’s responsible for self-awareness, which is perceiving your own state, and for situational awareness, which is perceiving other objects in the environment and tracking them.

Self-awareness

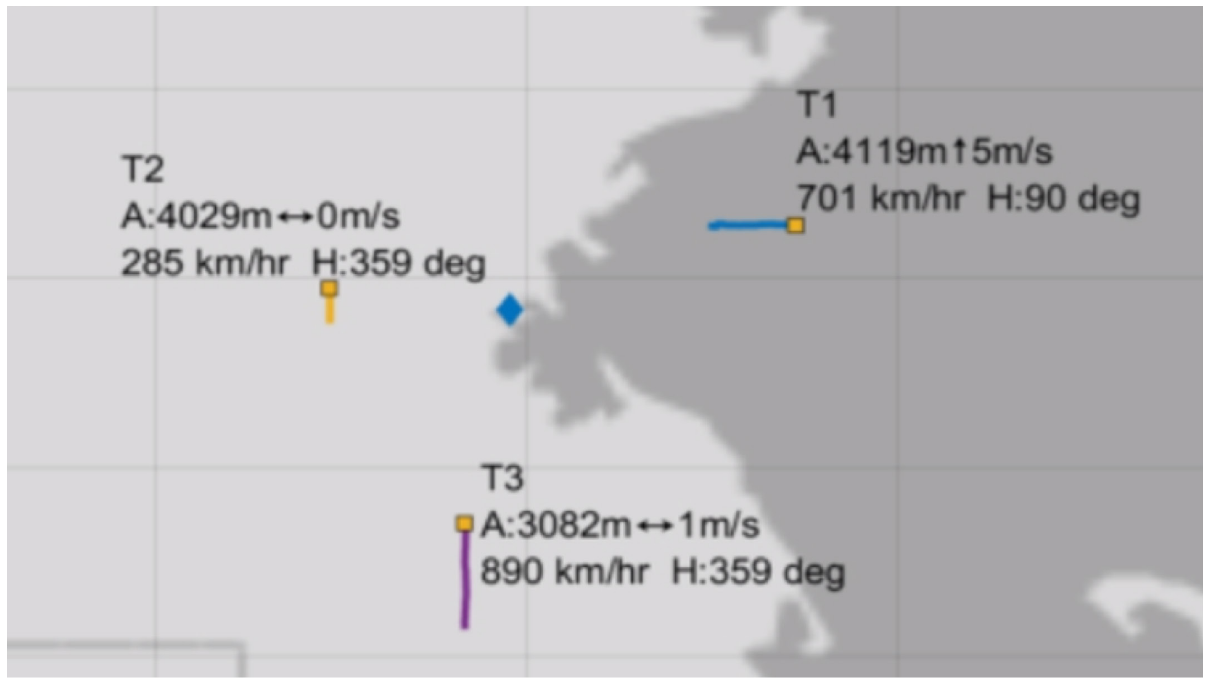
Where am I?
What am I doing?



Accelerometer, magnetometer, gyro, GPS...

Situational awareness

What is that?
What is it doing?



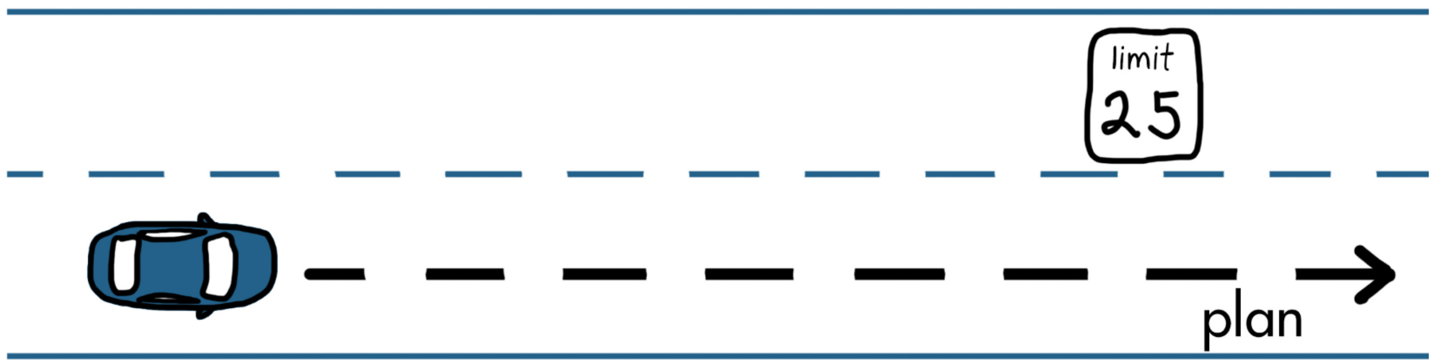
Radar, camera, IR, sonar, lidar, ...

A system needs to be able to understand itself and its surroundings in order to have enough information to make decisions. And the first step in decision making is creating a plan.

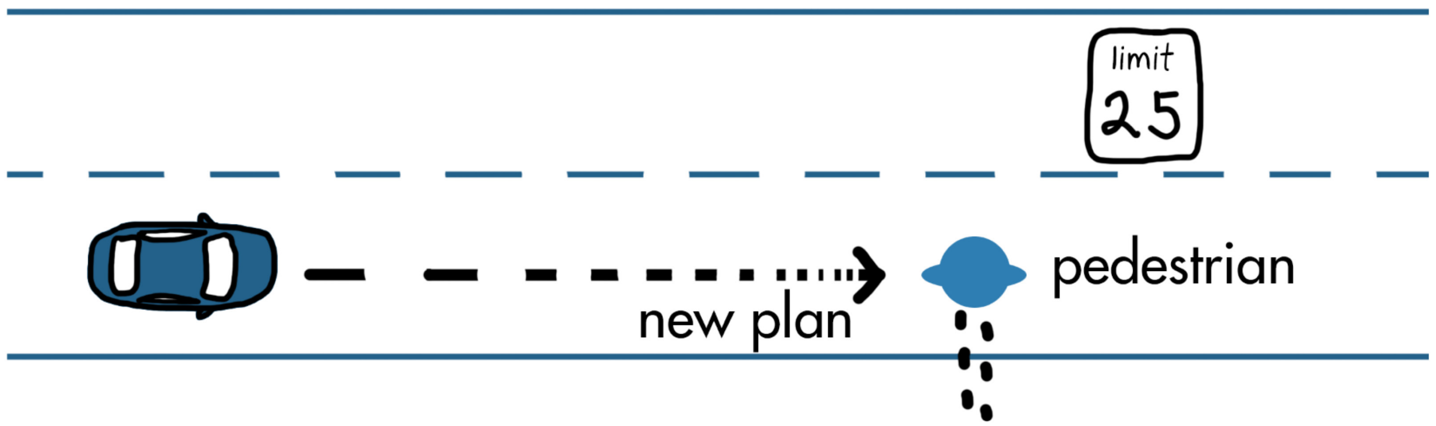
Decide and Plan

The planning step is where the autonomous system figures out what it would like to do and charts a path to get there.

For example, let's look at cruise control for a vehicle—a relatively simple autonomous activity. The goal of cruise control is to maintain a fixed speed even in the presence of disturbances. In this case, the plan would be to set the cruise control at the speed limit of the road.



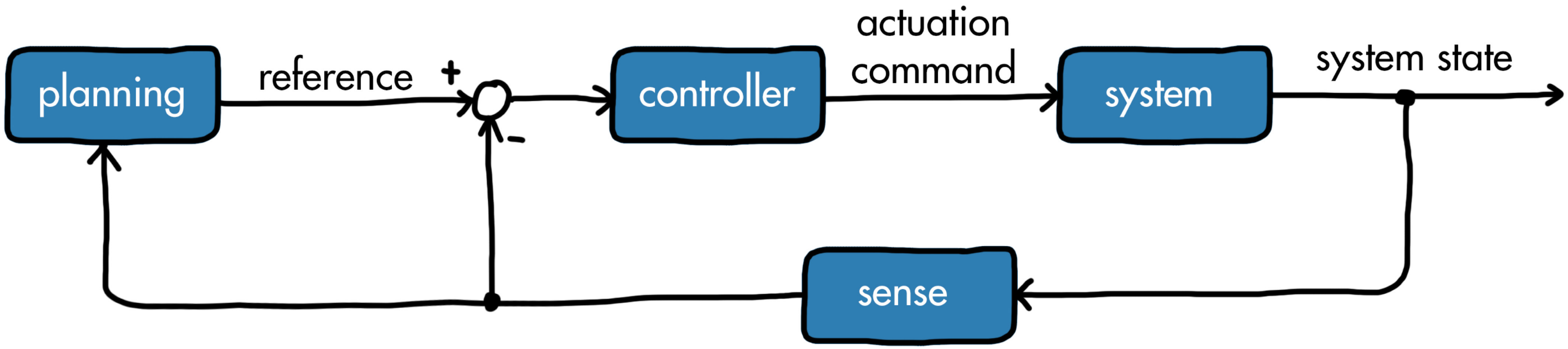
However, this plan falls apart if traffic is moving slower than the speed limit or if a pedestrian walks in front of the vehicle. In these cases, the vehicle would sense the change in the environment, perceive that change as an obstacle, make a new plan to slow down or stop altogether, and then ask the cruise control algorithm to follow that plan.



Act

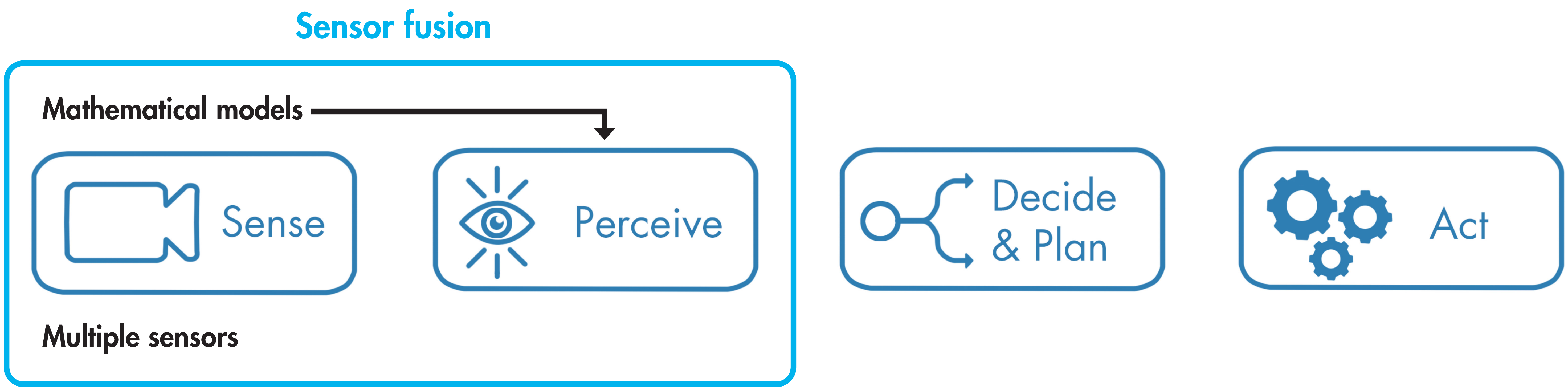
This brings us to the last step: calculating and executing the actions necessary for the system to follow the plan. This is the job of the controller and the control system.

So, we can think of the plan as the reference signal that the controller uses to command the actuators and other control effectors in a way that manipulates the system to move along that path. In the previous example, the actions are accomplished with the cruise controller. This controller is sensing the actual vehicle speed, comparing it with the reference speed, and then commanding the engine to produce the power necessary to drive the error to zero.



Where Does Sensor Fusion Fit?

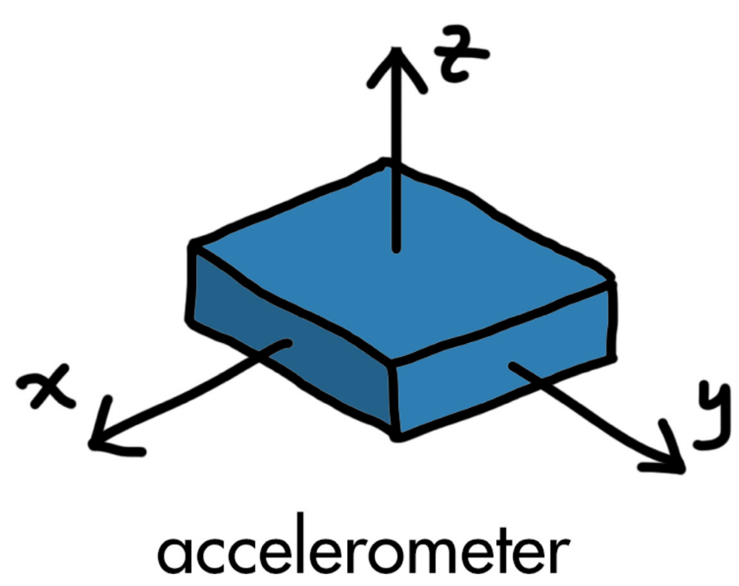
Sensor fusion straddles the sense and perceive steps. Sensor fusion is the way we can blend measurements from multiple sensors and combine them with additional information from mathematical models with the goal of having a better understanding of the world. The system can use this understanding to plan and act.

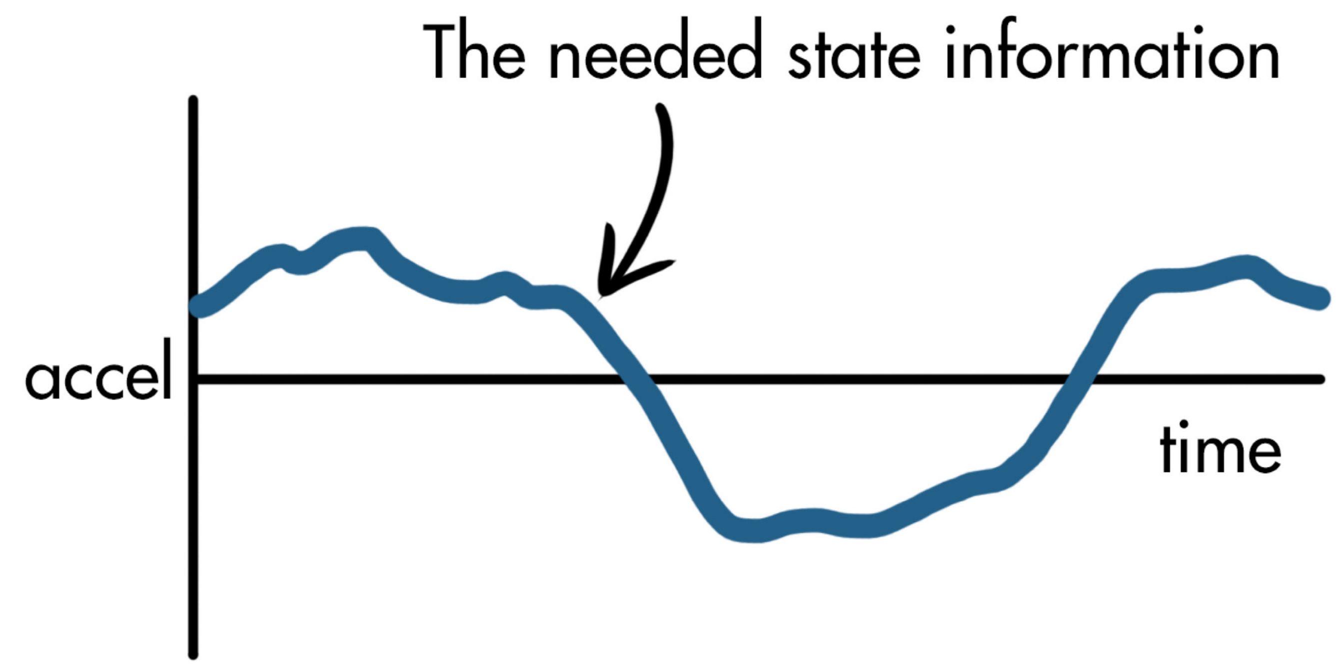


A Single Data Source

We could accomplish perception with just a single data source from which we could derive all of the important information. That is, we could estimate state using a single sensor and no model of the system. This simplicity is desirable, especially if a single data source is sufficient.

For example, if we need our system to be aware of how it's accelerating then we could use a single accelerometer. Here, the sensor measures the state we need, and we take care of the sense and perceive steps at the same time.



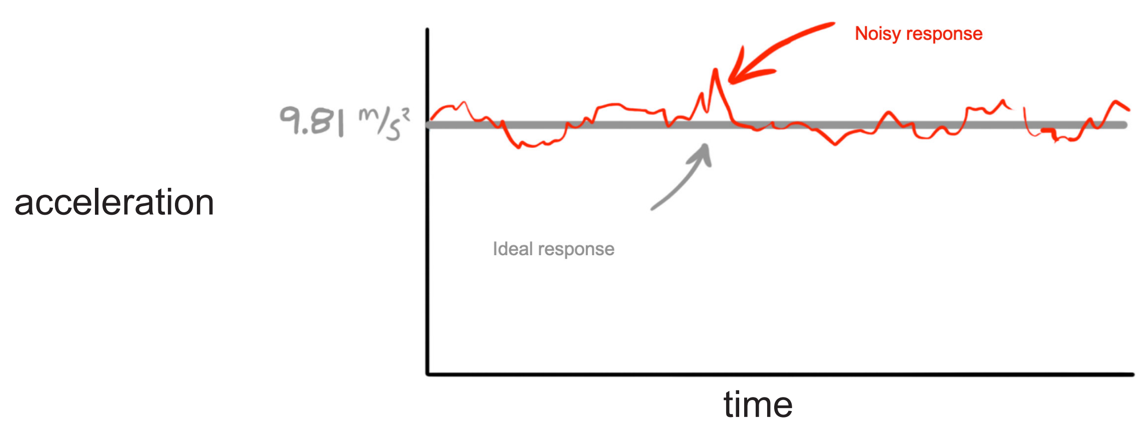


However, even if one data source is sufficient, fusing multiple sources together can provide additional benefits.

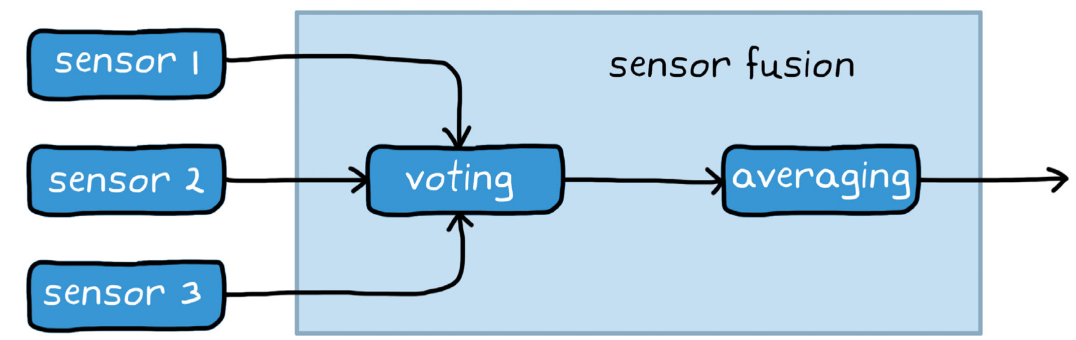
The Benefit of Multiple Data Sources

We can fuse together multiple data sources to:

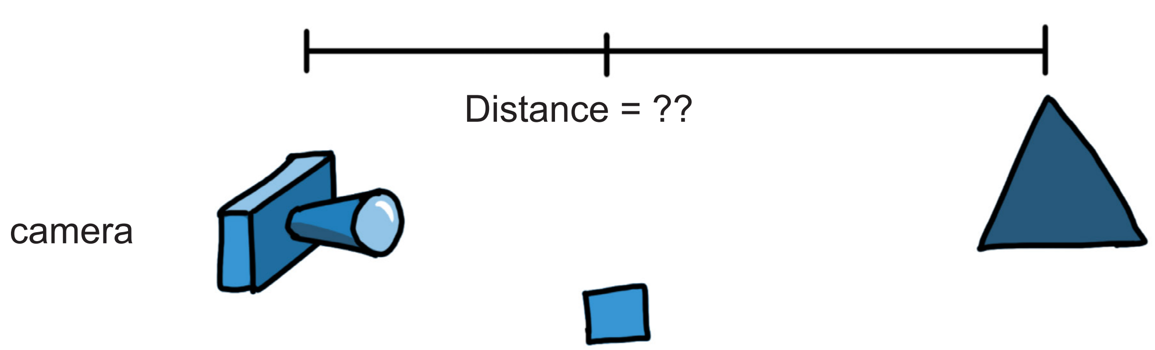
Increase data quality



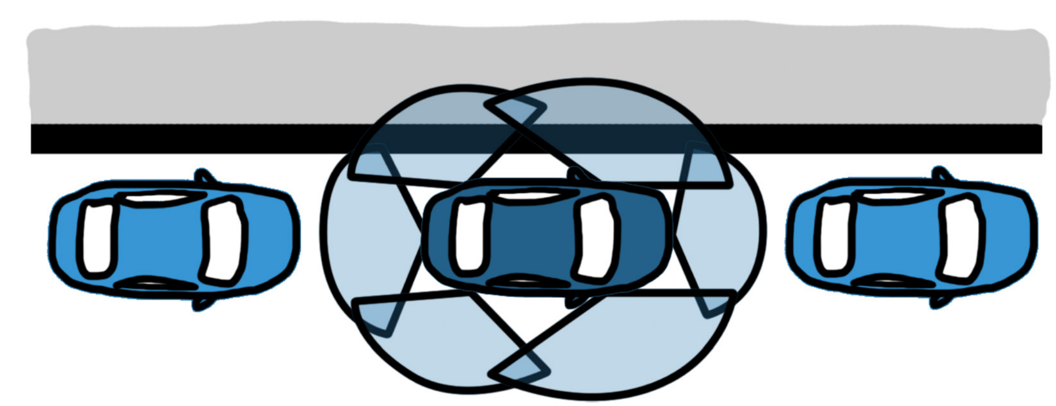
Increase data availability



Estimate unmeasured states



Increase coverage area

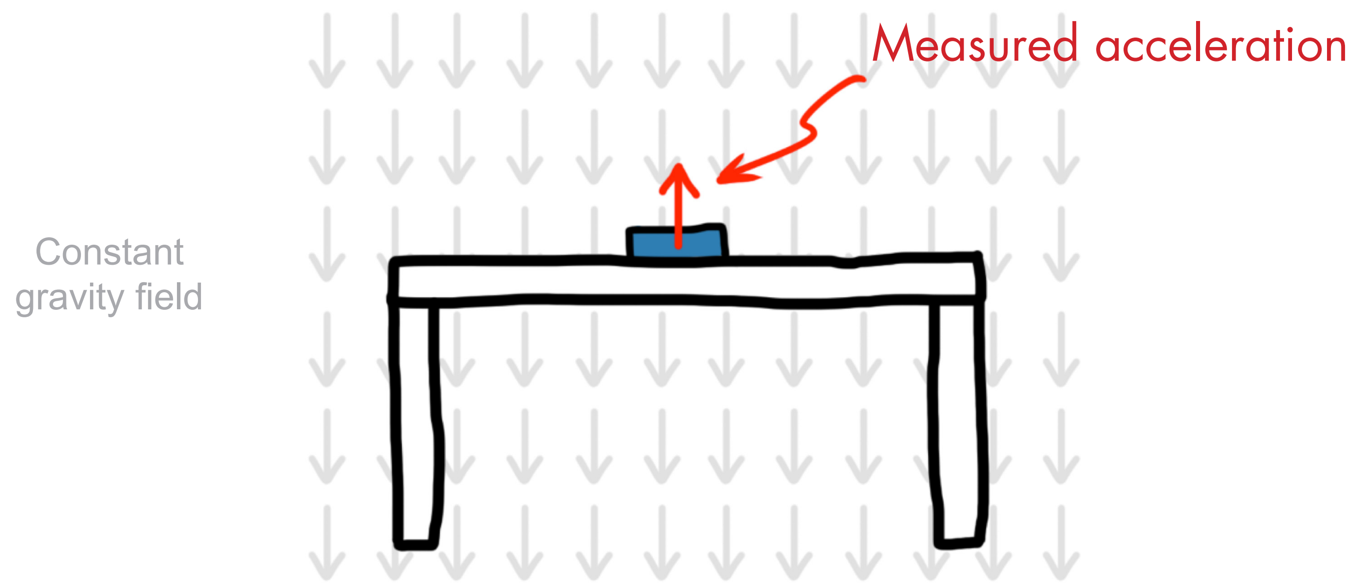


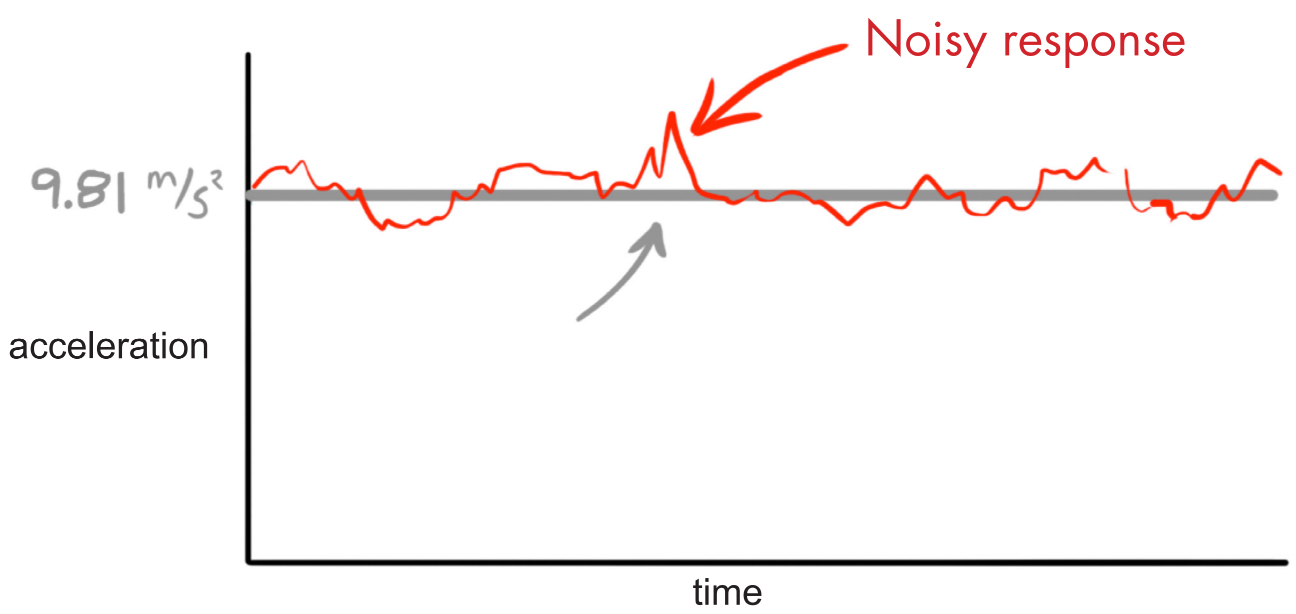
Let's go through each one in more detail.

Increasing Data Quality

Increasing the quality of data is one of the hallmarks of sensor fusion. We always want to work with data that has less noise, less uncertainty, and fewer deviations from the truth. Just overall, nice, clean data that we can trust! And often the best way to increase data quality is by fusing data from multiple sources.

As a simple example, let's take a single accelerometer and place it on a stationary table so that it's measuring only the acceleration due to gravity. If this were a perfect sensor, the output would read a constant 9.81 m/s^2. However, the actual measurement will have noise, which degrades the quality of the measurement.

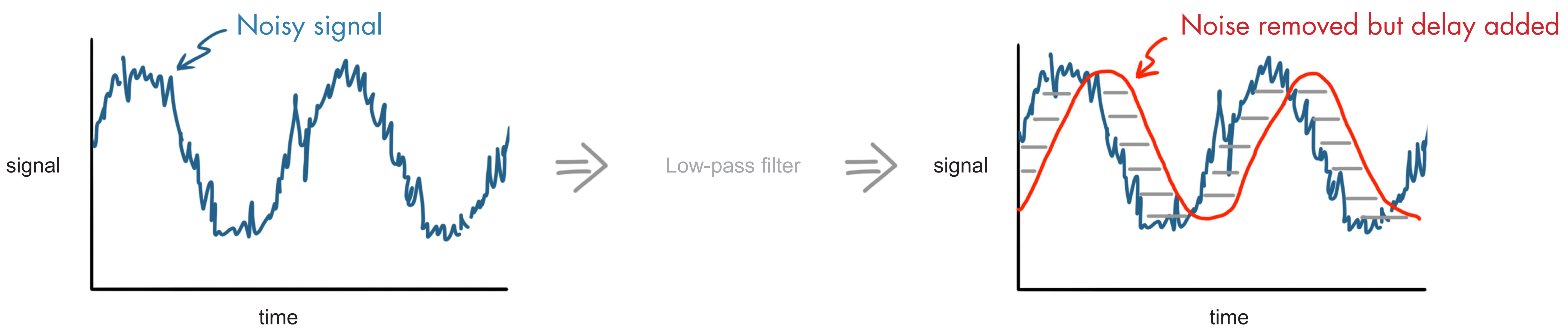




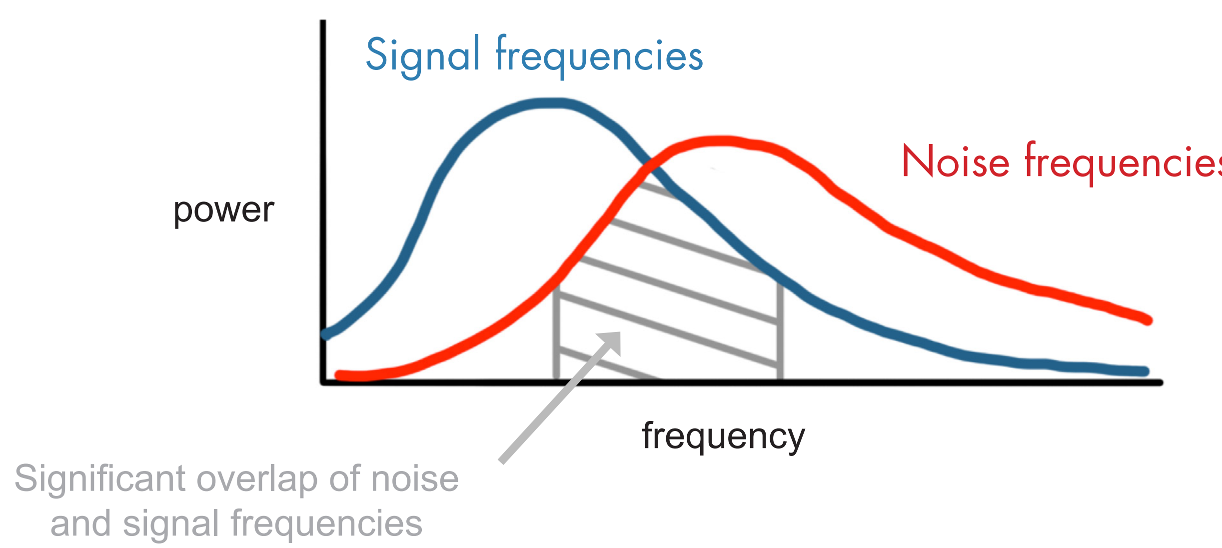
This is unpredictable noise, so we can't remove it through sensor calibration like we could if it was something deterministic like an axis misalignment or sensor bias.

Why Not a Low-Pass Filter?

If the sensor noise is at a higher frequency than the signal, it can be removed with some form of a low-pass filter (LPF). An LPF will pass through the low signal frequencies and attenuate the higher frequency noise. This can be a good solution but, unfortunately, it has the side effect of making the signal less responsive by adding delay.



Additionally, a low-pass filter won't work if the noise is at the same frequency as the signal that you're trying to measure. If the cutoff of the low-pass filter is set to remove most of the noise, then some of the signal will necessarily be removed as well. This is where sensor fusion can help. It can reduce sensor noise without adding delay or removing the real signal.



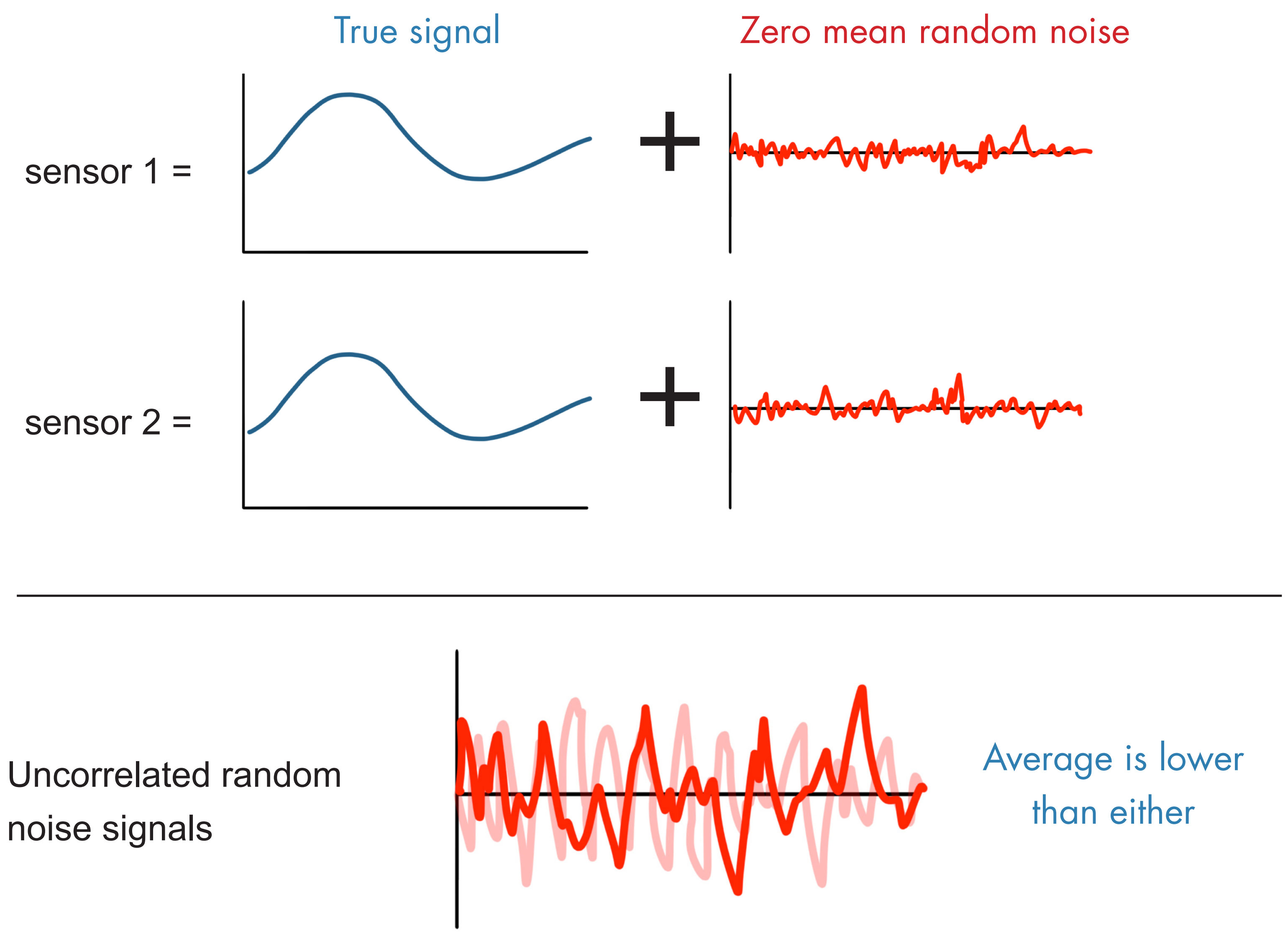
Doubling the Number of Sensors

Possibly the simplest way to increase the signal-to-noise ratio without adding delay is to add a second sensor and average the two measurements. As long as the noise isn't correlated between sensors and has a zero mean and constant variance, fusing similar sensors like this reduces the variance of the combined noise by a factor of the number of sensors. So, two averaged sensors will have half of the variance as a single sensor. We will see this shown mathematically in the next section.

We can visualize why this is true if we think of a signal as the superposition of the real quantity being measured and a random noise signal. If we average two sensors, the true portion of the signal will average out to be the same since both sensors are measuring the same true motion.

However, the average for the uncorrelated noise portion will be lower since the average of two different values is necessarily less than the largest value.

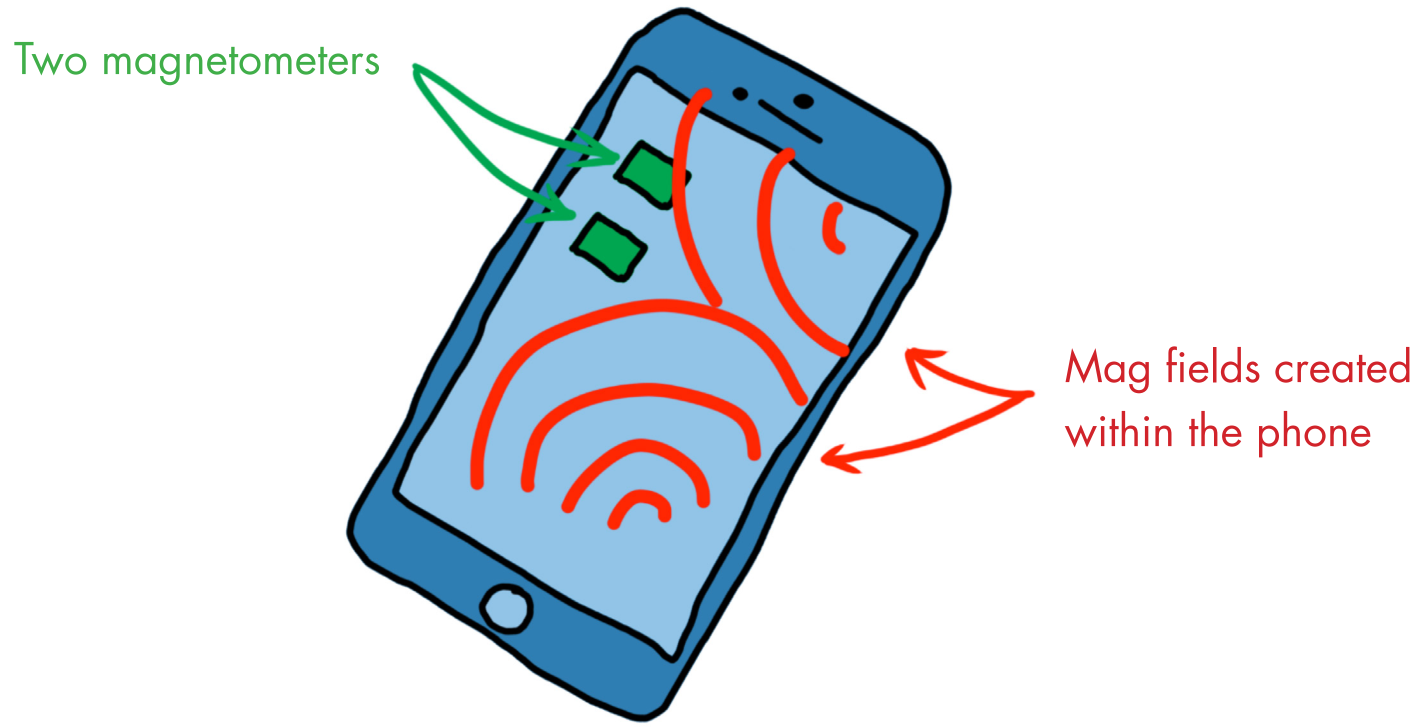
In this case, the sensor fusion algorithm is simply an averaging function.

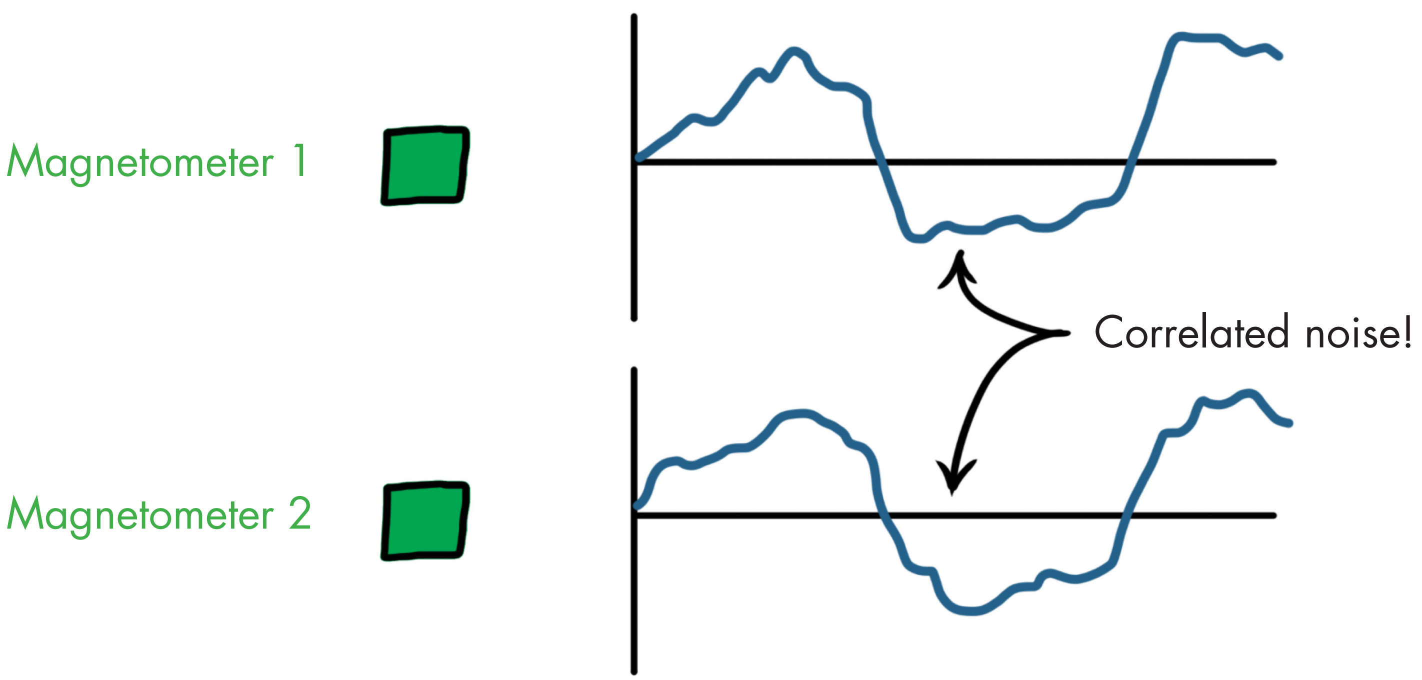


Doubling Sensors Doesn't Always Work

Doubling the number of sensors works when the noise isn't correlated—that is, the noise doesn't produce the same measurement error for both sensors at the same time. If that's not the case, and the noise is correlated, then adding more sensors won't help.

For example, let's say you're trying to measure the direction your phone is facing relative to magnetic north. We could use the phone's magnetometer to measure the direction of the magnetic field in the phone's coordinate frame. However, just like with the accelerometer, this measurement will be noisy, and if we want to reduce that noise, we may be tempted to add a second magnetometer.

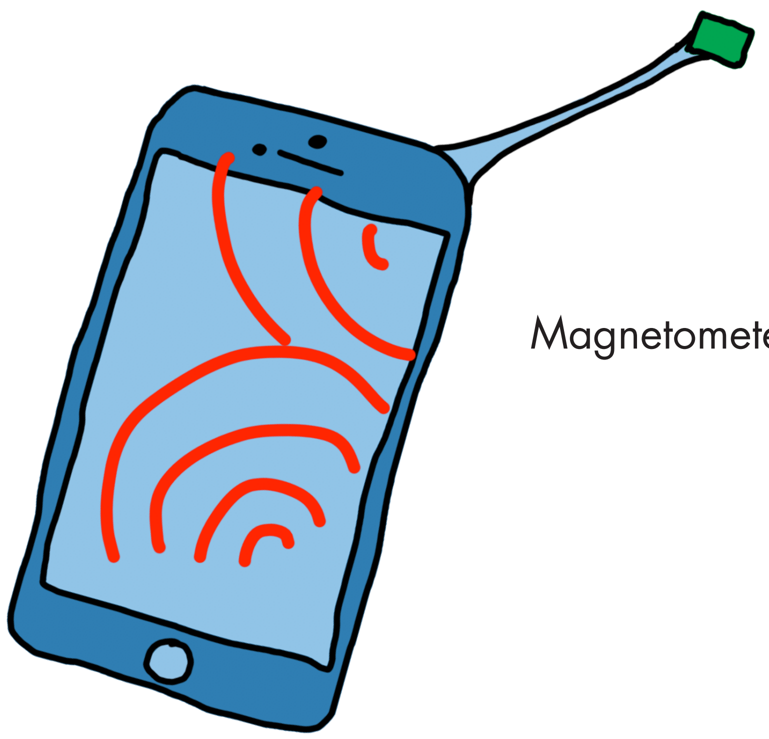


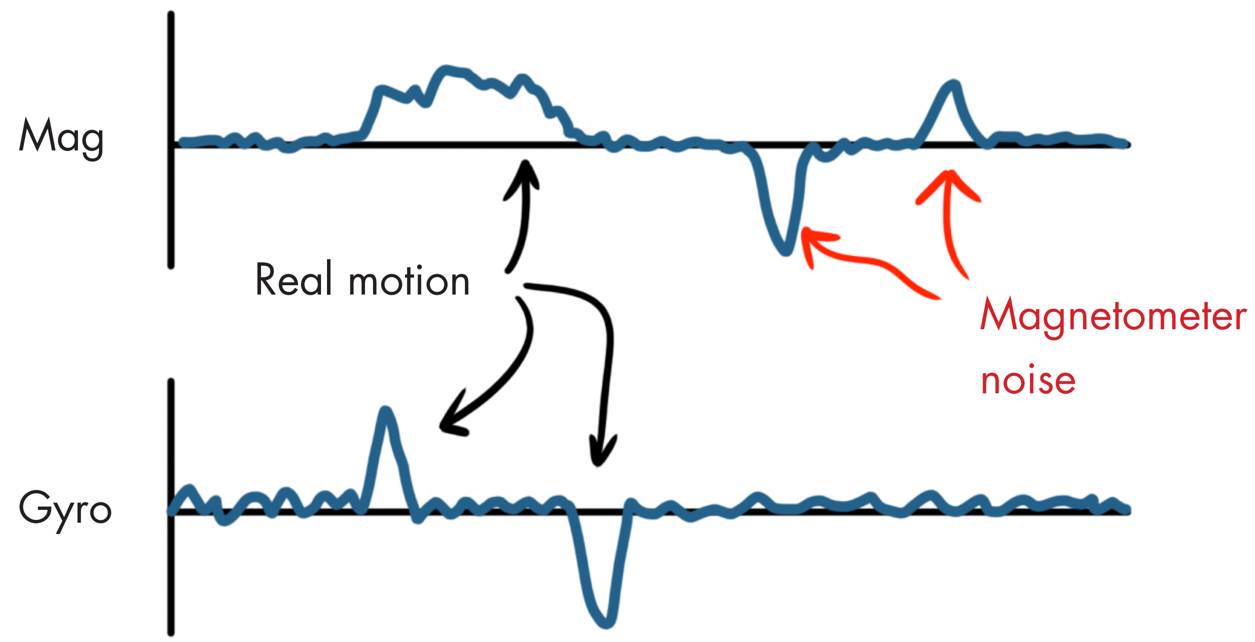


However, at least some contribution of noise is coming from the moving magnetic fields created by the electronics within the phone itself. This means that both magnetometer measurements will be affected in the same way by this noise source and, just like when the signal existed in both measurements, won't be removed through averaging.

Multiple Sensor Types

One way to solve the problem of correlated noise is to move the sensors away from the offending noise source. For the phone example, we'd have to move the magnetometers away from the internal magnetic fields. For spacecraft, the magnetometer can be placed at the end of a boom to get it away from the rest of the spacecraft electronics; however, this is hard to do with a phone that needs to fit in your pocket. Therefore, rather than duplicating magnetometers, another option is to fuse the magnetometer with a different type of sensor—one that can be used to estimate the orientation of the phone but isn't impacted by the same correlated noise source.





For example, a gyro that measures angular rate could supplement the magnetometer and help provide a more accurate estimate of the phone's orientation. The basic idea is that if the magnetometer measures a change in the magnetic field, the gyro can be used to confirm whether that rotation came from the phone physically moving or if it's just from noise in the magnetometer. We'll come back to this idea in the next section when we talk about how we can fuse multiple signals using a Kalman filter.

Increasing Availability

Increasing data quality is not the only benefit; sensor fusion also can increase the availability of the measurements.

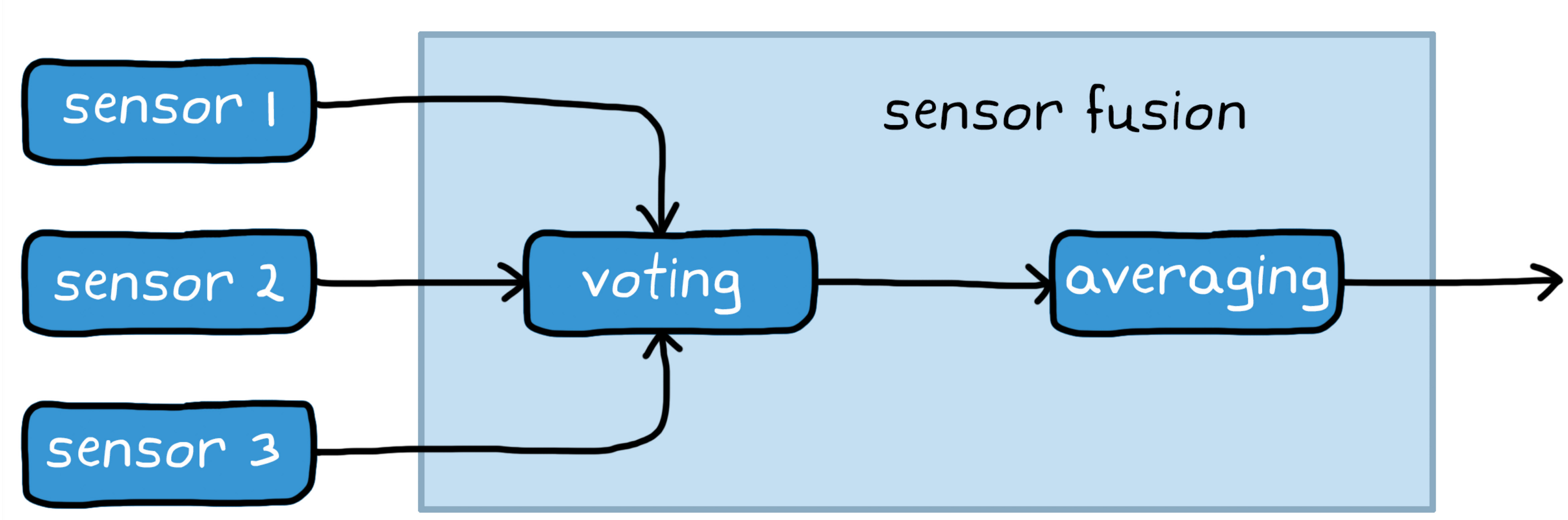
Here, availability means that we have a consistently valid measurement in the presence of sensor failures, loss of signal, loss of line-of-sight, and other causes of a missed measurement. Data availability is an important companion to data quality because it doesn't matter how great the data is if it's not available in the first place.

The next few pages show some of the ways that sensor fusion can help ensure that our system has access to the measurements it needs when faced with hardware failures and other obstructions.

Duplicating Sensors

If we have two identical sensors, like we had with the averaged accelerometers, then we have a hot back up in case one fails. Of course, with this scenario we lose quality if one sensor fails, but at least we don't lose the whole measurement!

We can also add a third sensor into the mix, and the fusion algorithm could vote out any one sensor that is producing a measurement that differs from the other two by a specified amount. The sensors that make it through voting can then be averaged.



Protect Against Common Mode Failures

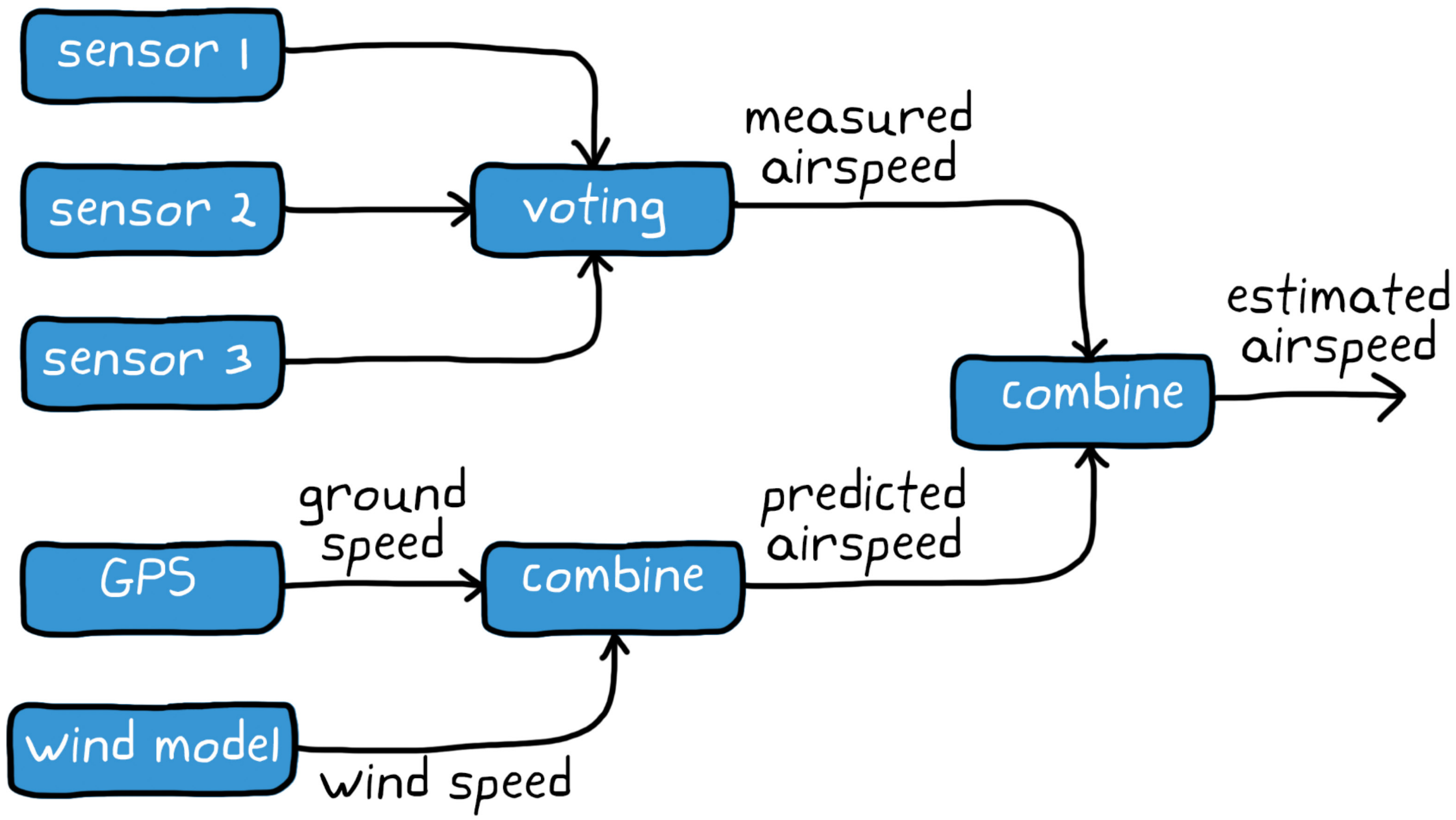
For an aircraft that needs a reliable measure of airspeed, an option could be to use three separate pitot tubes and set up a voting-based sensor fusion algorithm. That way if one breaks or reads incorrectly, that pitot tube is ignored and the airspeed is still known using the other two.

Duplicating sensors like this is an effective way to increase availability of unreliable sensors; however, we have to be careful of common mode failures. These are failures that can cause all of the sensors to fail at the same time or from the same cause. This is similar to correlated noise sources but a bit more catastrophic since it results in a failed sensor. An aircraft with redundant pitot tubes might find that all three freeze up when the aircraft flies through freezing rain; in that case, no amount of voting or sensor fusion will save the measurement.

Using Multiple Sensor Types … Again

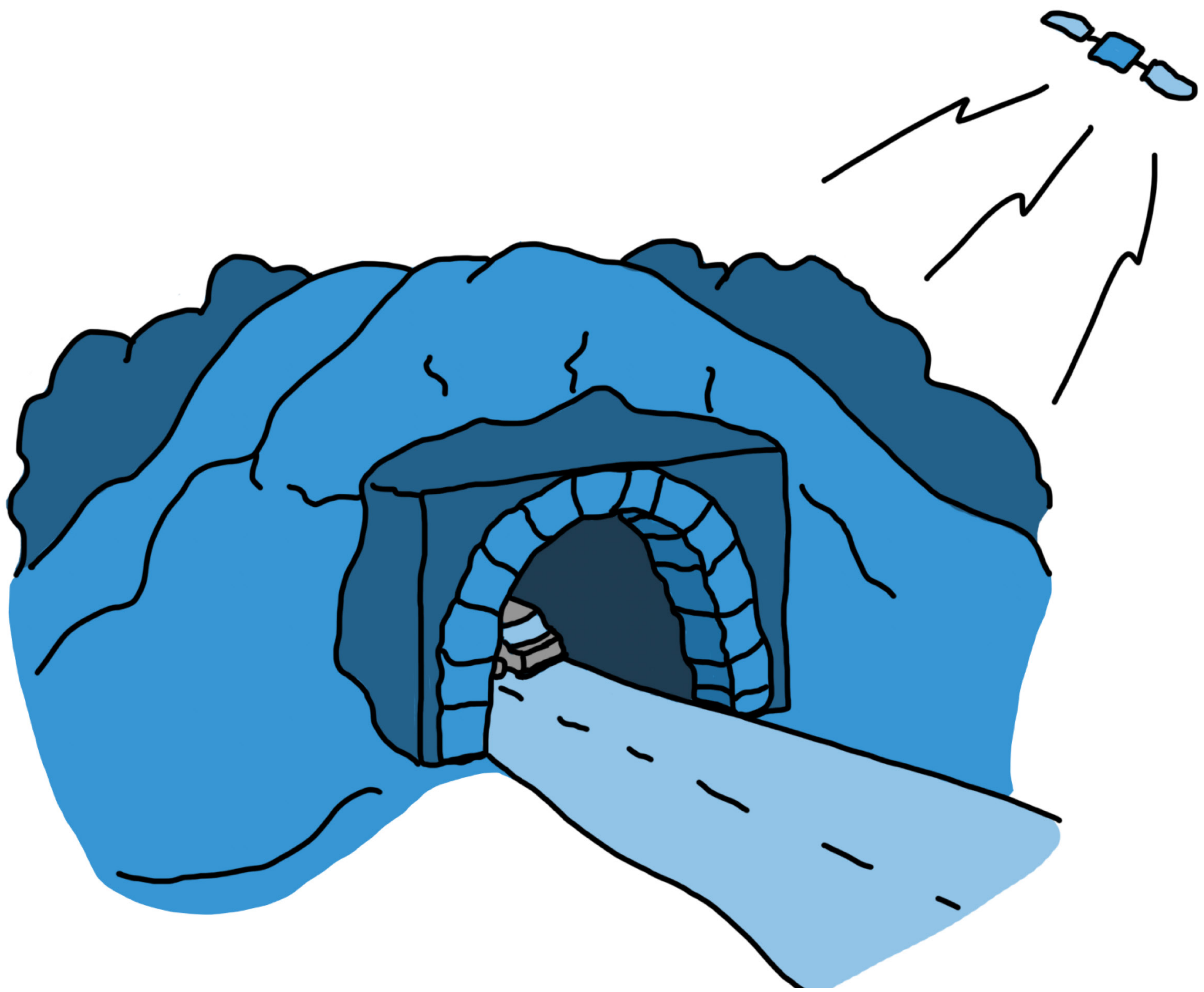
Protecting against common mode failures is where fusing sensors that measure different quantities can help once again. The aircraft could be set up to supplement the airspeed measurements from the pitot tubes with an airspeed estimate using GPS and atmospheric wind models. The idea is that if you know the aircraft ground speed using GPS, and you have a good estimate of high altitude air currents, then you can combine them to estimate airspeed.

This allows the aircraft to have an estimate of airspeed even if the primary sensor suite is unavailable. Note that data quality may be reduced since the wind speed model will probably have larger uncertainty than the pitot tubes, but airspeed can still be estimated, which is important for the safety of the aircraft.



Protect Against Loss of Signal

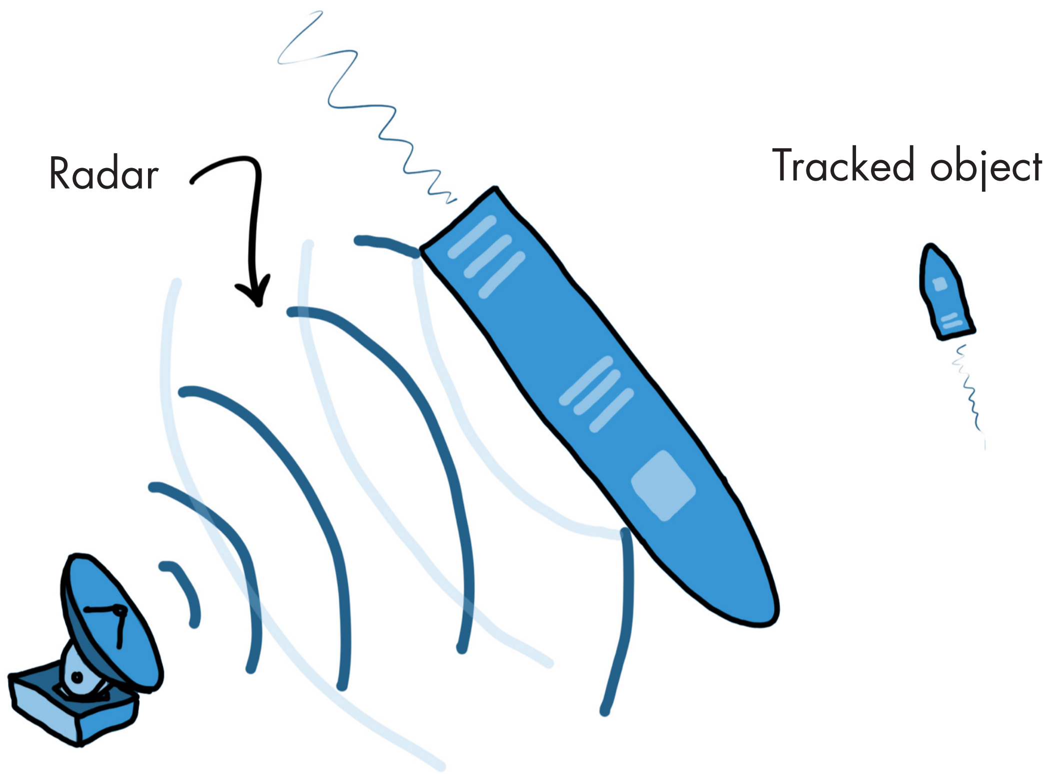
A sensor that fails to produce a measurement doesn't always mean the sensor itself failed. It could mean that the quantity it's measuring drops out momentarily. This is the case, for example, if a car travels through a deep tunnel while measuring position with GPS. The GPS satellite signals can't penetrate the thick walls of the tunnel, so the GPS sensor can't resolve a position.

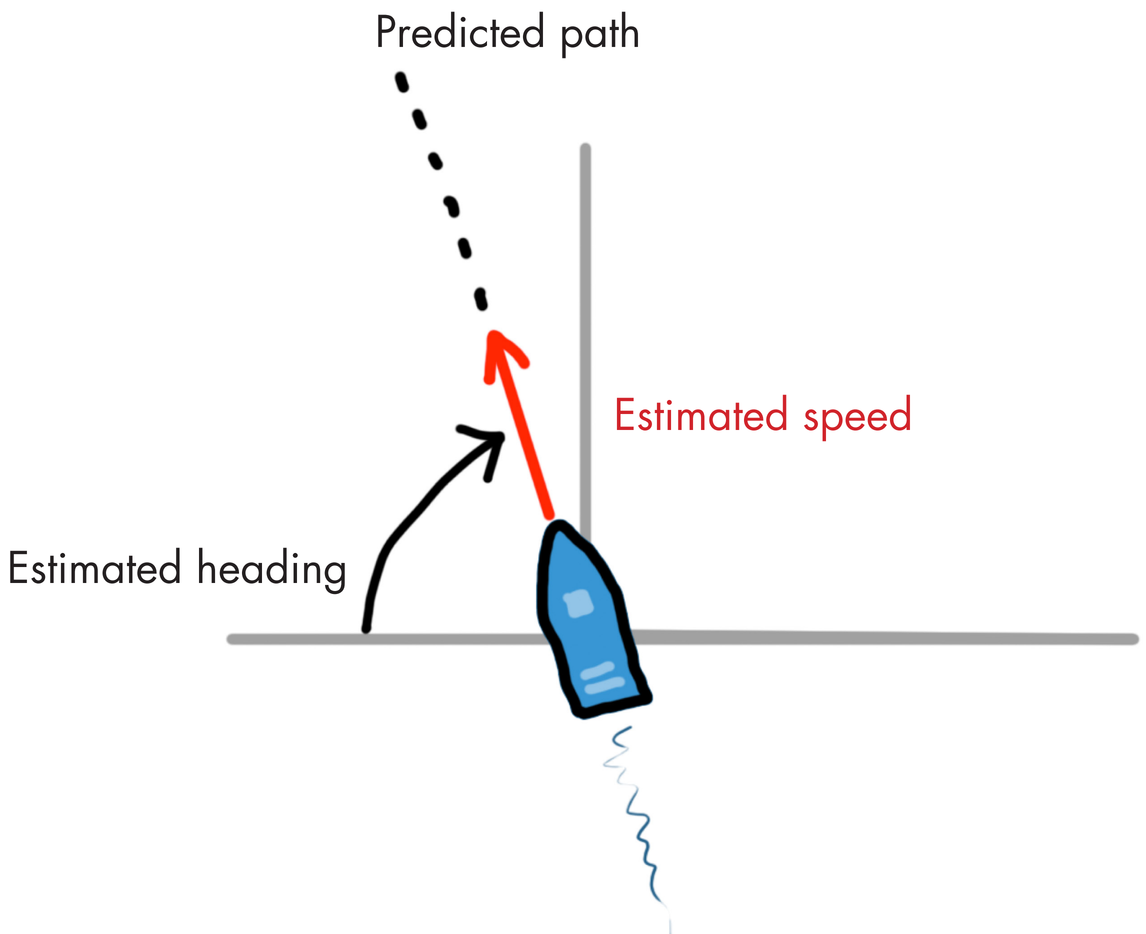


With sensor fusion, there are options available that allow the car to continue estimating position when the GPS signal has dropped out. For example, we could include other inertial sensors such as an accelerometer and a gyro, which measure how an object changes its position and orientation over time. We can integrate these values to estimate the position of the car and use this additional position estimate to supplement the GPS.

Protect Against Obstructed Line of Sight

A similar “loss of signal” situation can occur when tracking an object that becomes obstructed by something. An example is a radar system that is tracking the location of a small boat on the ocean. If a larger cargo ship or some other obstacle gets between the radar station and the smaller boat, the measurement will shift instantly to that of the blocking object. In this case, the radar doesn’t drop out; it just measures a different object than the one being tracked.



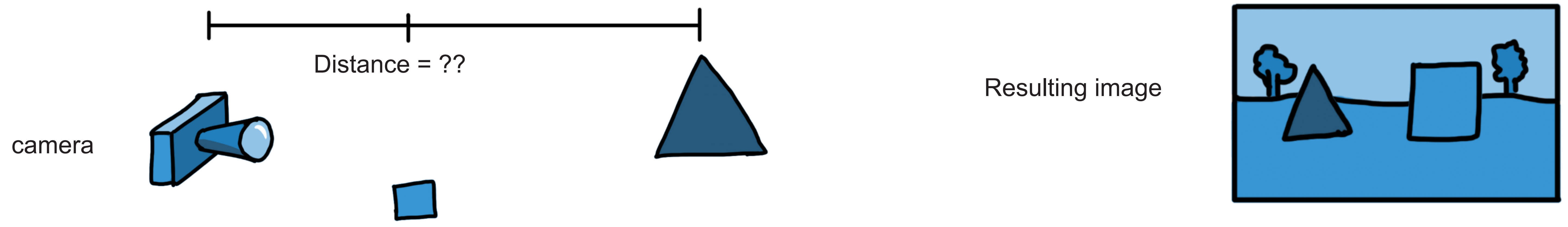


To protect against this, an algorithm could develop a speed and heading model of the object that’s being tracked and then when the object is out of the radar line of sight, the model can take over and make predictions of where it’ll be in the future.

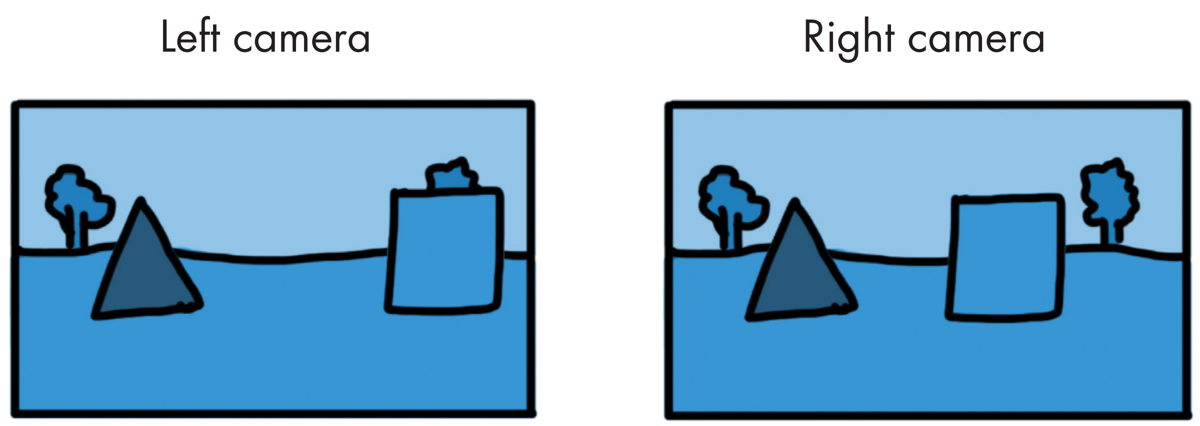
One benefit of comparing the measurement with a model is that we can use the model to challenge the measurement’s accuracy. This is the general idea behind the Kalman filter.

Estimating Unmeasured States

A third benefit of sensor fusion is that we can use it to estimate unmeasured states. It's important to recognize that unmeasured doesn't mean unmeasurable; it just means that the system doesn't have a sensor that can directly measure the state we're interested in. For example, a vision camera can't measure the distance to an object in its field of view. A large object far away can take up the same number of pixels as a small, close object.

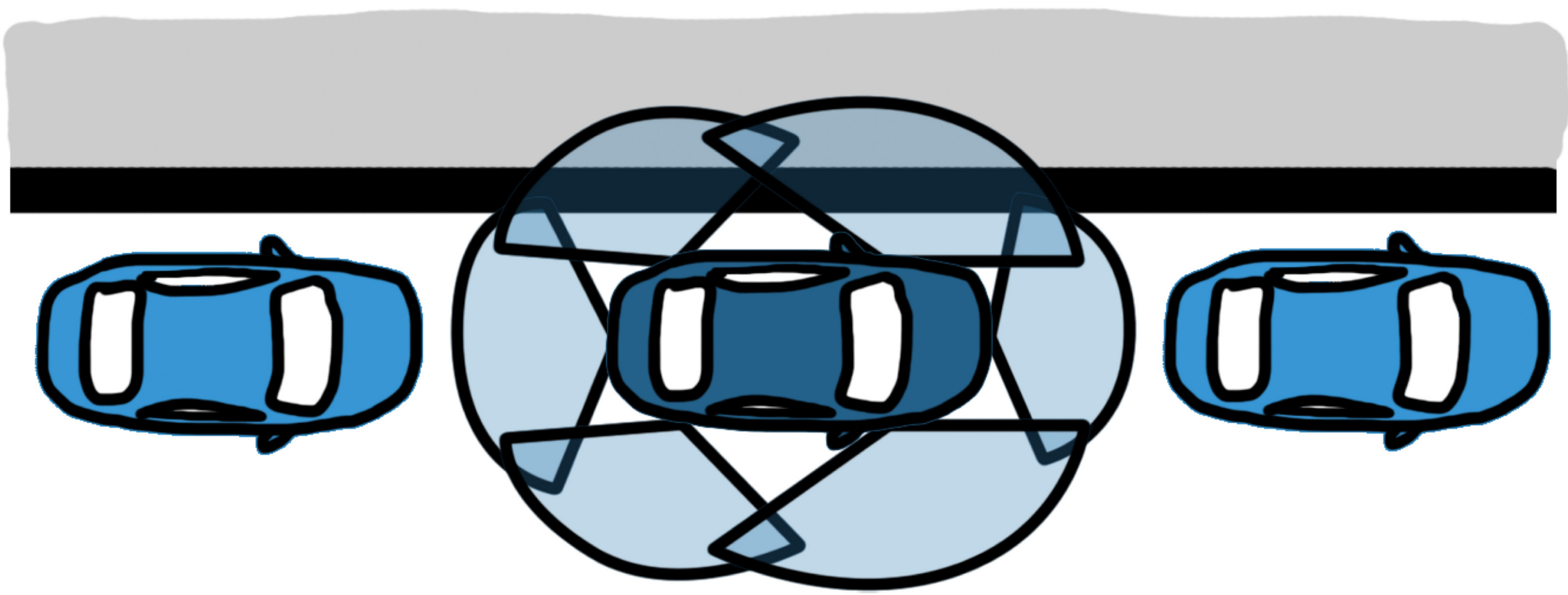


We can add a second camera and through sensor fusion, extract three-dimensional (3D) information. The fusion algorithm would compare the scene from the two different angles and measure the relative distances between the objects in the two images. By knowing the location and optical properties of the two cameras, distance can be inferred. These two sensors can't measure distance individually, but they can when combined in this way.



Increasing Coverage Area

Lastly, sensor fusion can be used to increase a sensor's effective coverage area. For this example, we'll use the short-range ultrasonic sensors on a car, which are used for parking assist. These are the sensors that are measuring the distance to nearby objects such as other parked cars and the curb to let you know when you're close to impact. Each individual sensor may have a range of only a few feet and a narrow field of view. Therefore, if the car needs to have full coverage on all four sides, additional sensors need to be added and the measurements fused to produce a larger total field of view.



These measurements won't necessarily be averaged or combined mathematically since it's usually helpful to know which sensor is registering an object so that you have an idea of where that object is relative to the car. But the algorithm that pulls all of these sensors together into one coherent system is a form of sensor fusion.

Conclusion

There are a lot of different ways to accomplish sensor fusion, and the approach can be applied across many different types of autonomous systems. Even though the methods don't necessarily share common algorithms or even have the same design objective, the general idea behind them is ubiquitous: use multiple data sources to improve measurement quality, reliability, and coverage and to estimate states that aren't measured directly. Plus, we can apply sensor fusion to help us estimate the state of a system that we have control over, as well as track one or more remote objects.

Learn More

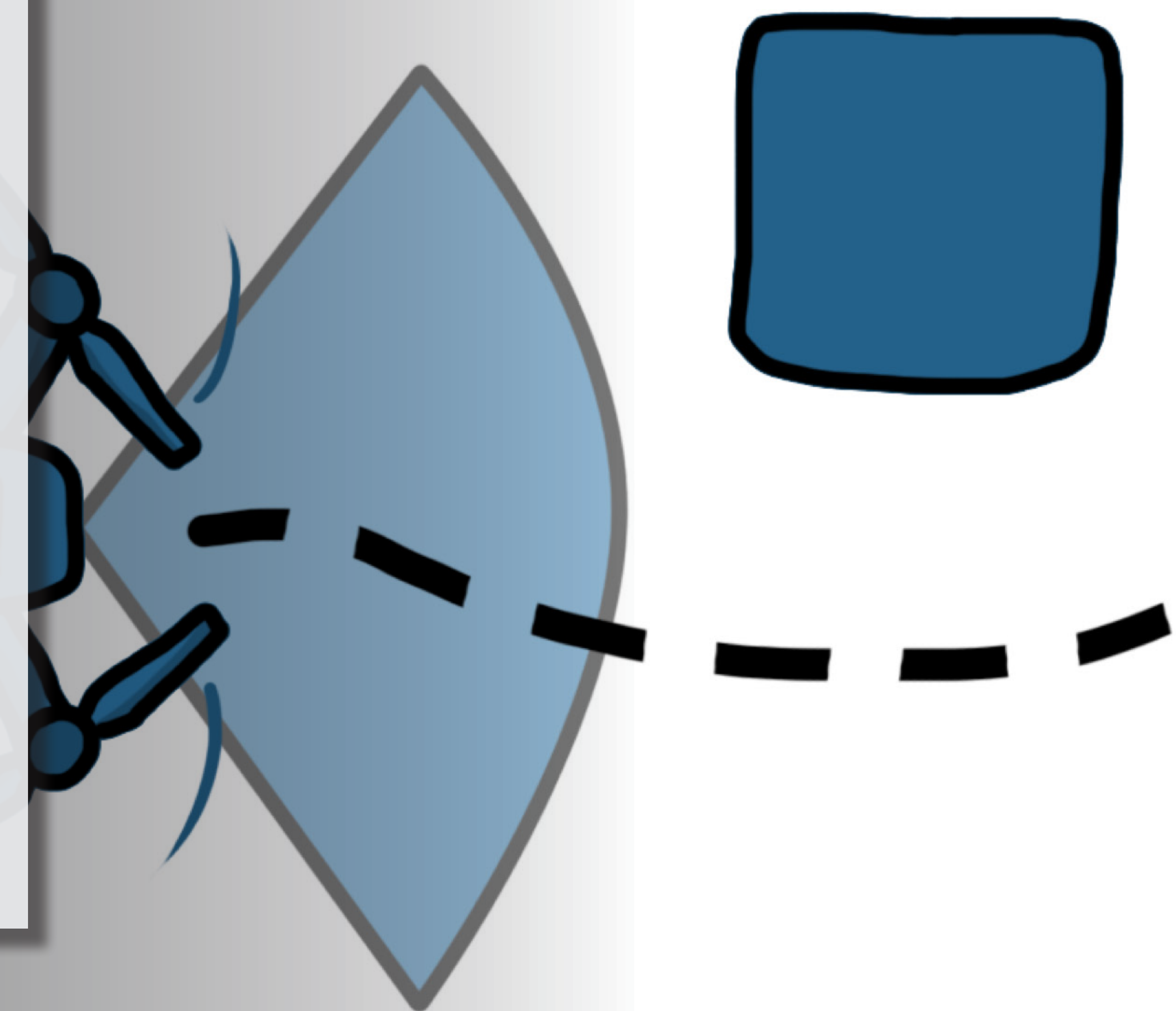
Ready for a deeper dive? Explore these resources to learn more about sensor fusion and get started.

Read

Sensor Fusion Toolbox - Overview
Sensor Fusion and Tracking for Autonomous Systems - White Paper
Getting Started with Sensor Fusion and Tracking Toolbox™ - Quick Start Guide

Watch

Understanding Sensor Fusion and Tracking - Video Series



© 2020 The MathWorks, Inc. MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See mathworks.com/trademarks for a list of additional trademarks. Other product or brand names may be trademarks or registered trademarks of their respective holders.