Local navigation

Autonomous robots, TME290

Ola Benderius

ola.benderius@chalmers.se

Applied artificial intelligence
Vehicle engineering and autonomous systems
Mechanics and maritime sciences
Chalmers

Introduction

Local navigation

Assuming that we can sense and perceive the local frame. How can we apply control to achieve our goals?

A coordinate system suitable for local navigation (1 of 4)

In our vision algorithm, the camera simply represents an image. For images, coordinates are given as (i, j) originating from the **top left** corner.

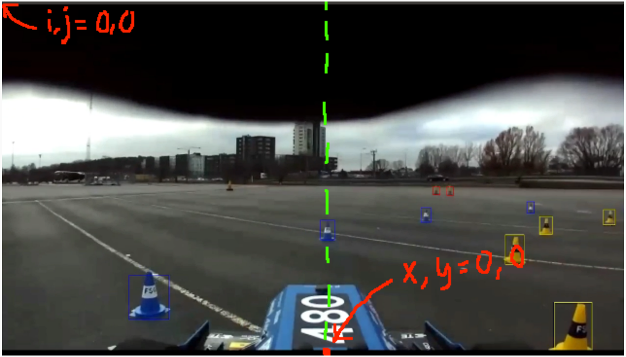
In our vision algorithm, the camera simply represents an image. For images, coordinates are given as (i, j) originating from the **top left** corner.

A coordinate system suitable for local navigation (2 of 4)

To fit better with navigation we would like to have the origin fixed to the centre axle of the robot.

A coordinate system suitable for local navigation (2 of 4)

To fit better with navigation we would like to have the origin fixed to the centre axle of the robot.

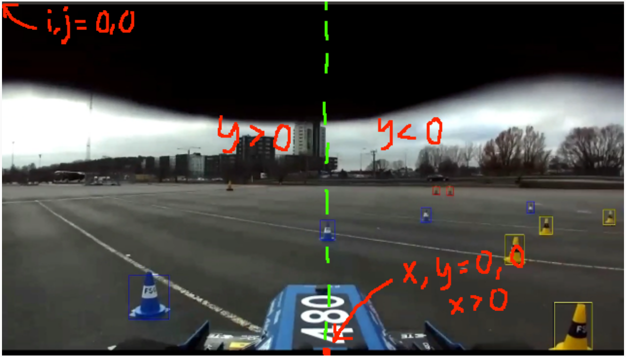


A coordinate system suitable for local navigation (3 of 4)

To comply with the ISO standard for vehicle-fixed coordinate systems: longitudinal (x) is positive forwards, lateral (y) is positive leftwards.

A coordinate system suitable for local navigation (3 of 4)

To comply with the ISO standard for vehicle-fixed coordinate systems: longitudinal (x) is positive forwards, lateral (y) is positive leftwards.

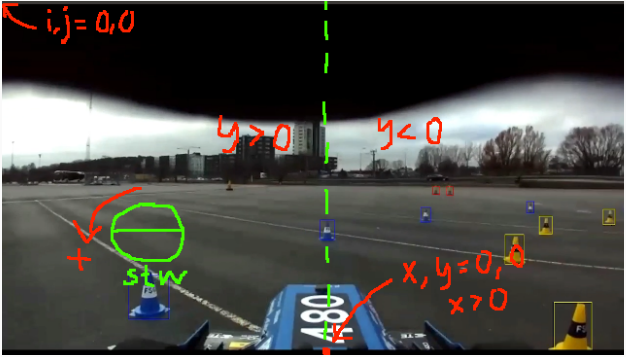


A coordinate system suitable for local navigation (4 of 4)

Steering is then defined as positive when steering counter clockwise.

A coordinate system suitable for local navigation (4 of 4)

Steering is then defined as positive when steering counter clockwise.



Applying control

We need to calculate the best cause of action in each time step, for example after each new video frame (e.g. at 50 Hz).

Applying control

We need to calculate the best cause of action in each time step, for example after each new video frame (e.g. at 50 Hz).

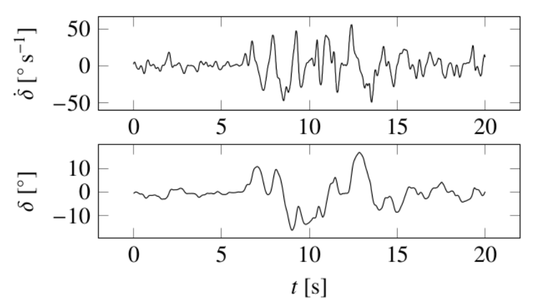
- What we need is a driver model! (control algorithm)

What is a driver model?

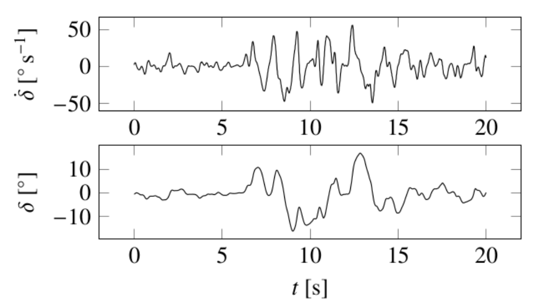
A driver model determines (mathematically) how to operate a machine (e.g. a vehicle) based on perceived environmental inputs (stimuli), in order to reach a given goal.

What is a driver model?

A driver model determines (mathematically) how to operate a machine (e.g. a vehicle) based on perceived environmental inputs (stimuli), in order to reach a given goal.



What is a driver model?



- Can be any mode of operation, e.g. steering wheel, pedals, lever
- Typically a driver model is mode exclusive, but sometimes combine modes

Comparing two approaches

Control theory

- The classic models
- Path following (target)
- Optimal control
- High repeatability

Comparing two approaches

Control theory

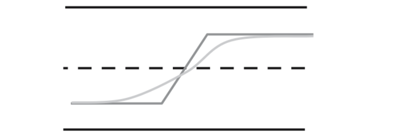
- The classic models
- Path following (target)
- Optimal control
- High repeatability

Bio-inspired methods

- Newer models
- Perception and cognition
- Satisficing behavior
- High driver realism

Comparing two approaches

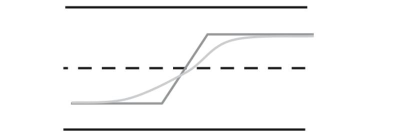
Path following



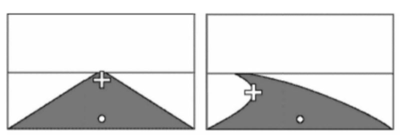
Optimal control

Comparing two approaches

Path following



Perception and cognition



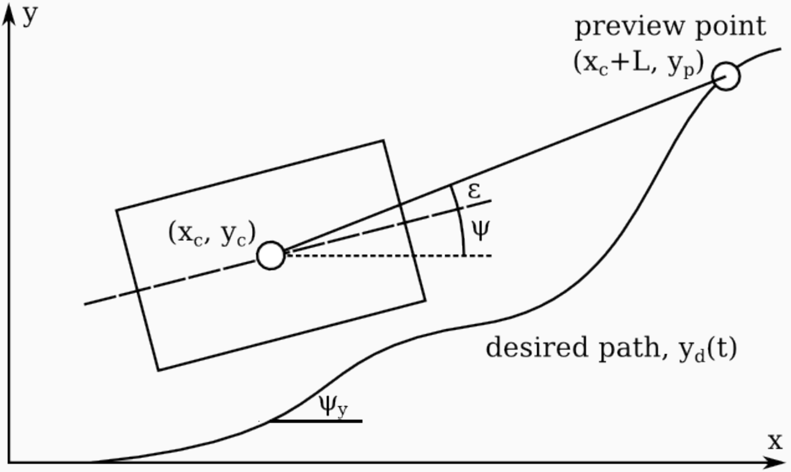
Optimal control

Satisficing behavior

Lateral control

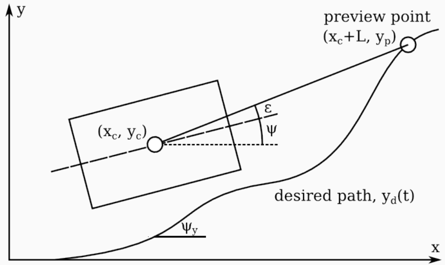
Lateral control is local steering relative to the environment, or control on the Y axis. Since Y is positive to the left, any steering angle relative to the ground is positive in the counterclockwise direction.

Simple path-following model



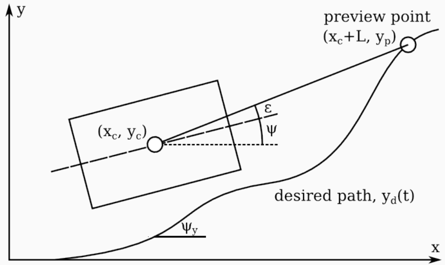
- The driver should follow the desired path
- A preview-point is used

Simple path-following model



$$y_p(t) = y_d\left(t + \frac{L}{v\cos(\psi)}\right)$$

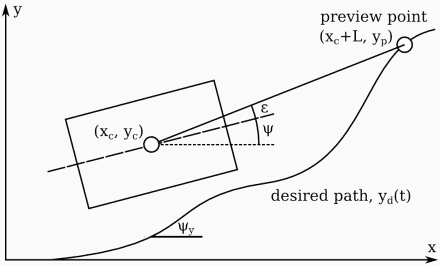
Simple path-following model



$$y_p(t) = y_d\left(t + \frac{L}{v\cos(\psi)}\right)$$

$$y_e(t) = y_p(t) - y_c(t), \psi_e(t) = \psi_y(t) - \psi(t)$$

Simple path-following model



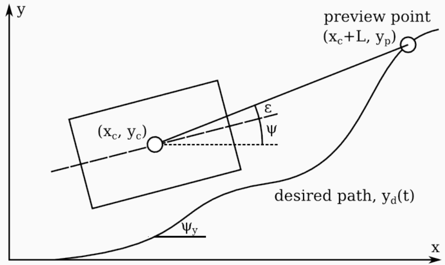
$$y_p(t) = y_d\left(t + \frac{L}{v\cos(\psi)}\right)$$

$$y_e(t) = y_p(t) - y_c(t), \psi_e(t) = \psi_y(t) - \psi(t)$$

Simplest preview point model

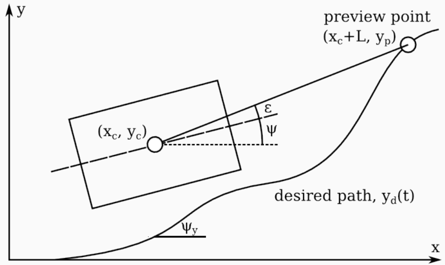
$$\delta(t) = k_1\psi_e(t - t_r) + k_2 y_e(t - t_r)$$

Simple path-following model



We can generalise the model a bit by defining it as an *aim point*:

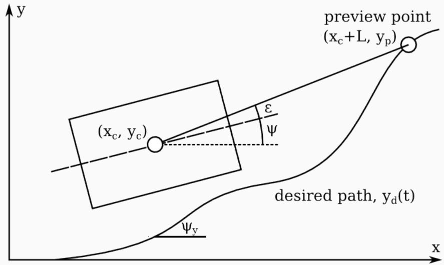
Simple path-following model



We can generalise the model a bit by defining it as an *aim point*:

$$\epsilon(t) = \text{arctan}\left(\frac{y_e(t)}{L}\right) - \psi(t)$$

Simple path-following model



We can generalise the model a bit by defining it as an *aim point*:

$$\epsilon(t) = \arctan\left(\frac{y_e(t)}{L}\right) - \psi(t)$$

Simplest aim point model

$$\delta(t) = k\epsilon(t - t_r)$$

Simple path-following model

To linearise, one can assume ψ and ϵ to be small:

Linearised aim point model

$$\delta(t) = k \left(\frac{y_d\left(t + \frac{L}{v} - t_r\right) - y_c(t - t_r)}{L} - \psi(t - t_r) \right)$$

Preview points and aim points

- Both defined by Kondo in the 50s

Preview points and aim points

- Both defined by Kondo in the 50s
- One of the most famous models use 8 preview points

Preview points and aim points

- Both defined by Kondo in the 50s
- One of the most famous models use 8 preview points
- One can also use a *preview horizon* (or *interval*)

Internal vehicle model

- *Some* driver models use an internal vehicle model
- Inverted (linear):
    - Input: lateral acceleration
    - Output: steering angle
- Typical in optimal control

Internal vehicle model

- *Some* driver models use an internal vehicle model
- Inverted (linear):
  - Input: lateral acceleration
  - Output: steering angle
- Typical in optimal control

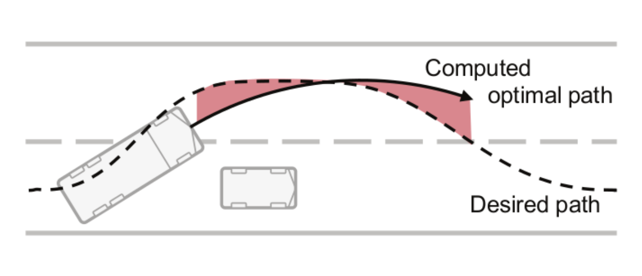
However, I would say that ordinary (realistic) driver behavior does *not* depend very much on vehicle dynamics. A good understanding of human perception is more important.

Example from the driving simulator

Example from the driving simulator

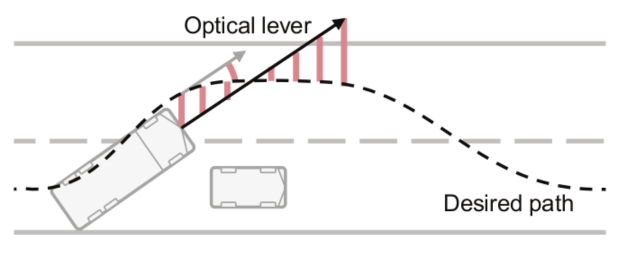
Now I will present four well-know driver models. Your task is to decide which one is the best to predict the demonstrated steering behaviour based on visual input.

Steer to minimise path deviation: MacAdam, 1981



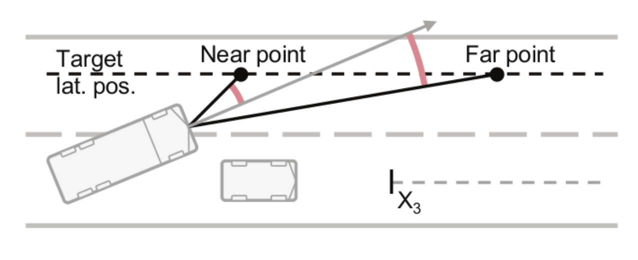
$$J(t) = \int_{t-T_\mathrm{R}}^{t-T_\mathrm{R}+T_\mathrm{P}} \left(f(\eta) - y(\eta)\right)^2 \, d\eta$$

Steer when preview points are deviating: Sharp et al., 2000



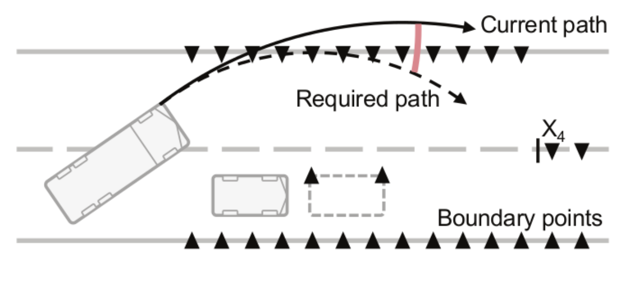
$$\delta = K_\psi e_\psi + K_1 e_1 + K_\mathrm{p} \sum_{i=2}^{n} K_i e_i$$

Steer to keep points stationary: Salvucci and Gray, 2004



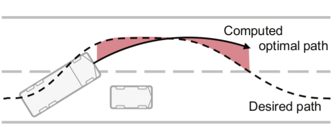
$$\dot{\delta} = k_{\mathrm{nP}}\dot{\theta}_{\mathrm{n}} + k_{\mathrm{f}}\dot{\theta}_{\mathrm{f}} + k_{\mathrm{nI}}\theta_{\mathrm{n}}$$

Steer by collision avoidance: Gordon and Magnuski, 2006

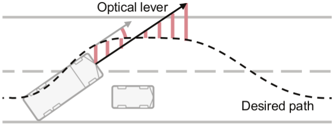


$$\dot{\delta} = -\frac{L}{G\tau_{\mathrm{s}}v_x}(\dot{\psi} - \dot{\psi}_{\mathrm{req}})$$

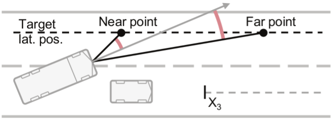
Any idea which one is the best?



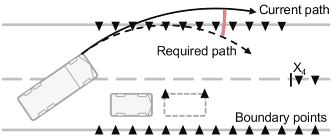
(a) MacAdam



(b) Sharp *et al.*

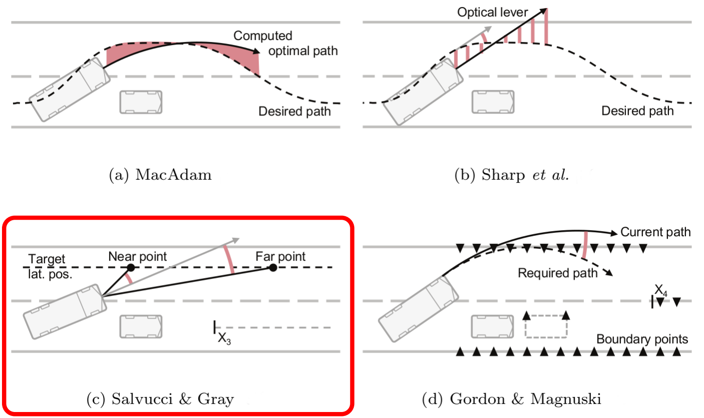


(c) Salvucci & Gray



(d) Gordon & Magnuski

Any idea which one is the best?



(a) MacAdam

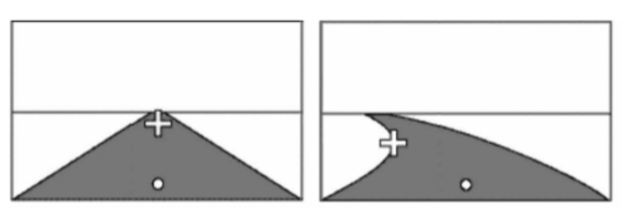
(b) Sharp *et al.*

(c) Salvucci & Gray

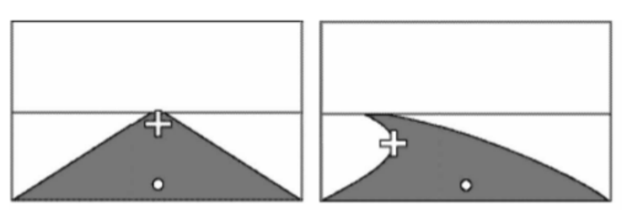
(d) Gordon & Magnuski

Salvucci and Gray: More details

Two different scenarios:



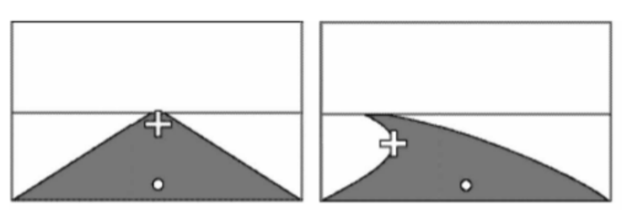
Salvucci and Gray: More details



The far point (cross)

- Distant fixation
- Position is situation dependent (see figure)
- Monitor lateral stability
- Predict steering for upcoming curvature

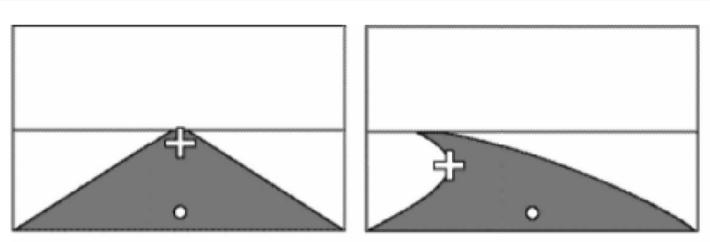
Salvucci and Gray: More details



The near point (circle)

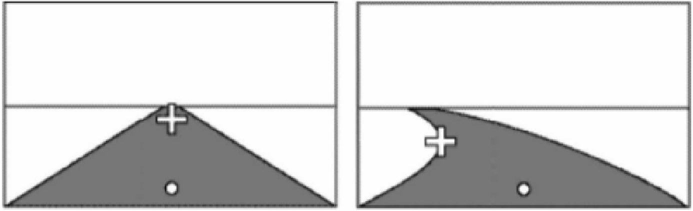
- Nearby distance, visible through windshield
- Center of road
- Monitor lateral position and stability
- Peripherally, no visual fixation

What does this mean?



$$\dot{\delta} = k_{\mathrm{nP}}\dot{\theta}_{\mathrm{n}} + k_{\mathrm{f}}\dot{\theta}_{\mathrm{f}} + k_{\mathrm{nI}}\theta_{\mathrm{n}}$$

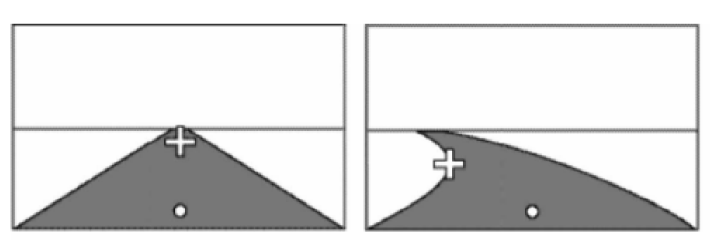
What does this mean?



$$\dot{\delta} = k_{\mathrm{nP}}\dot{\theta}_{\mathrm{n}} + k_{\mathrm{f}}\dot{\theta}_{\mathrm{f}} + k_{\mathrm{nI}}\theta_{\mathrm{n}}$$

we move the steering wheel **when**

What does this mean?

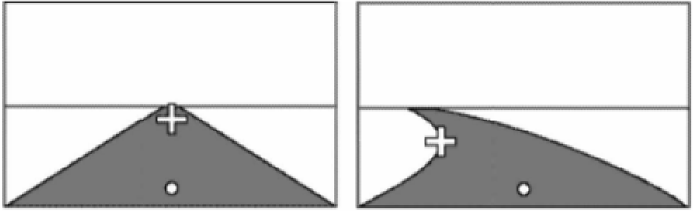


$$\dot{\delta} = k_{\mathrm{nP}}\dot{\theta}_{\mathrm{n}} + k_{\mathrm{f}}\dot{\theta}_{\mathrm{f}} + k_{\mathrm{nI}}\theta_{\mathrm{n}}$$

we move the steering wheel **when**

the near point is moving **or**

What does this mean?



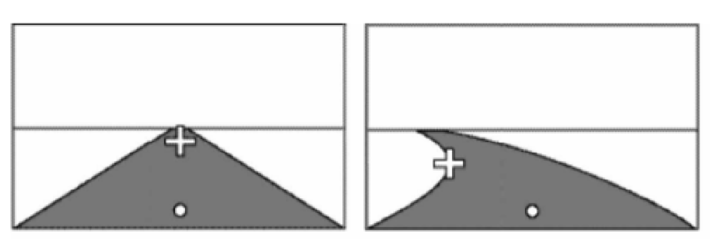
$$\dot{\delta} = k_{\mathrm{nP}}\dot{\theta}_{\mathrm{n}} + k_{\mathrm{f}}\dot{\theta}_{\mathrm{f}} + k_{\mathrm{nI}}\theta_{\mathrm{n}}$$

we move the steering wheel **when**

 the near point is moving **or**

 the far point is moving **or**

What does this mean?



$$\dot{\delta} = k_{\mathrm{nP}}\dot{\theta}_{\mathrm{n}} + k_{\mathrm{f}}\dot{\theta}_{\mathrm{f}} + k_{\mathrm{nI}}\theta_{\mathrm{n}}$$

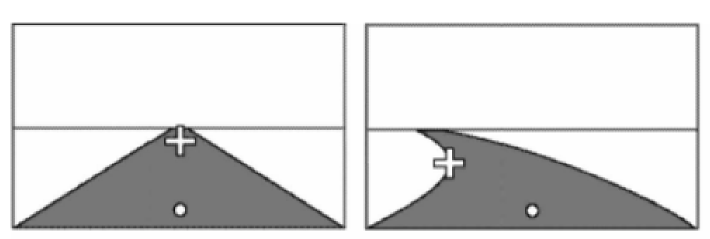
we move the steering wheel **when**

the near point is moving **or**

the far point is moving **or**

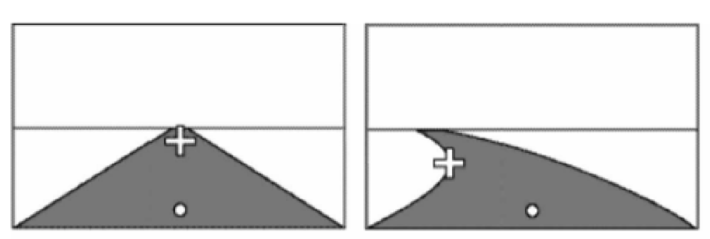
the near point is *not* directly in-front

What does this mean?



$$\dot{\delta} = k_{\mathrm{nP}}\dot{\theta}_{\mathrm{n}} + k_{\mathrm{f}}\dot{\theta}_{\mathrm{f}} + k_{\mathrm{nI}}\theta_{\mathrm{n}}$$

What does this mean?



$$\dot{\delta} = k_{\mathrm{nP}}\dot{\theta}_{\mathrm{n}} + k_{\mathrm{f}}\dot{\theta}_{\mathrm{f}} + k_{\mathrm{nI}}\theta_{\mathrm{n}}$$

Interpretation

Salvucci and Gray teaches that steering is not about moving the vehicle, but rather moving the perceived world!

Path-following vs. perception

- Using a path
    - Repeatable behaviour
    - Assessable (how close)
    - Defined before-hand (static)

Path-following vs. perception

- Using a path
    - Repeatable behaviour
    - Assessable (how close)
    - Defined before-hand (static)
- Human perception
    - One fixation point
    - No *extra* information
    - Assessing angles instead of distances

Longitudinal control

Longitudinal control is local speed adaptation relative to the environment, or control on the X axis (positive X is forwards).

Static behavior

In many cases, a static behavior is enough

- Constant deceleration or acceleration

Static behavior

In many cases, a static behavior is enough

- Constant deceleration or acceleration
- Speed profile, simple regulator
  - If faster: Add throttle
  - If slower: Release throttle
  - If much slower: Add brake

Dynamic behavior

For dynamic road environments

- Traffic simulation (car following)
- Sudden events (e.g. evasion)

Car-following model: GHR

Gazis-Herman-Rothery (GHR)

$$\ddot{x}(t) = \lambda \Delta \dot{x}(t - t_r)$$

$$\lambda = a \frac{\dot{x}^m(t)}{\Delta x^l(t - t_r)}$$

- Δx is the distance between the cars
- λ is called the sensitivity term
- a, m, l are model parameters
- Cars will behave like tied with rubber bands

Car-following model: GHR

Gazis-Herman-Rothery (GHR)

$$\ddot{x}(t) = \lambda \Delta \dot{x}(t - t_r)$$

$$\lambda = a \frac{\dot{x}^m(t)}{\Delta x^l(t - t_r)}$$

- Δx is the distance between the cars
- λ is called the sensitivity term
- a, m, l are model parameters
- Cars will behave like tied with rubber bands

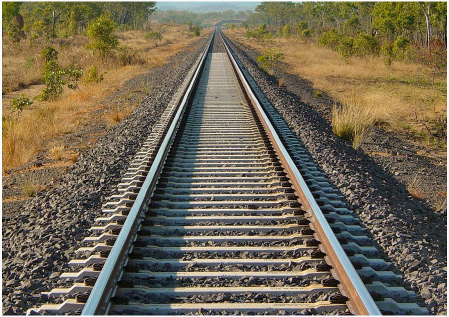
Note

This is just an example, the GHR model is not very useful in the general sense.

Is GHR a cognitive model?

- Well, its more of a behavioral model
- But the sensitivity term can be tuned to underlying cognitive aspects
- Here, the underlying cognitive principle is called *looming*

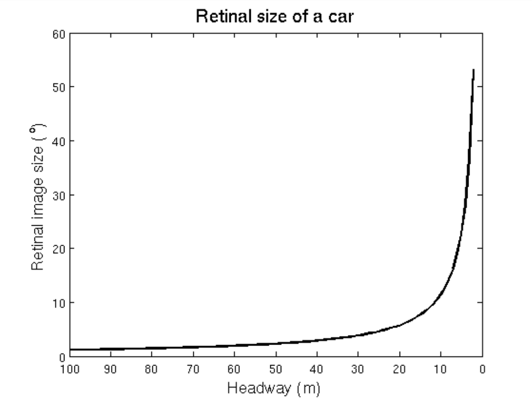
Perspective



$$\theta = 2\arctan\frac{w}{2d} \tag{1}$$

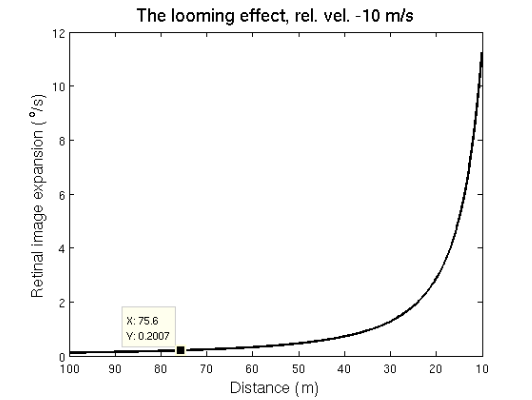
Perspective of a car

Perspective of a car



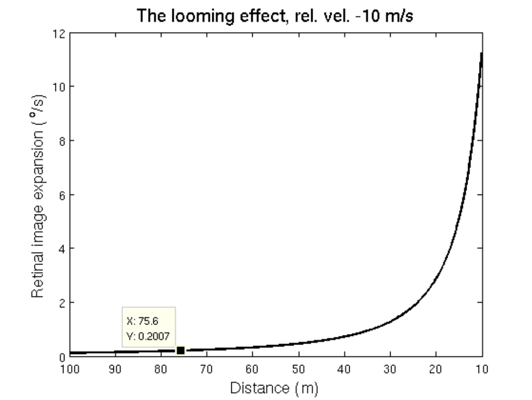
- The width of a car is about 2 m
- The key factor is the angle at the retina
- A human has a field-of-view of 180 degrees
- Looming is when the angle expands

The looming effect



- Expansion rate $> 0.2^\circ/\mathrm{s}$ – *attention* is triggered
- Large expansion rates – *fear* is triggered

The looming effect



- Expansion rate $> 0.2^\circ/\mathrm{s}$ – *attention* is triggered
- Large expansion rates – *fear* is triggered
- The expansion rate can be used as a modifier in a brake model

Cognitive braking, illustration

$$p_b(t) = \begin{cases} \frac{l(t-t_r)-l_0}{l_1-l_0}, & \text{if } l(t-t_r) > l_0. \\ 1, & \text{if } l(t-t_r) > l_1. \\ 0, & \text{otherwise.} \end{cases}$$

Where $l(t)$ is the looming rate, l_0 is 0.2 $^\circ/\text{s}$, and l_1 the maximum comfortable looming rate of the front vehicle.

Combining longitudinal and lateral control

As already mentioned, models are typically treating one mode at a time. Either one model can be used per modality, but there are a few models that needs to integrate several modes.

Questions

Please post all questions on the Canvas discussion pages, in that way we can all benefit from the answers, and I can highlight important outcomes.