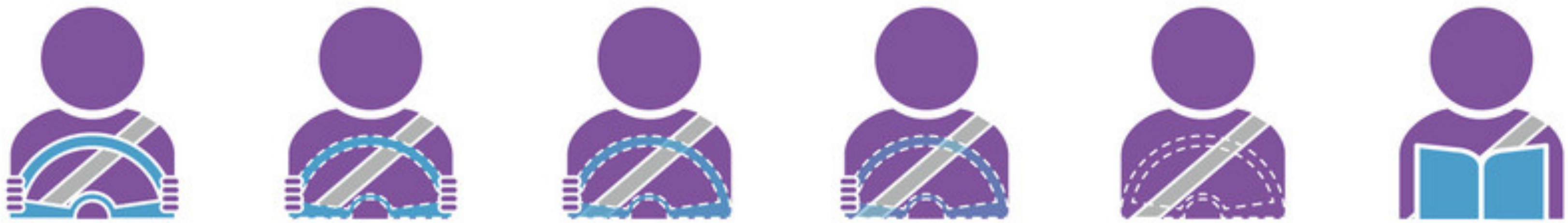
# LEVELS OF DRIVING AUTOMATION



| 0 | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| **NO AUTOMATION** | **DRIVER ASSISTANCE** | **PARTIAL AUTOMATION** | **CONDITIONAL AUTOMATION** | **HIGH AUTOMATION** | **FULL AUTOMATION** |
| Manual control. The human performs all driving tasks (steering, acceleration, braking, etc.). | The vehicle features a single automated system (e.g. it monitors speed through cruise control). | ADAS. The vehicle can perform steering and acceleration. The human still monitors all tasks and can take control at any time. | Environmental detection capabilities. The vehicle can perform most driving tasks, but human override is still required. | The vehicle performs all driving tasks under specific circumstances. Geofencing is required. Human override is still an option. | The vehicle performs all driving tasks under all conditions. Zero human attention or interaction is required. |

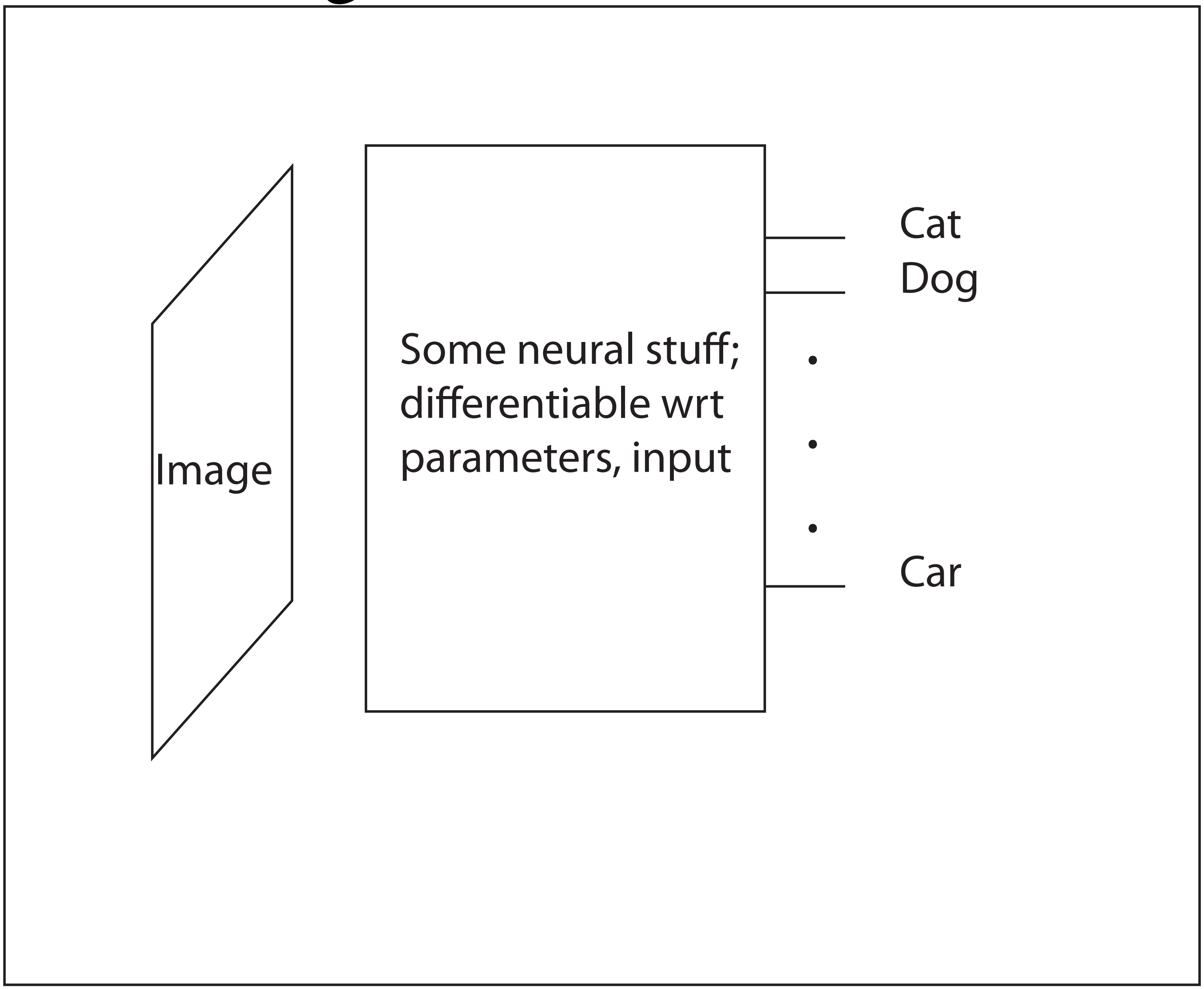
THE HUMAN MONITORS THE DRIVING ENVIRONMENT (levels 0–2)

THE AUTOMATED SYSTEM MONITORS THE DRIVING ENVIRONMENT (levels 3–5)

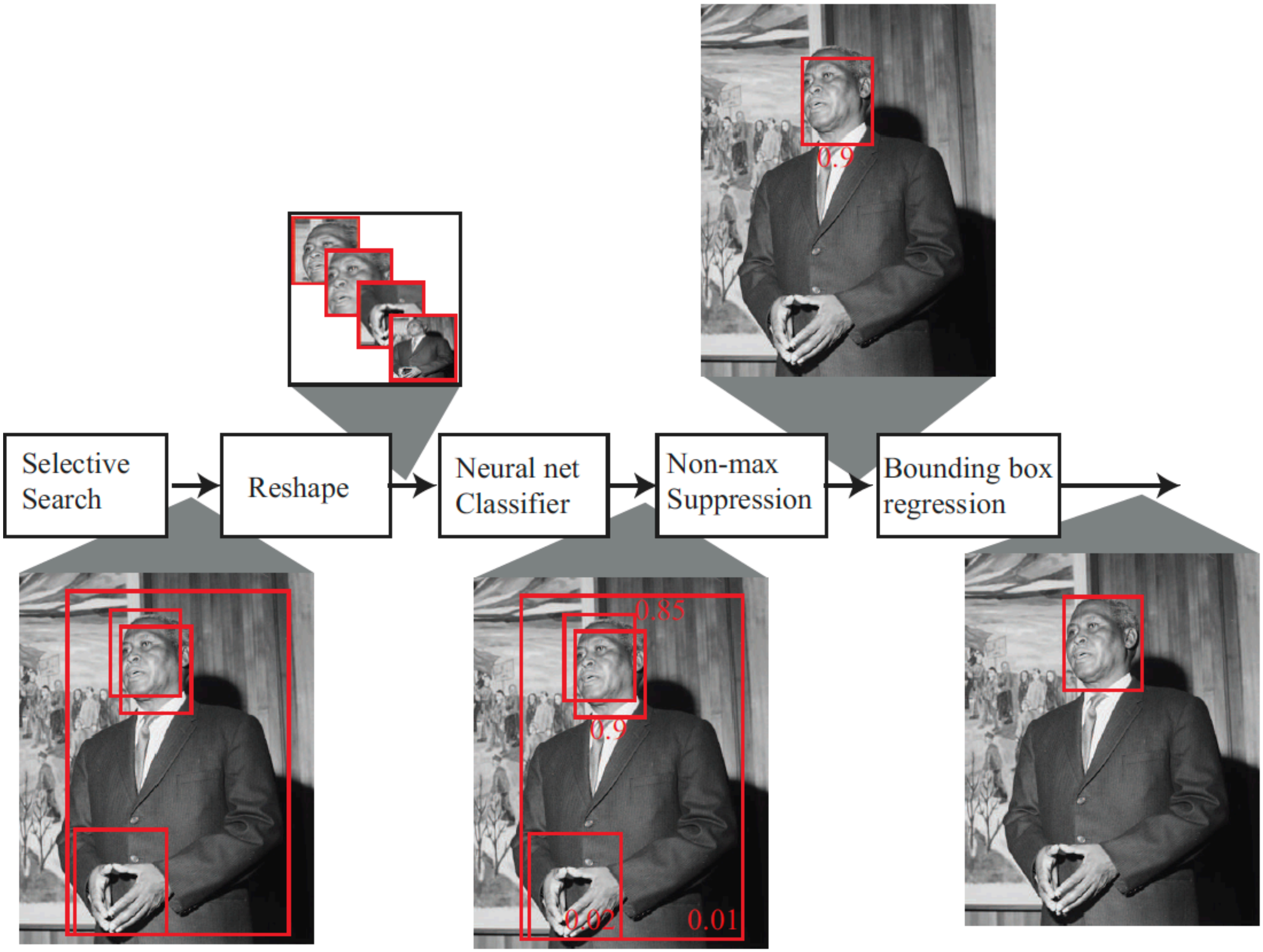
# Technologies we will look at

- Simple vision
  - Classification; Detection
- Registration
  - what transformation takes these data points to those?
- Simple control
  - how do you provide a system with inputs so that it does what you want?
- Tracking and Filtering
  - using a position estimate AND a motion estimate optimally

# Image Classification



# Detection



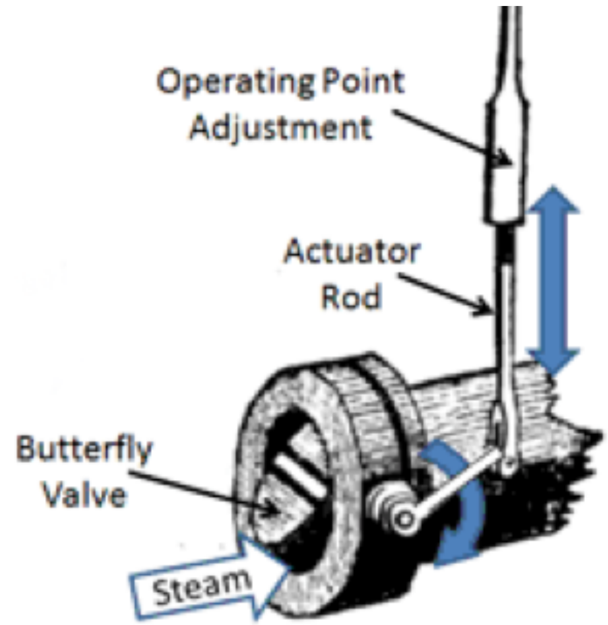
0.9
Selective Search
Reshape
Neural net Classifier
Non-max Suppression
Bounding box regression
0.85
0.9
0.02
0.01

# Registration

- Where am I?
  - Simplest: register observations and motion to a map
    - correspondence and robustness
- Build a map
  - Register observations to one another
    - global consistency
- Incorporating motion models
  - Registration should benefit from knowledge of motion
    - Filtering

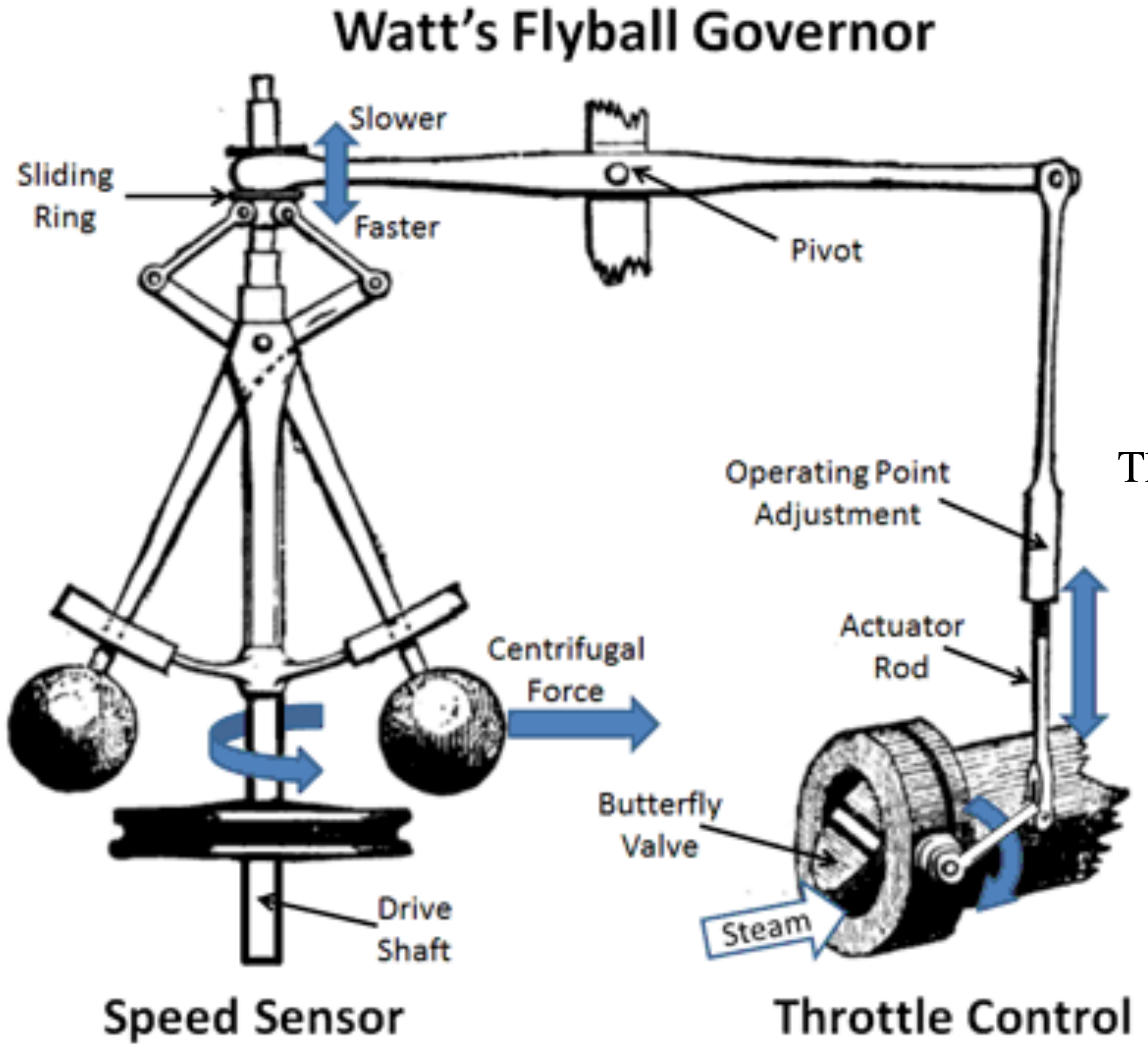
# Control

- Plant: K x(t) = c x(t)
  - here c is a known constant
- We'd like the output to be 1
  - feed plant with 1/c
    - and go home early
- Example of open loop control
  - compute a fixed input and supply to plant
    - whatever the plant
- Advantages:
  - simple, sometimes works
- Disadvantages:
  - what if your model is wrong?



Throttle Control

# Feedback




Watt's flyball governor, C19

These were still in use in late C20!

Q: what to feedback to get good behavior?

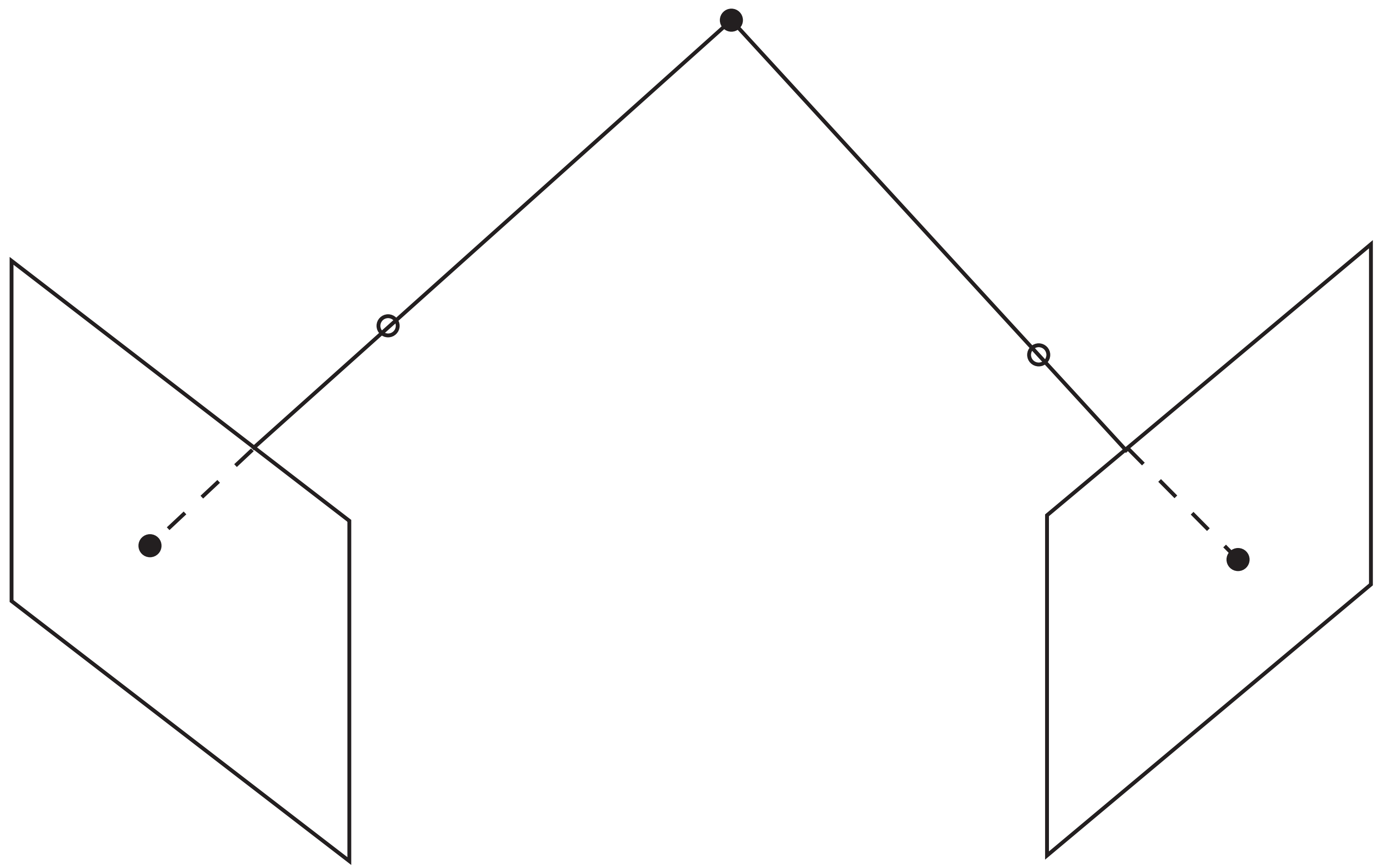
# Tracking and filtering

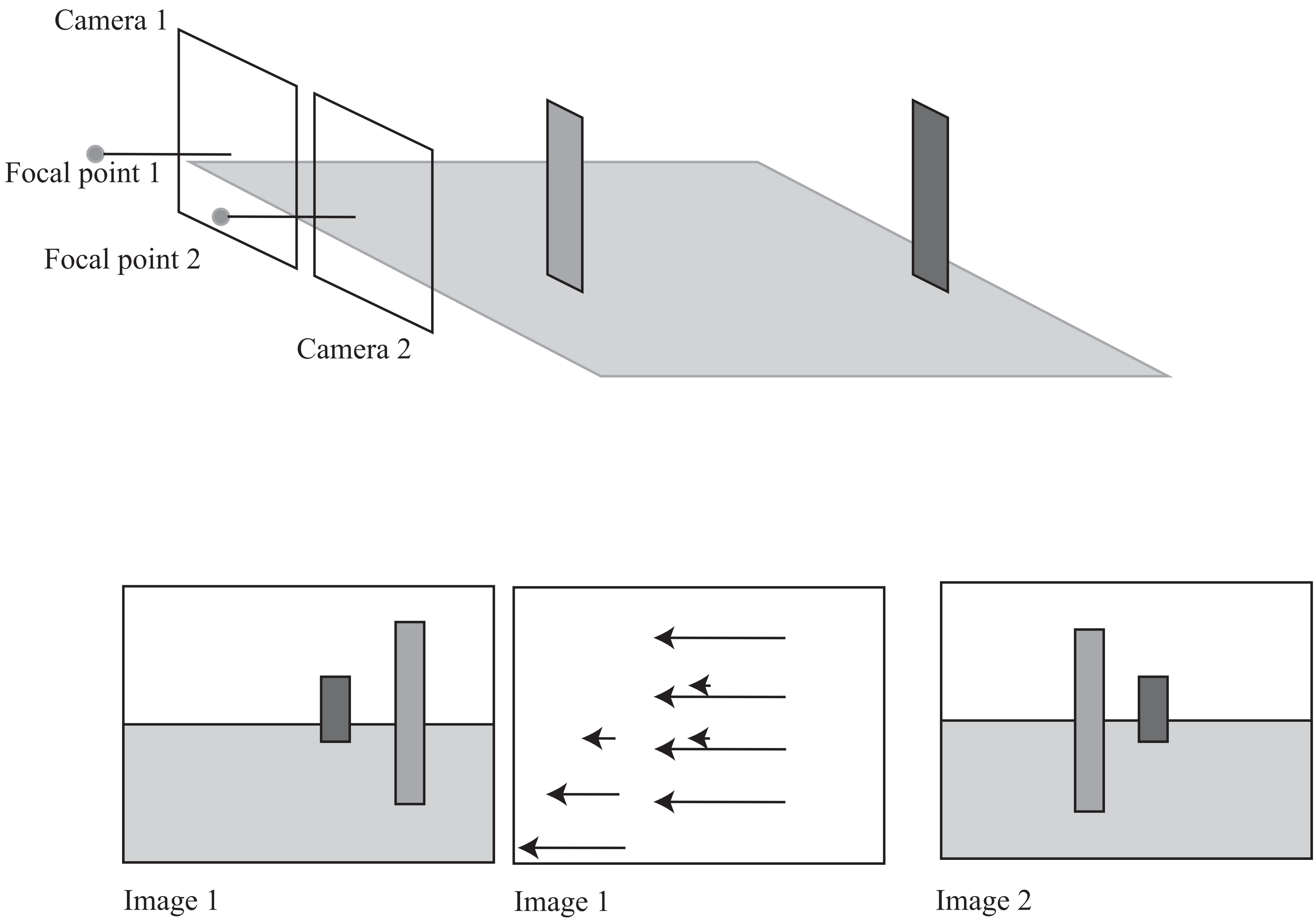
- Detect an object in frame i
  - known detection error model
  - known motion model, with known error properties
- Q: where will it go in frame i+1?
  - prediction
- Q: what is best estimate of location in i+1?
  - correct prediction using measurement

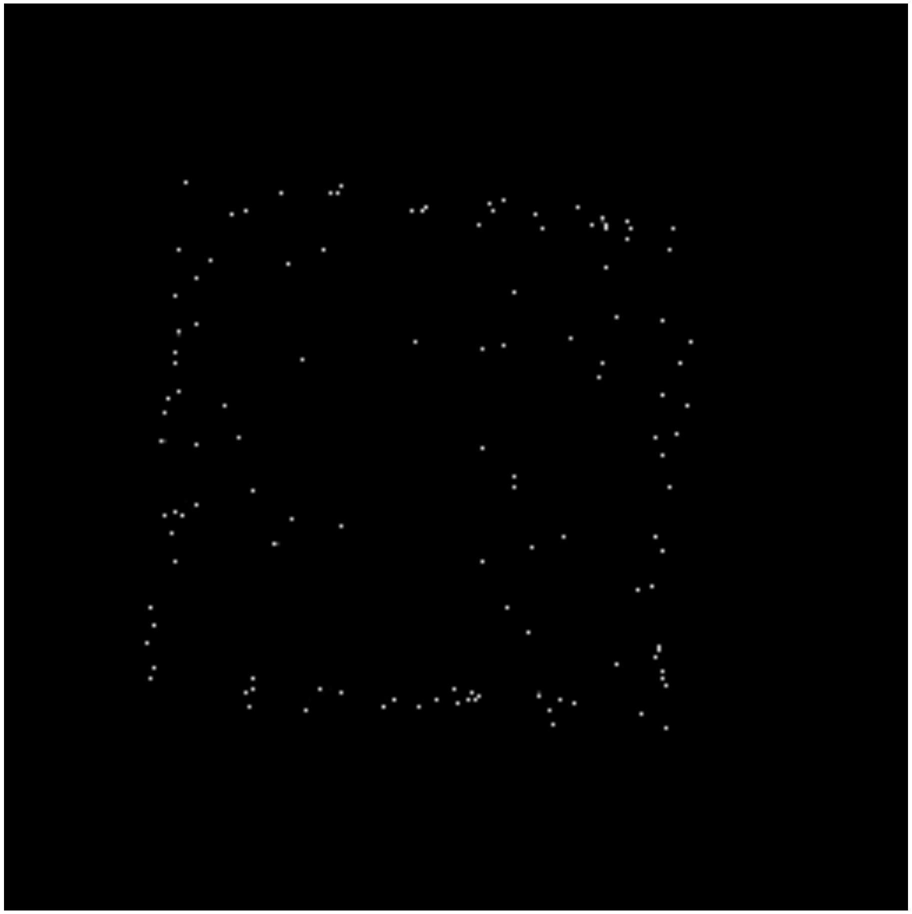
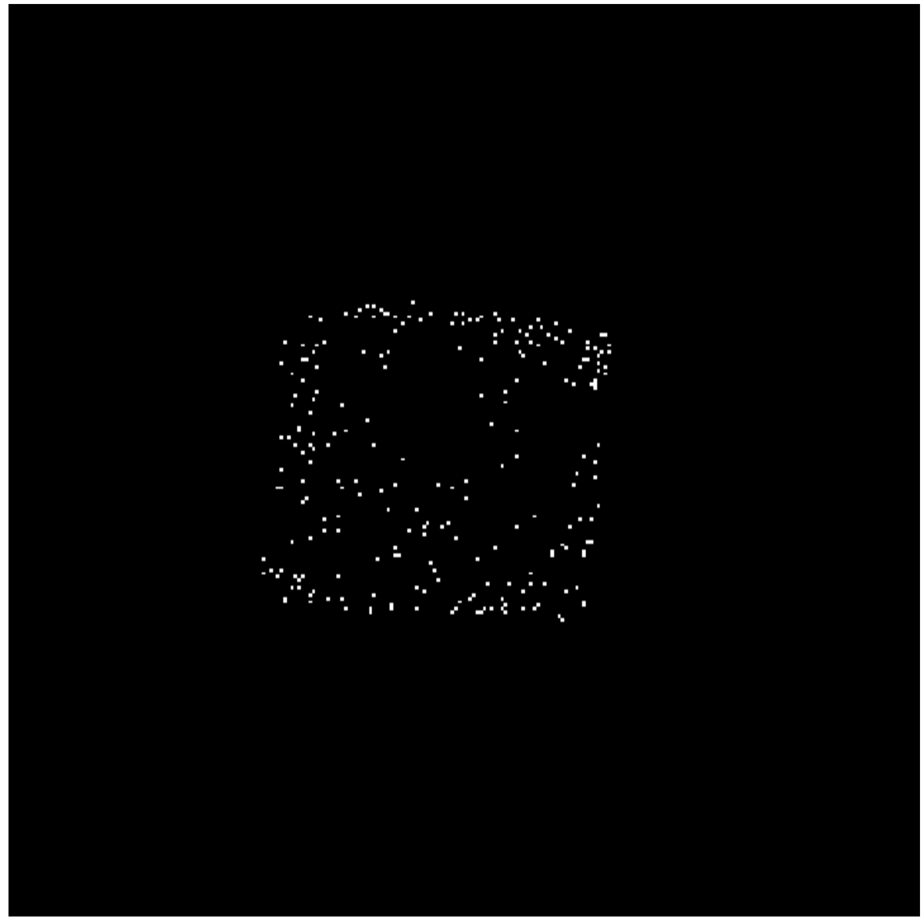
# Technologies we will look at

- Harder vision
  - depth and normals
  - multiple cameras
  - visual odometry
  - stereo and structure from motion
  - SLAM
  - building and using environment models
  - weather
  - intrinsic images

# Harder vision



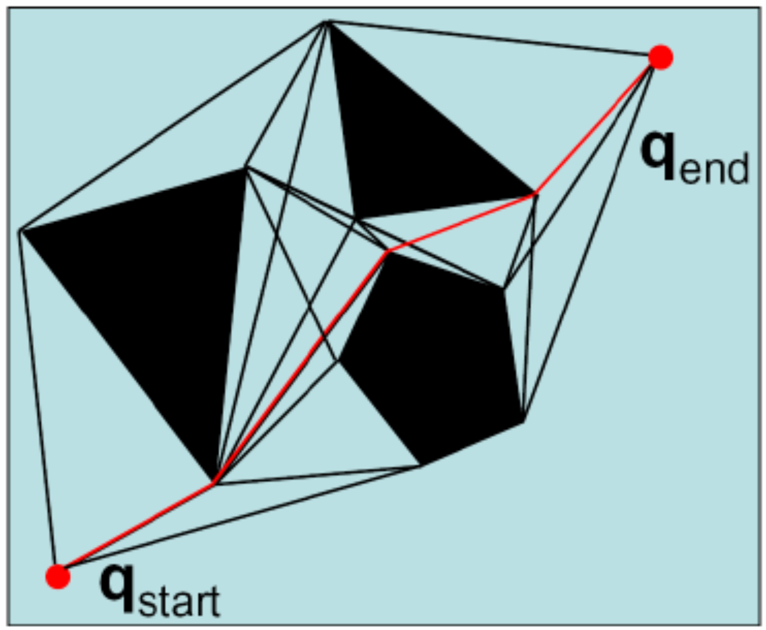

Camera 1
Focal point 1
Focal point 2
Camera 2
Image 1
Image 1
Image 2

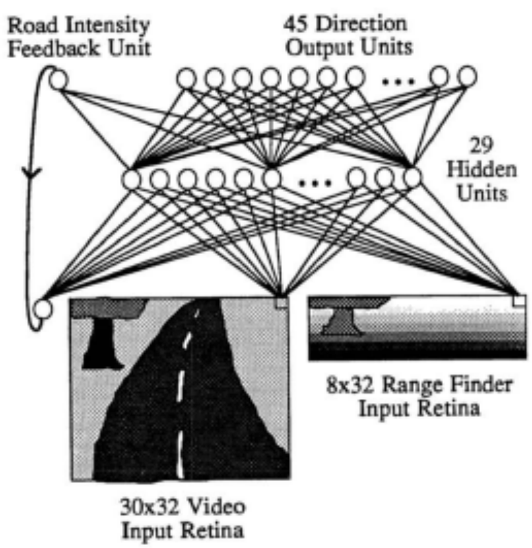
# Technologies we will look at

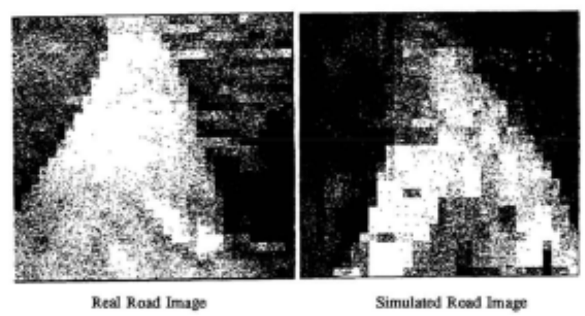
- Planning
  - what to do to get from here to there?
- Learning to control
  - imitation learning
  - reinforcement learning

# Planning



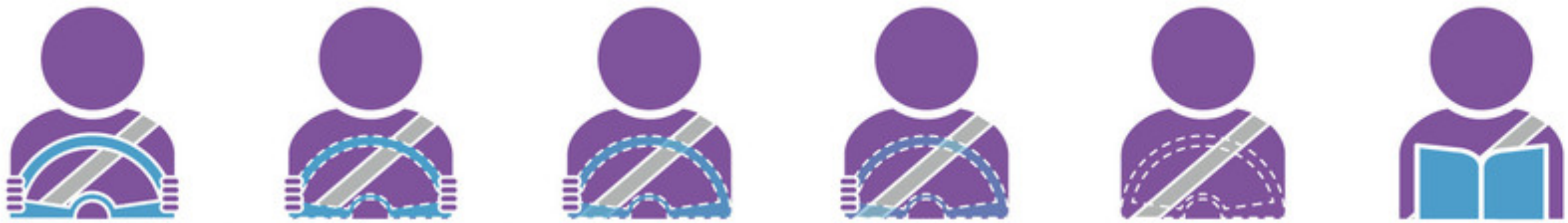
# Learning to control




*"In addition, the network must not solely be shown examples of accurate driving, but also how to recover (i.e. return to the road center) once a mistake has been made. Partial initial training on a variety of simulated road images should help eliminate these difficulties and facilitate better performance. " ALVINN: An autonomous Land vehicle in a neural Network, Pomerleau 1989*

# LEVELS OF DRIVING AUTOMATION



| 0 | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| **NO AUTOMATION** | **DRIVER ASSISTANCE** | **PARTIAL AUTOMATION** | **CONDITIONAL AUTOMATION** | **HIGH AUTOMATION** | **FULL AUTOMATION** |
| Manual control. The human performs all driving tasks (steering, acceleration, braking, etc.). | The vehicle features a single automated system (e.g. it monitors speed through cruise control). | ADAS. The vehicle can perform steering and acceleration. The human still monitors all tasks and can take control at any time. | Environmental detection capabilities. The vehicle can perform most driving tasks, but human override is still required. | The vehicle performs all driving tasks under specific circumstances. Geofencing is required. Human override is still an option. | The vehicle performs all driving tasks under all conditions. Zero human attention or interaction is required. |

THE HUMAN MONITORS THE DRIVING ENVIRONMENT (levels 0–2)

THE AUTOMATED SYSTEM MONITORS THE DRIVING ENVIRONMENT (levels 3–5)

# Context for vehicles and sensing

- Environments:
  - Case 1:
    - work anywhere
      - likely: various gadgets improve safety and experience
      - implausible: full autonomy
  - Case 1.5:
    - work some places
      - some are specialized to freeways, etc.
  - Case 2:
    - work only in tightly controlled environment (eg smart city)
      - there are models of full autonomy (eg transporters at airports)

# Case 1

- Various gadgets improve safety and experience
- case-by-case reasoning about representation and sensing
- Issues:
  - what's worth doing?
  - what can be done easily?
  - how much sensing?

# Case 1 examples

- Reversing cameras
- Reversing sonar
- Forward sonar for collision avoidance
- Active collision management
- Pedestrian detection
- Various safety cameras
  - driver attention
  - record events for dispute resolution
  - driver sobriety
- Smarter links to maps

# Case 1.5:

- Mostly, more specialized gadgets, mostly for highways
  - lane following for highways
  - predicting highway turnoffs
  - speed control that's aware of cars in front
  - neat tricks to reduce traffic jams
- Issues
  - what's worth doing?
  - what can be done easily?
  - how much sensing?

# Case 2: Strongly controlled environments

- Full autonomy quite plausible
  - depending on regulatory and environmental control
    - there are models of full autonomy (eg transporters at airports)
  - This case is valuable, and may be important
    - public transport -> apartment in high density living areas
- Issues:
  - how much control do you need?
  - what density of traffic can be sustained?
  - how do you ensure safe behavior if weird stuff happens?

# The questions that will plague us

- What representation do we need?
- How much data do we need to make it?
  - and where do we get it?
- How do we know if it works

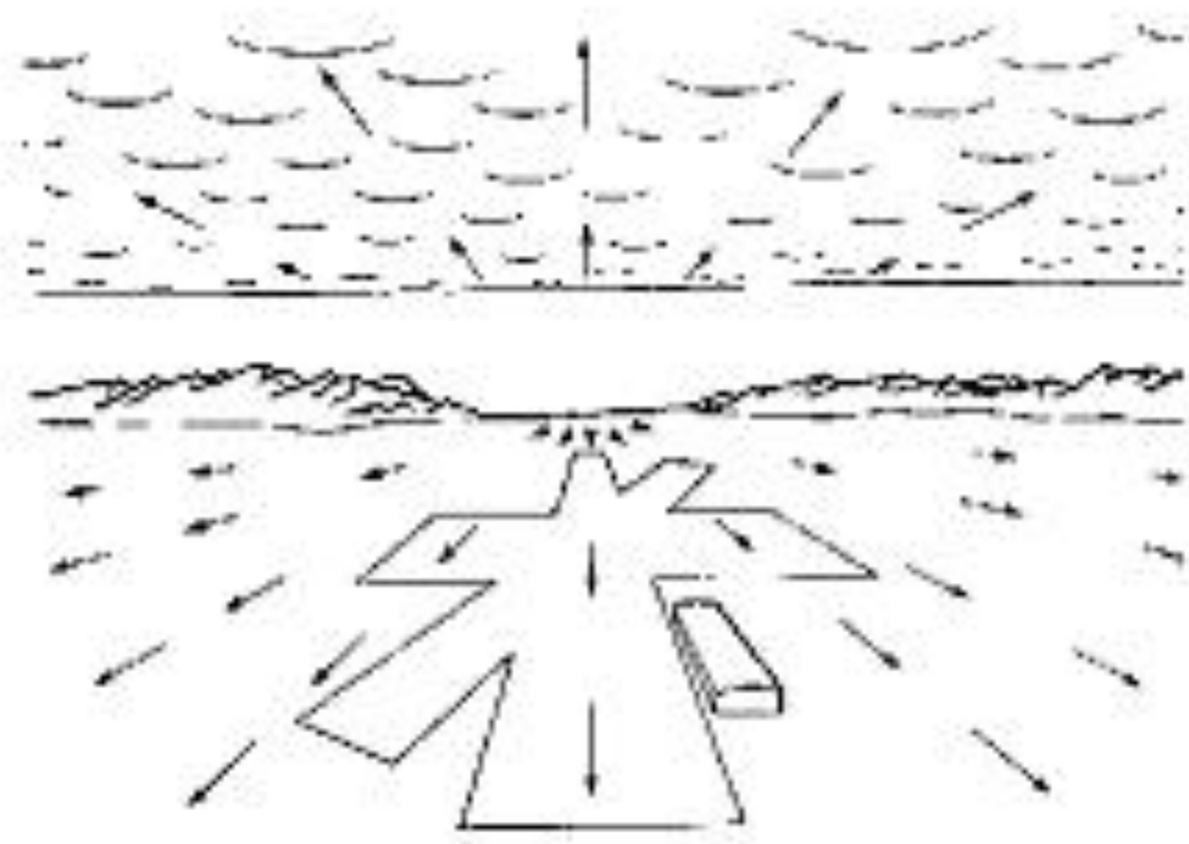
# Representation

- A1: No representation required
  - link control inputs to sensing with multiple network layers
  - train on simulation with reinforcement learning
  - dubious position, but…
    - notice that, IN PRINCIPLE, this deals with full autonomy
  - Q:
    - how do you know it will do the right thing in a given situation?
  - A (dubious)
    - watch what it does on training data

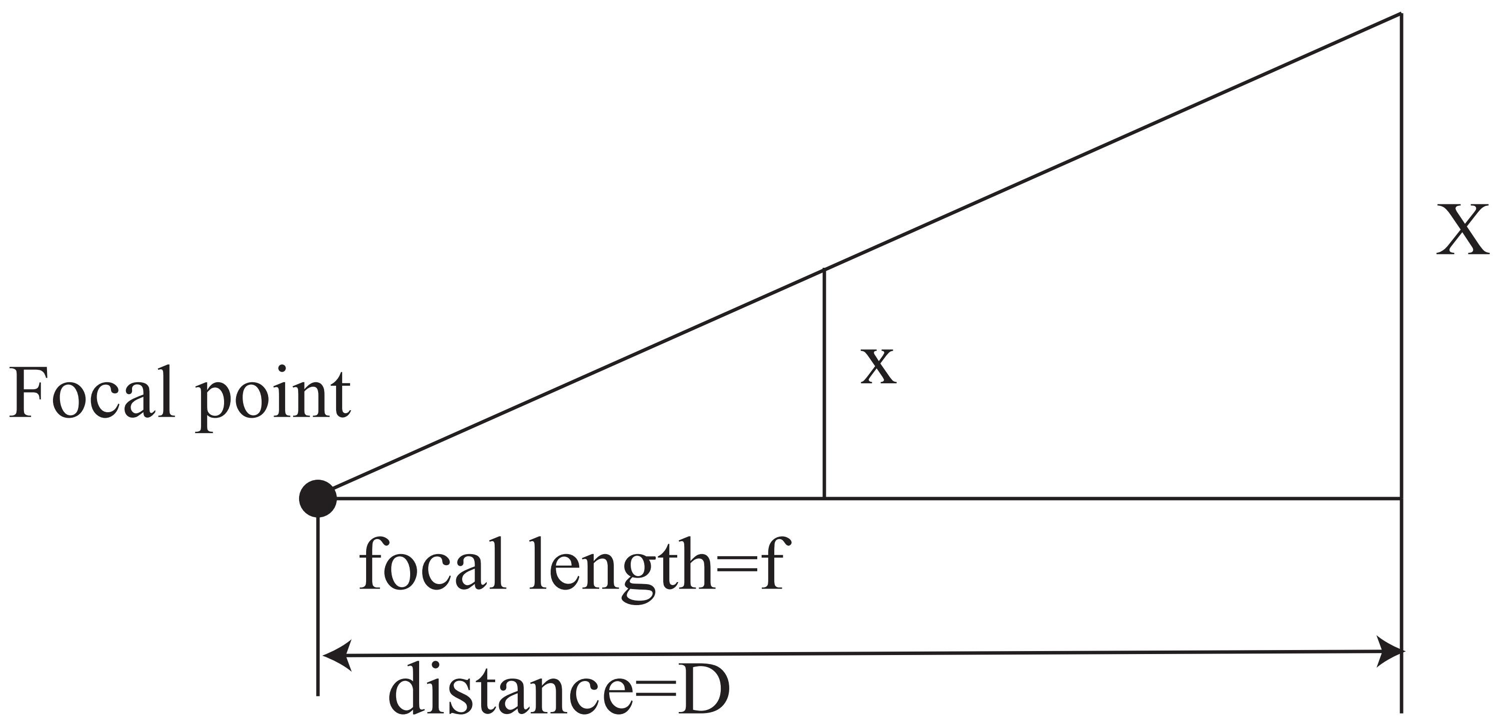
# Representation

- A2: 3D reconstruction
  - build complete 3D model of world around you
    - LIDAR, SFM, etc.
    - label it with appropriate labels (next slides)
  - use a planner, etc to make paths in that environment
  - follow paths
  - Q:
    - how do you know it will do the right thing in a given situation?
  - A (dubious):
    - prove that environment is right and software is correct
  - Q:
    - do you really need a 3D representation?
  - A:
    - who knows?

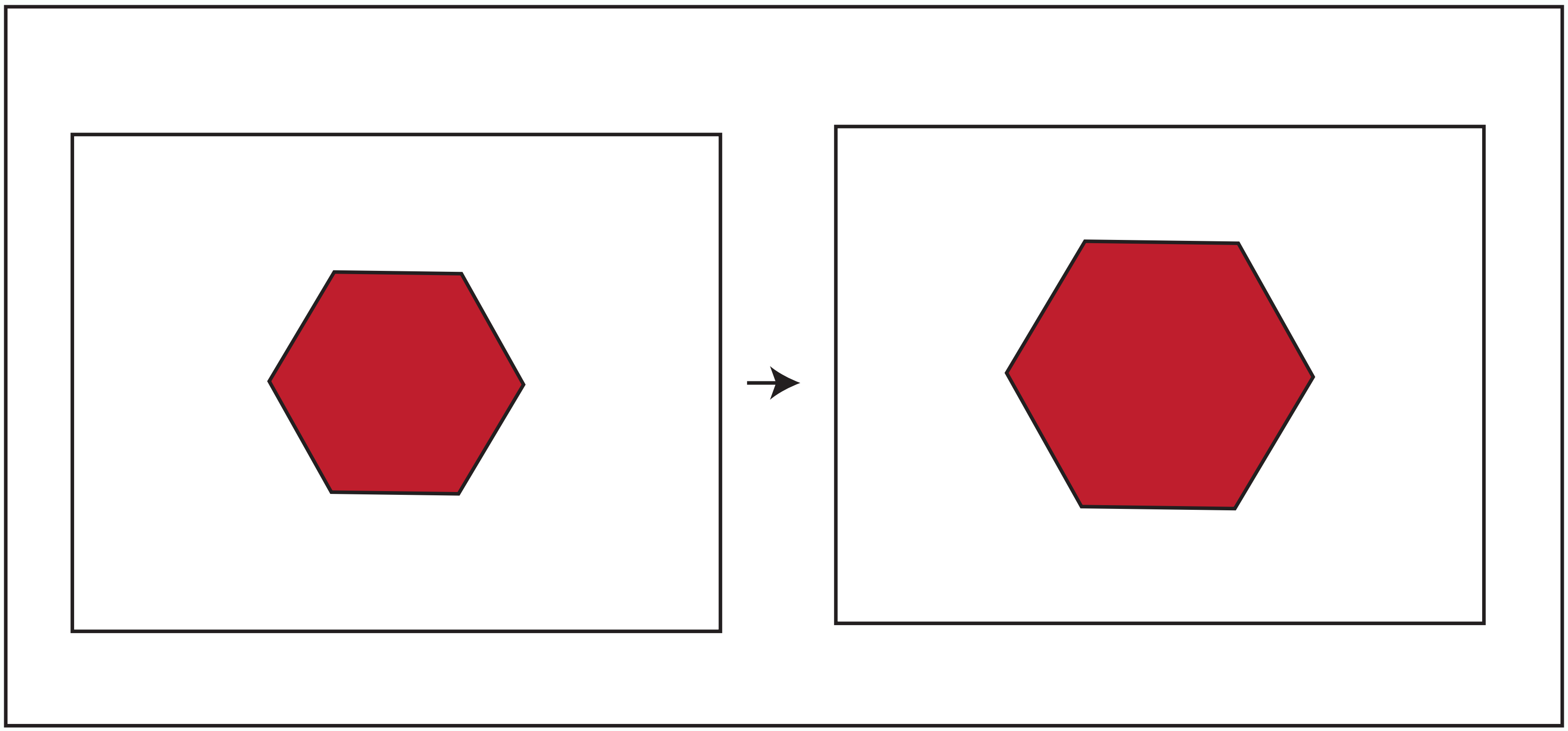
# Optic flow as a theory of perception



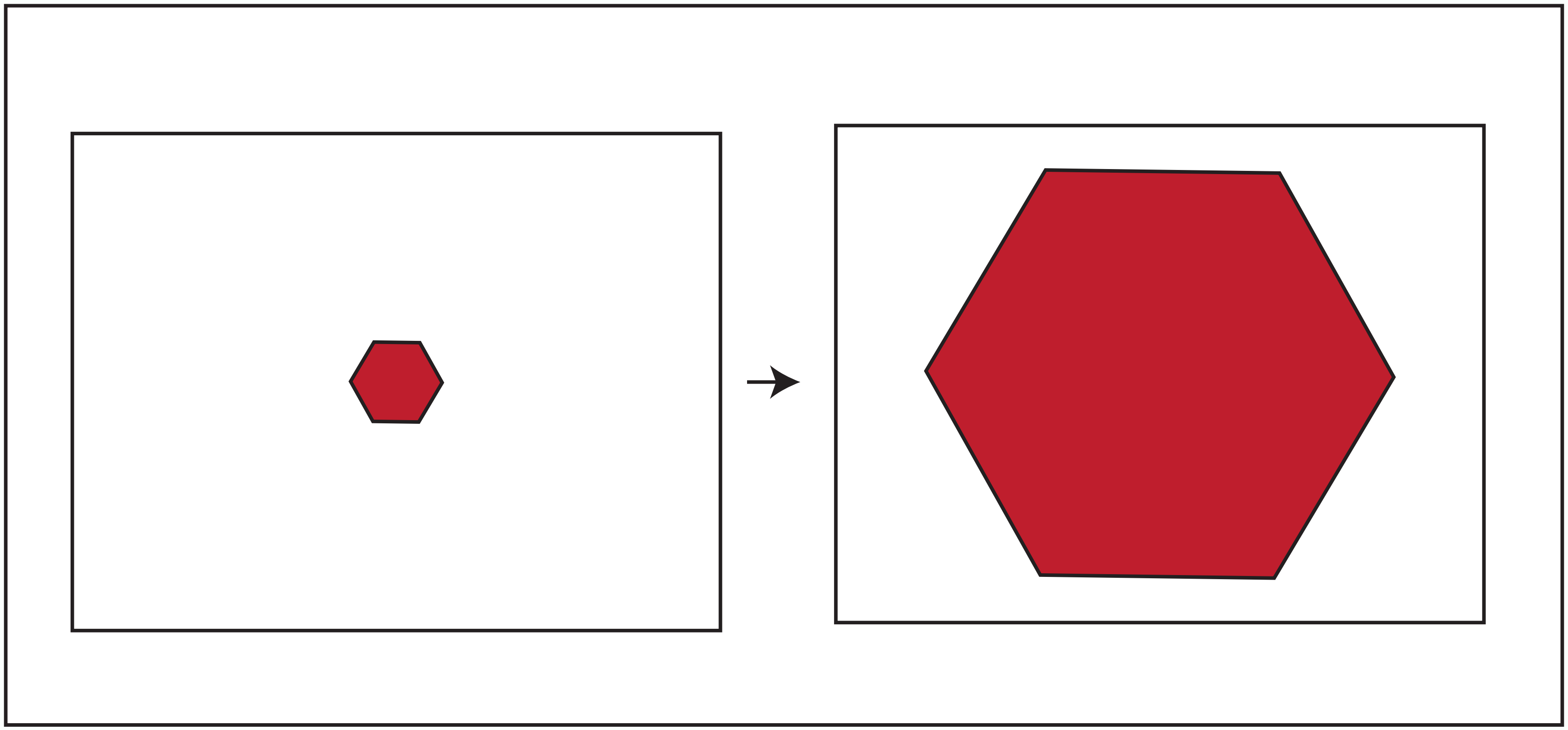
# Fun fact about vision



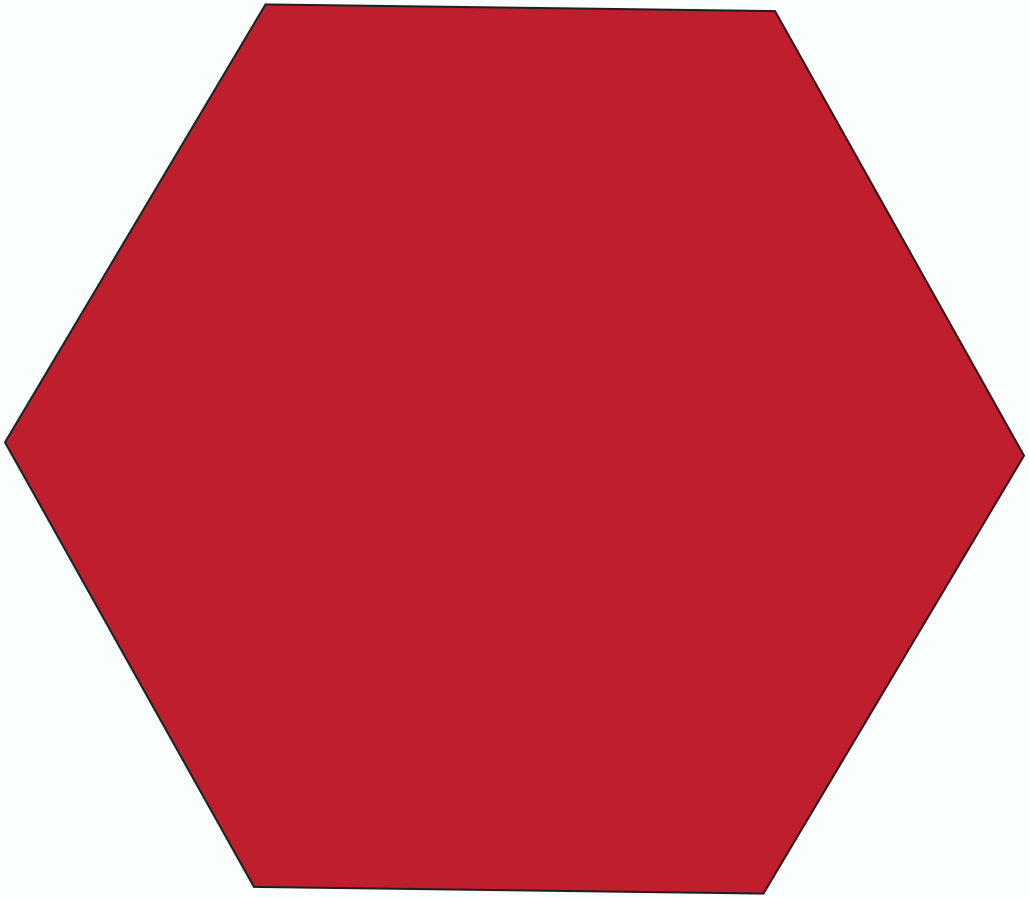

Fun fact: time to contact = x/(dx/dt)

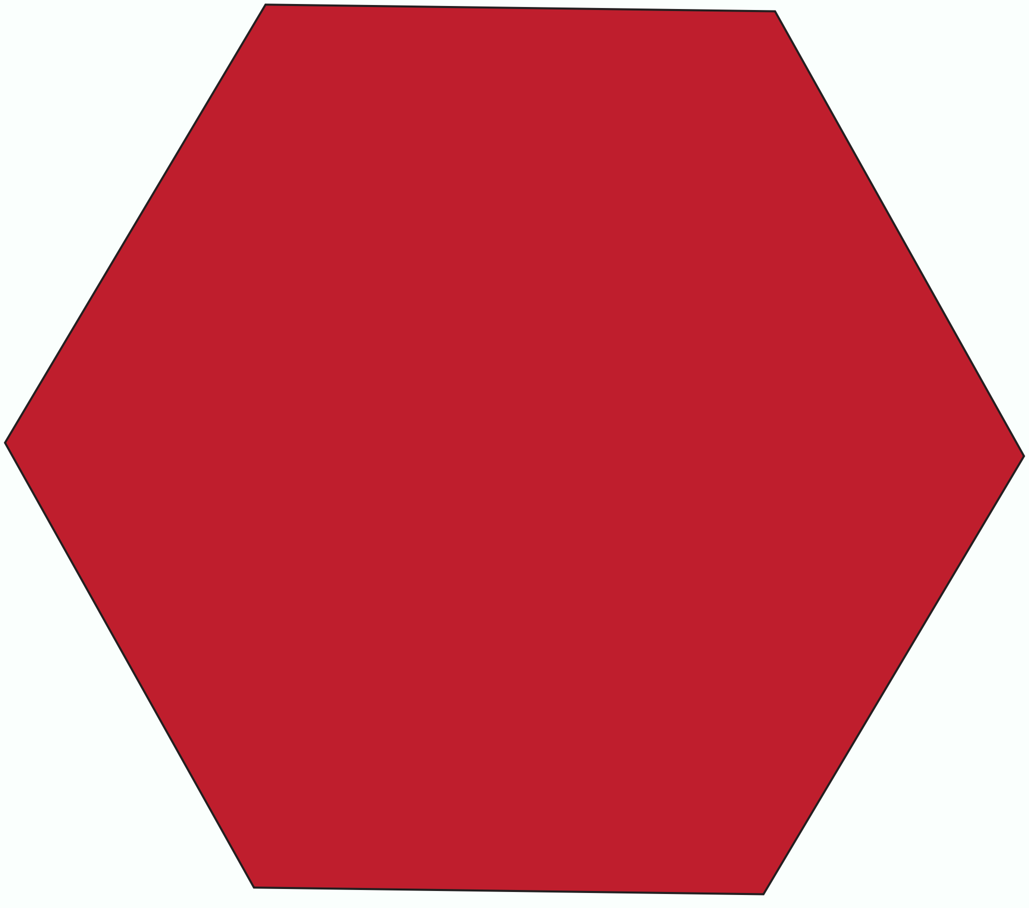


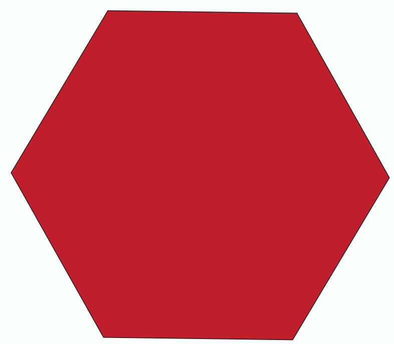
TTC - Long

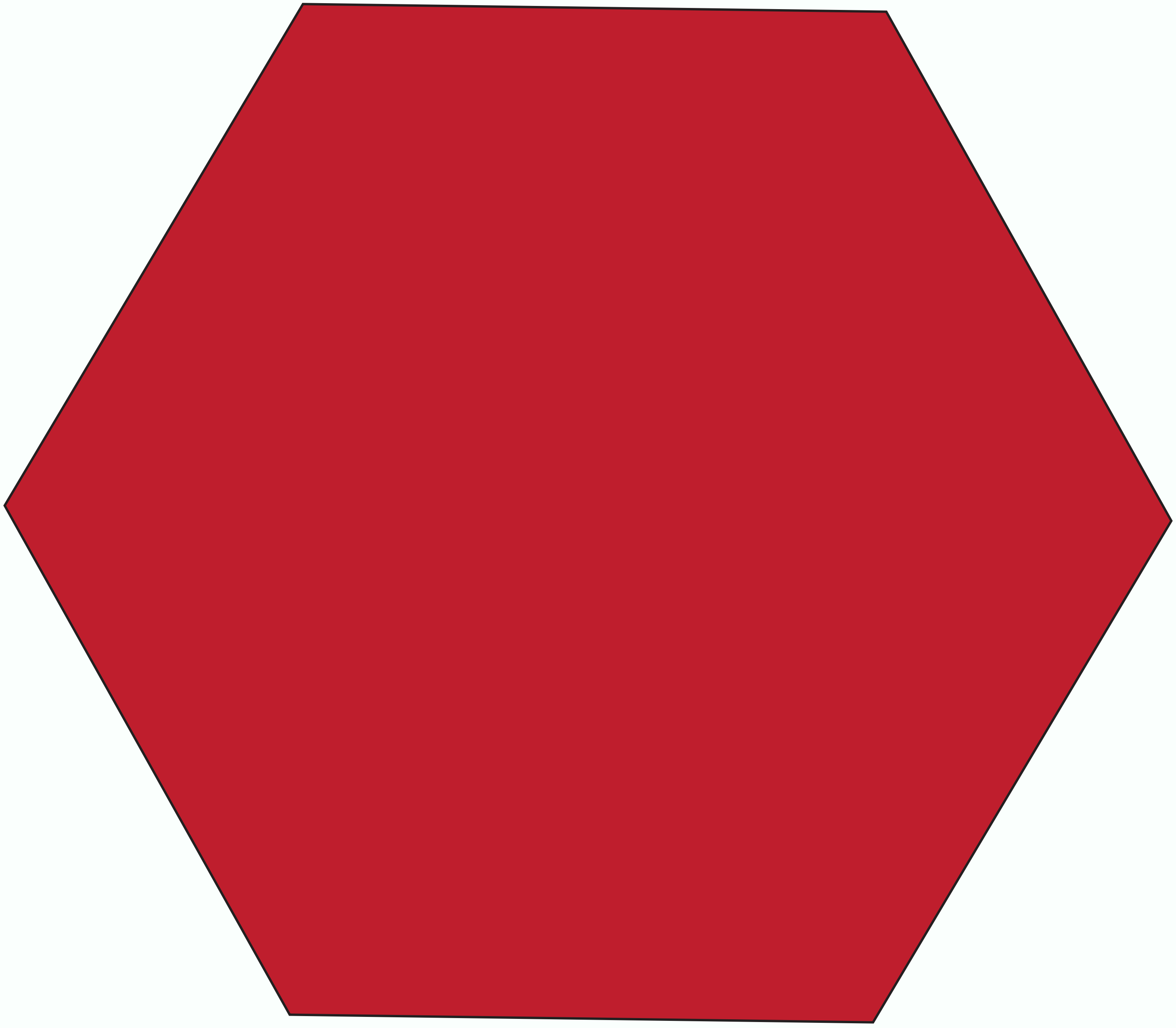


TTC - AAARGH!





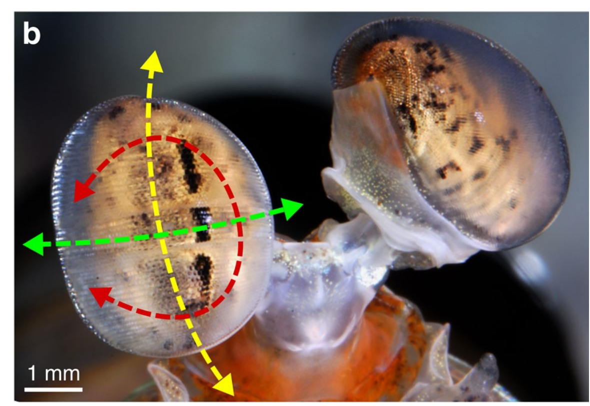




# Where am I?

- This doesn't get sufficient credit as 3D
  - early work (im2gps, etc; Hays+Efros 2008)
  - non-par regression (matching)
  - NOT the same as building a map

- Short scales, visually simple worlds are hard
  - get different visual sensors and use them well
    - Mantis shrimp (Daly et al 2016)



# How do I get home?

- Desert ants can forage, then go home directly
  - They're not doing SLAM! (scale)
  - Cues:
    - dead reckoning (count leg movements)
    - visual waypoints
    - polarization based sun compass
- Behavior can be explained *without* a map
  - multiple cues each produce a "go-home" vector
  - weighted combination  (Hoinville+Wehner, 2018)
  - can be imitated (Dupeyroux et al 2019)
- And they can go home backwards

https://www.labroots.com/trending/plants-and-animals/15376/desert-ants-sun-return-home-foraging

- Movie

https://www.youtube.com/watch?v=i-ahuZEvWH8

# What can I do?

- Path planning is not about geometric detail
  - which creates computational complexity
  - RRT methods; nearest neighbor methods; = strategies to duck detail
    - the key is a test: will this result in collision?
  - So why recover detail from images, rather than be able to answer query?
- We should recover geometric affordances of objects
  - what can be done to this, and where?
  - this likely isn't inherited from category
- Does a clam shell have a "hit here" tag?

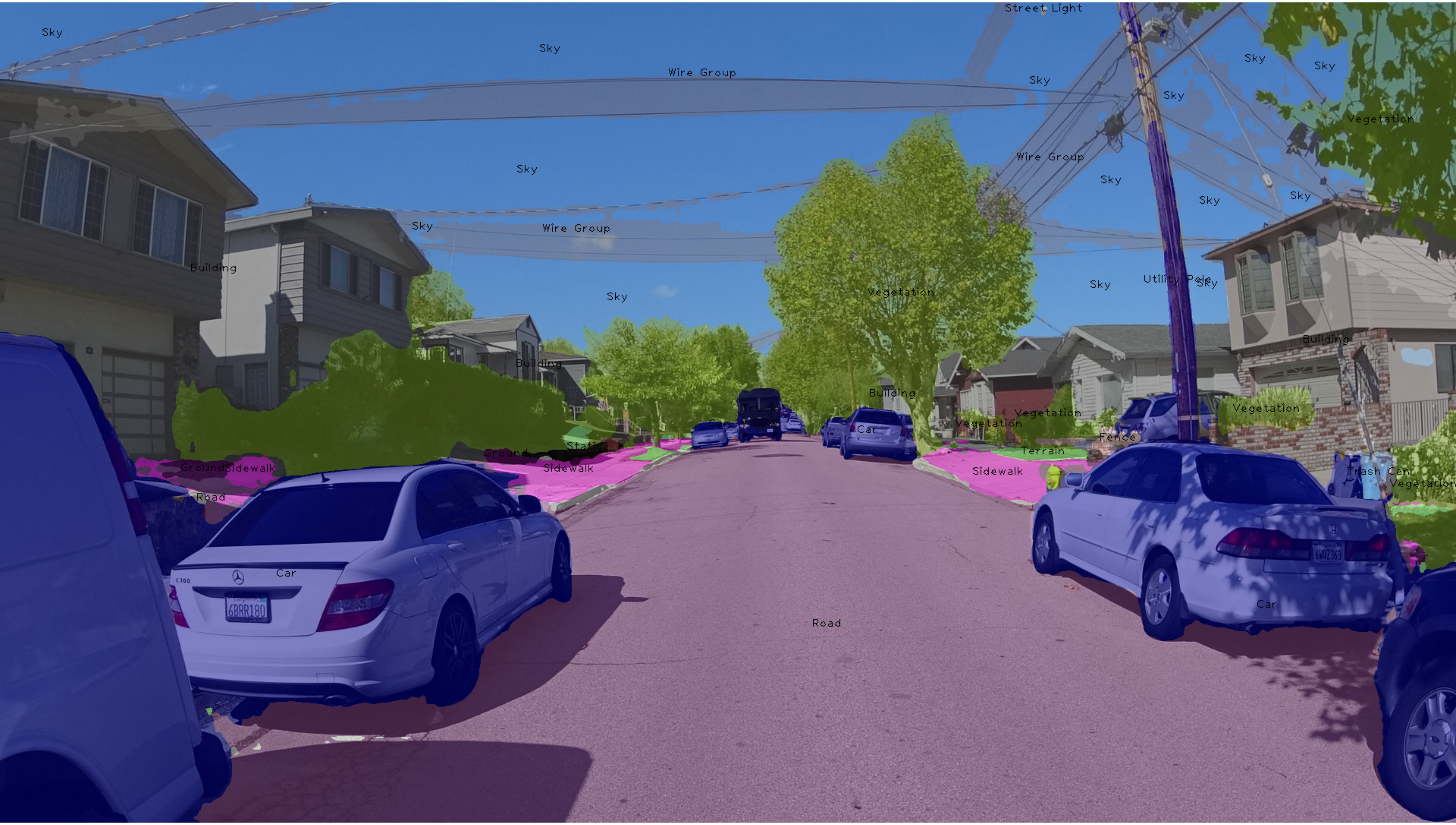
# Likely truth about 3D vs 2D

- Straightforward to convert from 2D to 3D repns and back
- This means anything you can do w/3D, you can do w/2D
- But: convenience is important
  - some planners want 3D
  - sensing 3D as 3D might be a good idea (LIDAR)
  - detection is generally faster in 2D, might be easier

# Representation

- A3: Label images (or 3D reconstruction)
  - with what?
    - label all possible objects with all names
    - label some classes, ignore others
  - what taxonomy?
    - likely a derived taxonomy from actions
  - Q:
    - how do you know it will do the right thing in a given situation?
  - A (dubious):
    - prove that environment is right and software is correct
  - Q:
    - what should be labelled and what should be ignored?
  - A:
    - who knows? likely the things that most affect performance?

# Labelling



Sky
Street Light
Sky
Wire Group
Sky
Sky
Sky
Sky
Vegetation
Wire Group
Sky
Sky
Sky
Sky
Wire Group
Sky
Building
Sky
Vegetation
Sky
Utility Pole
Sky
Building
Building
Building
Vegetation
Vegetation
Vegetation
Car
Fence
Terrain
Static
Ground
Sidewalk
Sidewalk
Sidewalk
Trash Can
Vegetation
Road
Car
Road
Car

# Labelling

MS-CoCo

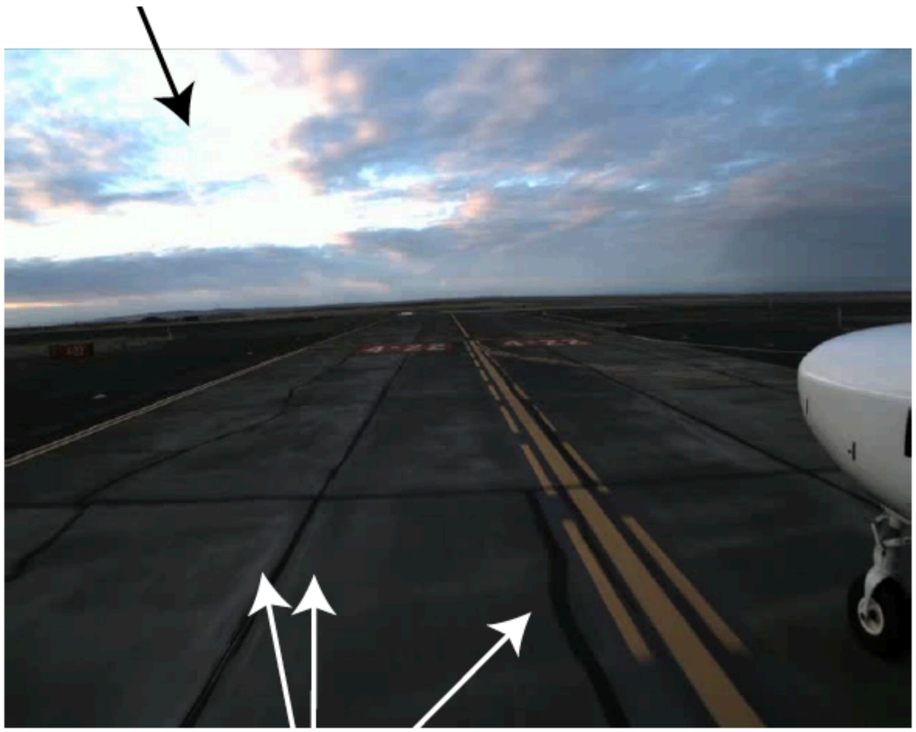
# Labelling

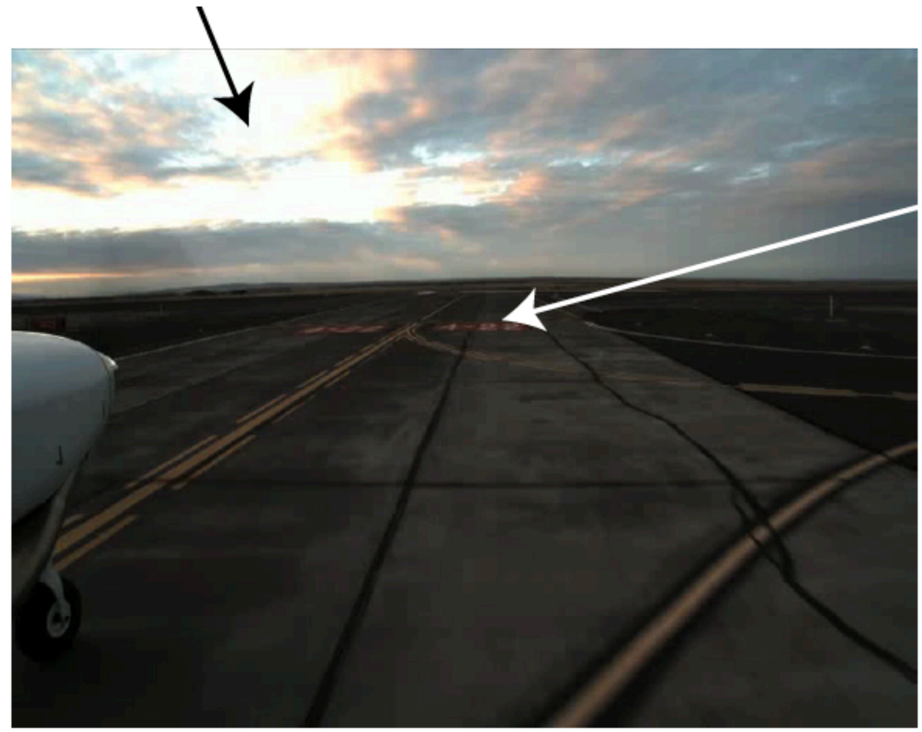
# The questions that will plague us

- What representation do we need?
- How much data do we need to make it?
  - and where do we get it?
- How do we know if it works?

# XXXX Autonomy data

# Special features: rich appearance variation

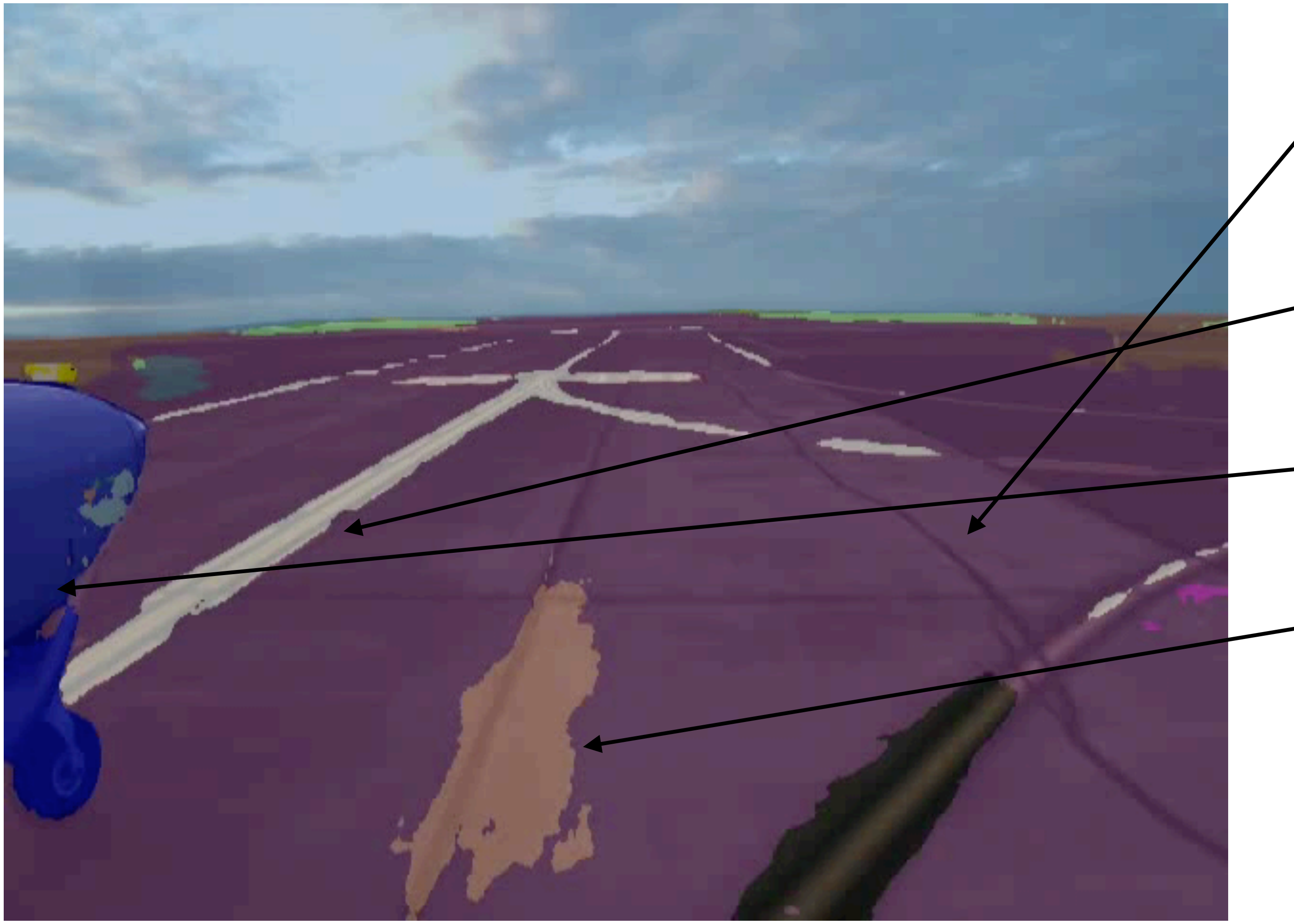




# Special features: rich appearance variation

# Special features: rich appearance variation

# Standard semantic segmenter



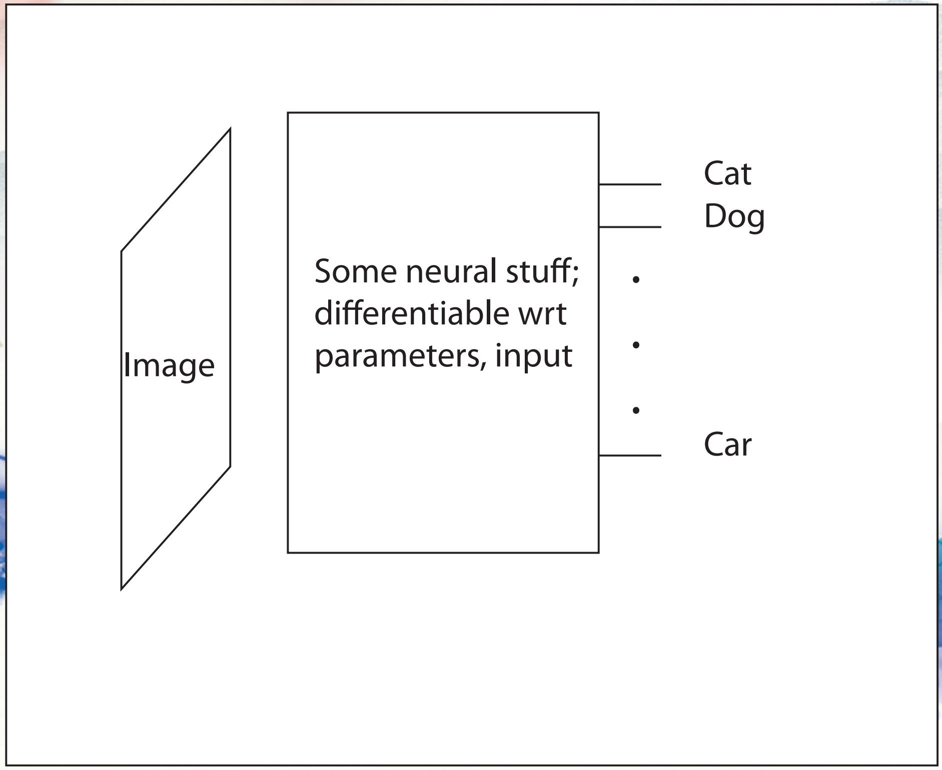
# XXX data consequences

# XXXX data consequences

# The questions that will plague us

- What representation do we need?
- How much data do we need to make it?
  - and where do we get it?
- How do we know if it works?

# Image classification



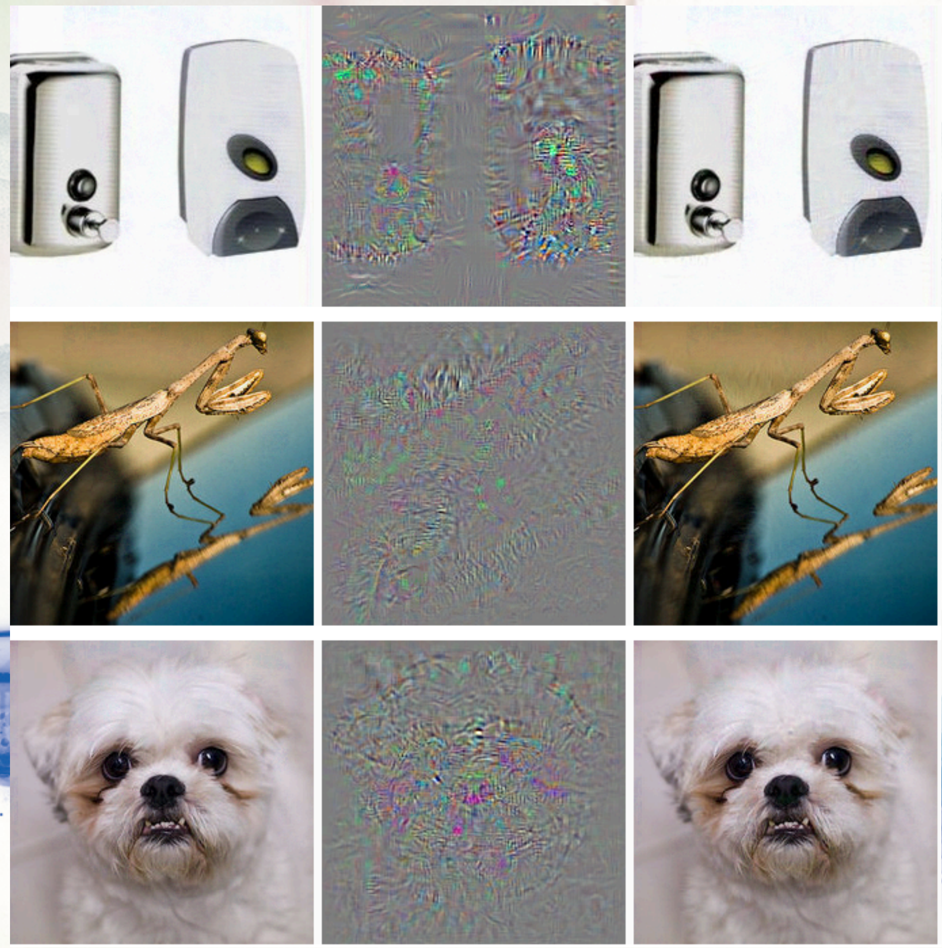
# Adversarial example

APOLLOSCAPE

- Search for
  - small update to image
  - such that
    - output for true class is low
    - output for some other class is high

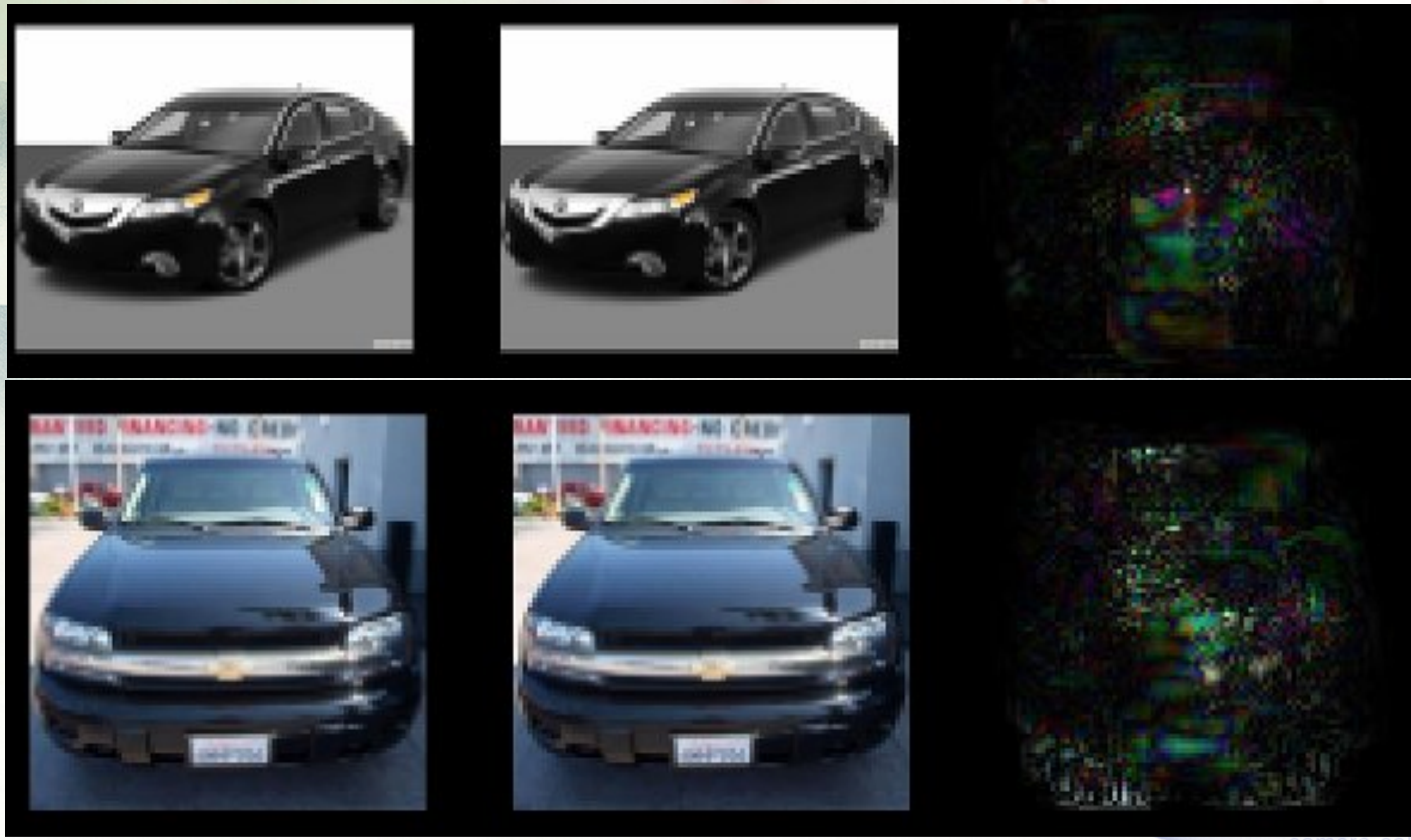
- Surprising fact:
  - such updates can be VERY small

Correctly classified



“Ostrich”

Szegedy et al, 13



Szegedy et al, 13

Lu et al 18

# Yolo attack

- Yolo uses a large image area to
  - predict boxes
  - predict classes
- This means that a detection is
  - affected by pixels OUTSIDE box

STOP
3-WAY
NO PARKING ANY TIME
person0.74167997
Autonomous navigation and exploration
advanced driver assistance
systems
monitoring
driver-vehicle
interfaces
STOP
3-WAY
NO PARKING ANY TIME


motorbike0.65933144
stop_sign0.83716106
STOP
HOOTERS
Casino Hotel
TAXI STAND


Computer vision technology
advanced driver assistance
systems
Visual driver
monitoring
navigation and exploration


stop_sign0.8481956
STOP
HOOTERS
Casino Hotel
TAXI STAND

# The questions that will plague us

- What representation do we need?
- How much data do we need to make it?
  - and where do we get it?
- How do we know if it works?