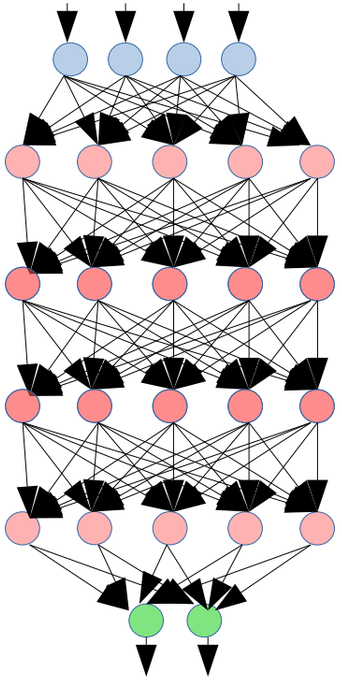
Any Hope for Perceptual DNNs?

$$(r_1, g_1, b_1, \ldots r_N, g_N, b_N)$$

Image (road scene)



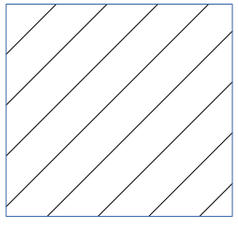
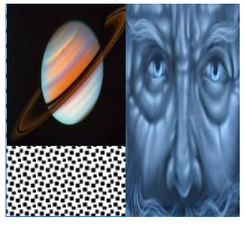
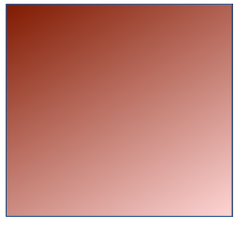
“Too congested to accelerate”

Input i

| Input Spac

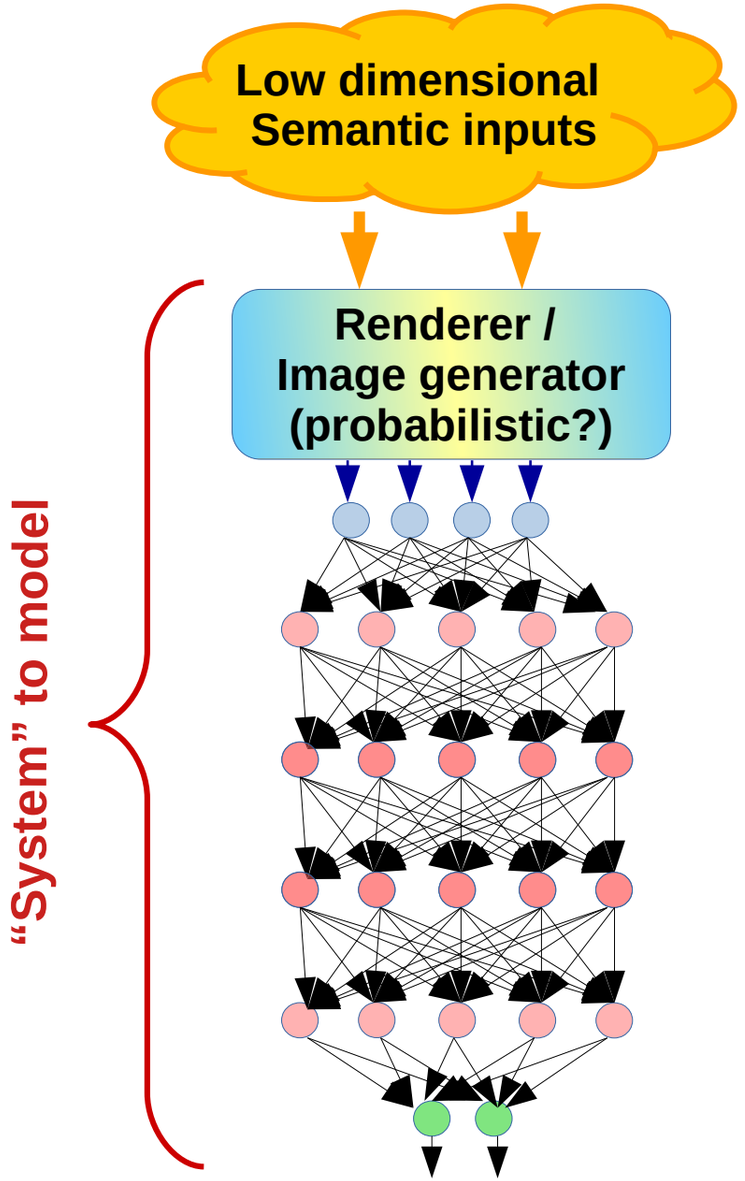
High dimensional, large input space

Most images inconsequential, have no semantic similarity to what can possibly arise on a road

Can we restrict specs to a lower dimensional, smaller, meaningful input space?

Any Hope for Perceptual DNNs?



Time of Day: {Morning, Noon, Afternoon, Dusk, Night}
Weather: {Clear, Cloudy, Snowing, Raining}
Lanes: {Wide, Medium, Narrow, None}
Road direction: {Straight, Bending}
Other vehicles within 10m: {0, 1-3, 4-8, 9-15, > 15}
Behaviour of other vehicles: {Lane disciplined, Chaotic}

Dimensions of semantic inp space = 6
|Semantic inp space| = 5x4x4x2x5x2 = 1600

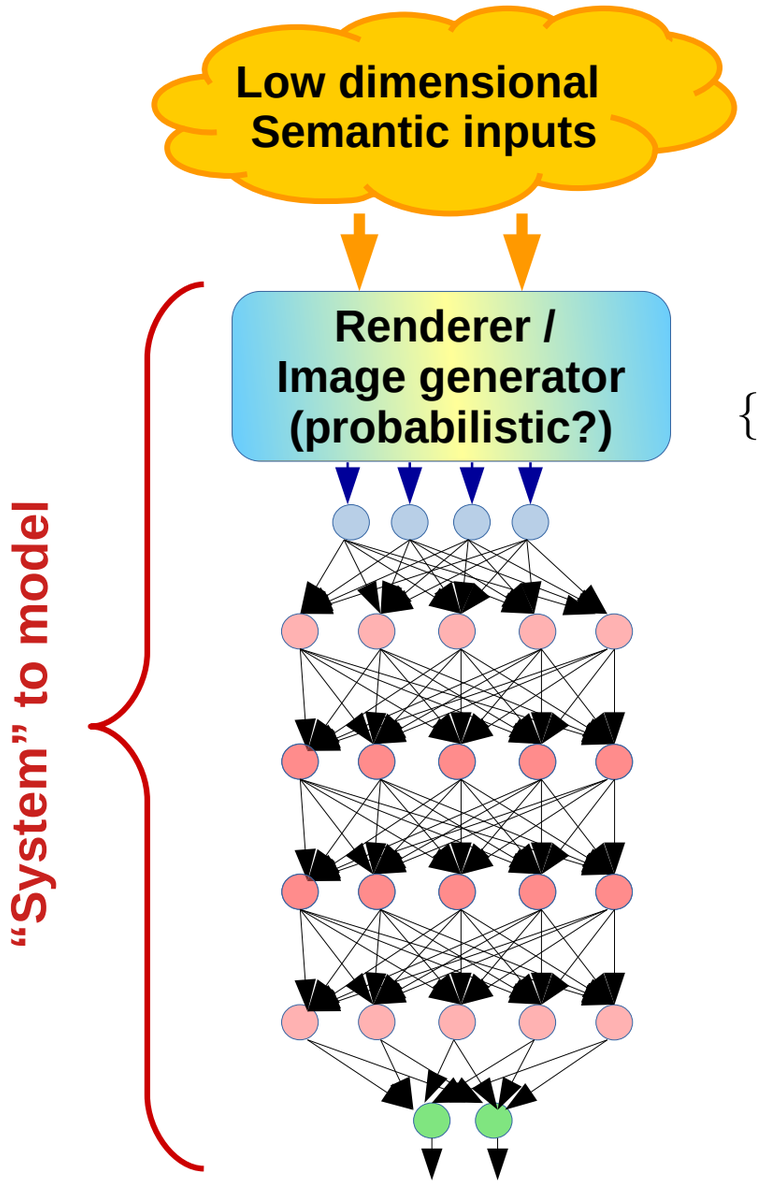
Dimensions of image inp space = 100x100x3 = 30000
|Image inp space| = $256^{100x100x3}$

{ Pre-condition on semantic inputs **s** }

$$i \leftarrow \rho(\mathbf{s});\ //\rho\text{: Model of renderer}$$
$$y \leftarrow \nu(i);\ //\nu\text{: Model of perceptual DNN}$$

{ Post-condition on y}

Any Hope for Perceptual DNNs?



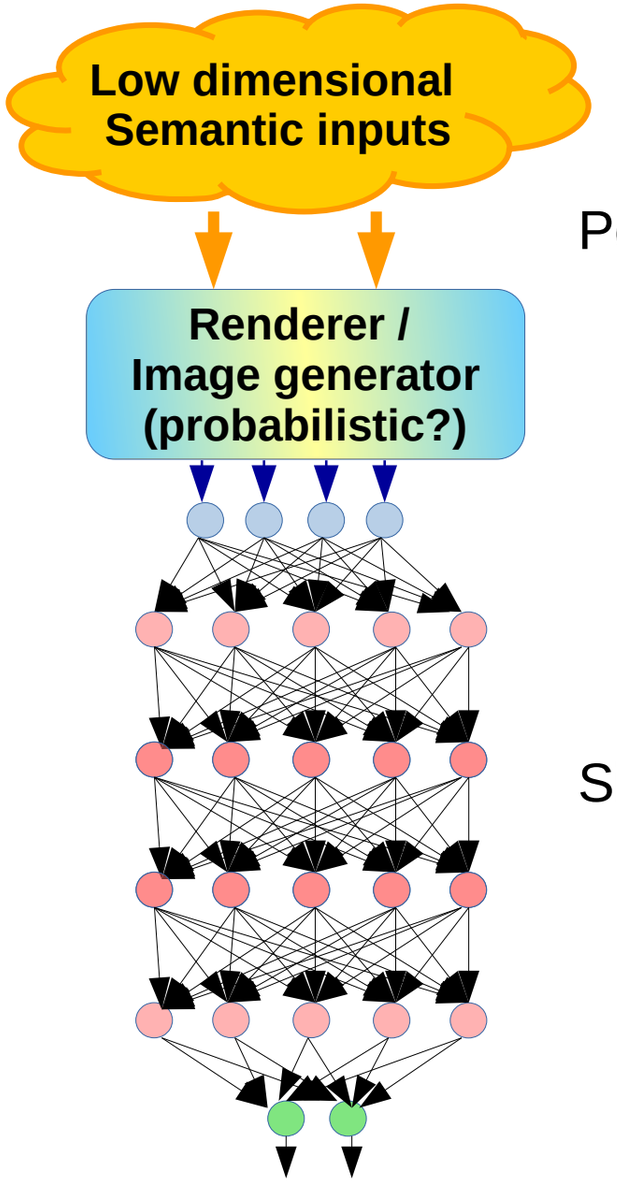
T: {Morning, Noon, Afternoon, Dusk, Night}
W: {Clear, Cloudy, Snowing, Raining}
L: {Wide, Medium, Narrow, None}
Rd: {Straight, Bending}
O: {0, 1-3, 4-8, 9-15, > 15}
B: {Lane disciplined, Chaotic}

$$\{(O > 15) \bigvee (L = \mathrm{W}) \wedge ((O \geq 9) \wedge (B = \mathrm{Ch})) \bigvee (L = \mathrm{M}) \wedge ((O \geq 9) \vee ((O \geq 4) \wedge (B = \mathrm{Ch}))) \bigvee ((L = \mathrm{N}) \vee (L = \mathrm{None})) \wedge ((O \geq 4) \vee ((O \geq 1) \wedge (B = \mathrm{Ch})))\}$$

$$i \leftarrow \rho(T, W, L, Rd, O, B);\ //\rho\text{: Model of renderer}$$
$$y \leftarrow \nu(i);\ //\nu\text{: Model of perceptual DNN}$$

{ y = "Too congested to accelerate" }

Any Hope for Perceptual DNNs?



Potential **“problems”**:

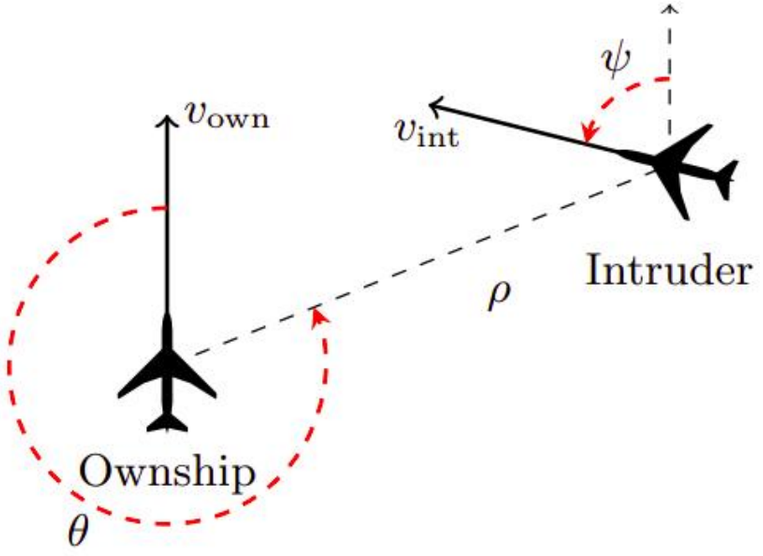
- Doesn’t cover entire input space
  - Enrich semantic space to cover most/all meaningful inputs
  - Use richer rendering modules
- Need to model renderer
  - Use abstract / non-deterministic / probabilistic models

Significant **“benefits”**:

- Can eliminate large parts of irrelevant/meaningless input space
- Provide guarantees over large parts of meaningful input space

One Spec vs Multiple Sub-specs

ACAS-Xu



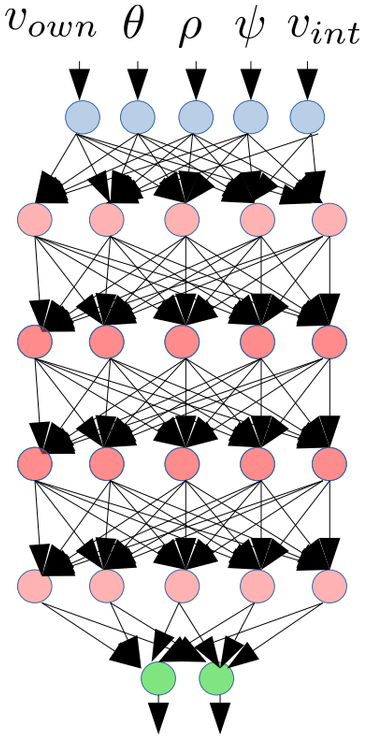
$$\{(\rho \geq 55947.691ft) \wedge (v_{own} \geq 1145ft/s) \wedge (v_{int} \leq 60ft/s)\}$$

$$\textbf{Score} \leftarrow \nu(\rho, v_{own}, v_{int}, \theta, \psi)$$

$$\{\textbf{Score}[\text{COC}] \leq 1500\}$$

Spec 1

Flight parameters

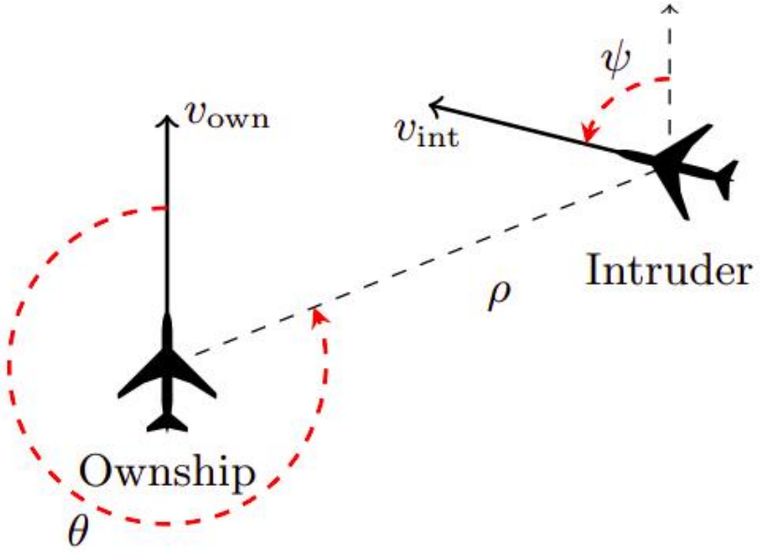


Score
(Horizontal Advisory)

Source: Reluplex: An Efficient SMT Solver for Verifying Deep Neural Networks, by Katz et al, 2017

One Spec vs Multiple Sub-specs

ACAS-Xu



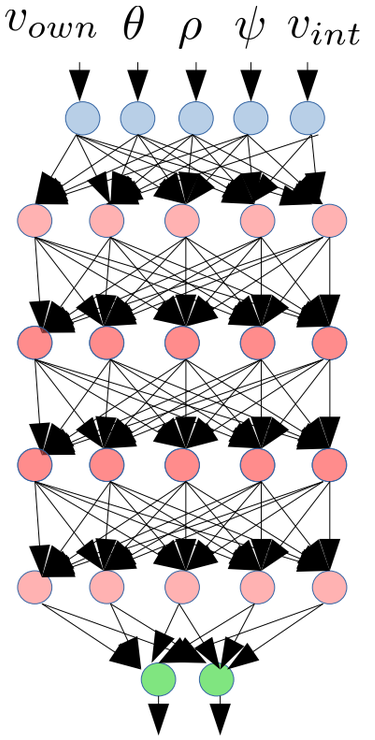
$$\{(0 \leq \rho \leq 60760ft) \wedge (1000 \leq v_{own} \leq 1200ft/s) \wedge$$
$$(0 \leq v_{int} \leq 1200ft/s) \wedge (-3.141592 \leq \theta, \psi \leq 3.141592)\}$$

$$\textbf{Score} \leftarrow \nu(\rho, v_{own}, v_{int}, \theta, \psi)$$

Spec 7

$$\{\text{argmin}_x \textbf{Score}[\text{x}] \notin \{\text{StrongRight}, \text{StrongLeft}\}\}$$

Flight parameters



Score
(Horizontal Advisory)

Source: Reluplex: An Efficient SMT Solver for Verifying Deep Neural Networks, by Katz et al, 2017

One Spec vs Multiple Sub-specs

ACAS-Xu

$$\{(v_{own} \geq 1000ft/s) \ \wedge \ (0 \leq v_{int} \leq 1200ft/s)\}$$

$$\mathbf{Score} \leftarrow \nu(\rho, v_{own}, v_{int}, \theta, \psi)$$

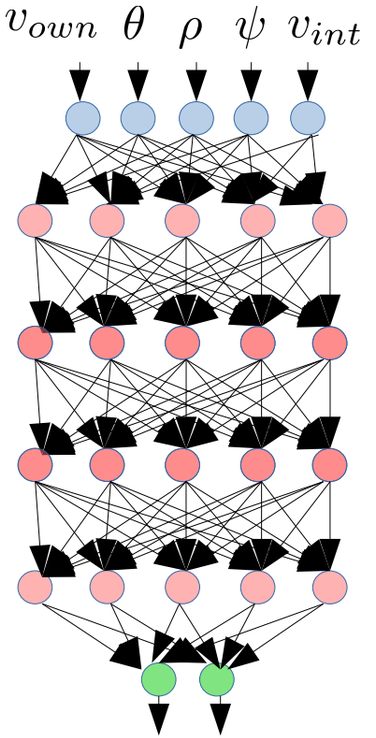
$$(\rho \geq 55947.691ft) \wedge (v_{own} \geq 1145ft/s) \wedge (v_{int} \leq 60ft/s)$$
$$\Rightarrow \quad \mathbf{Score}[\text{COC}] \leq 1500$$

$$\wedge$$

$$(0 \leq \rho \leq 60760ft) \wedge (1000 \leq v_{own} \leq 1200ft/s) \wedge$$
$$(0 \leq v_{int} \leq 1200ft/s) \wedge (-3.141592 \leq \theta, \psi \leq 3.141592)$$
$$\Rightarrow \quad \text{argmin}_x \mathbf{Score}[\text{x}] \notin \{\text{StrongRight}, \text{StrongLeft}\}$$

Specs 1+7

Flight parameters



Score
(Horizontal Advisory)

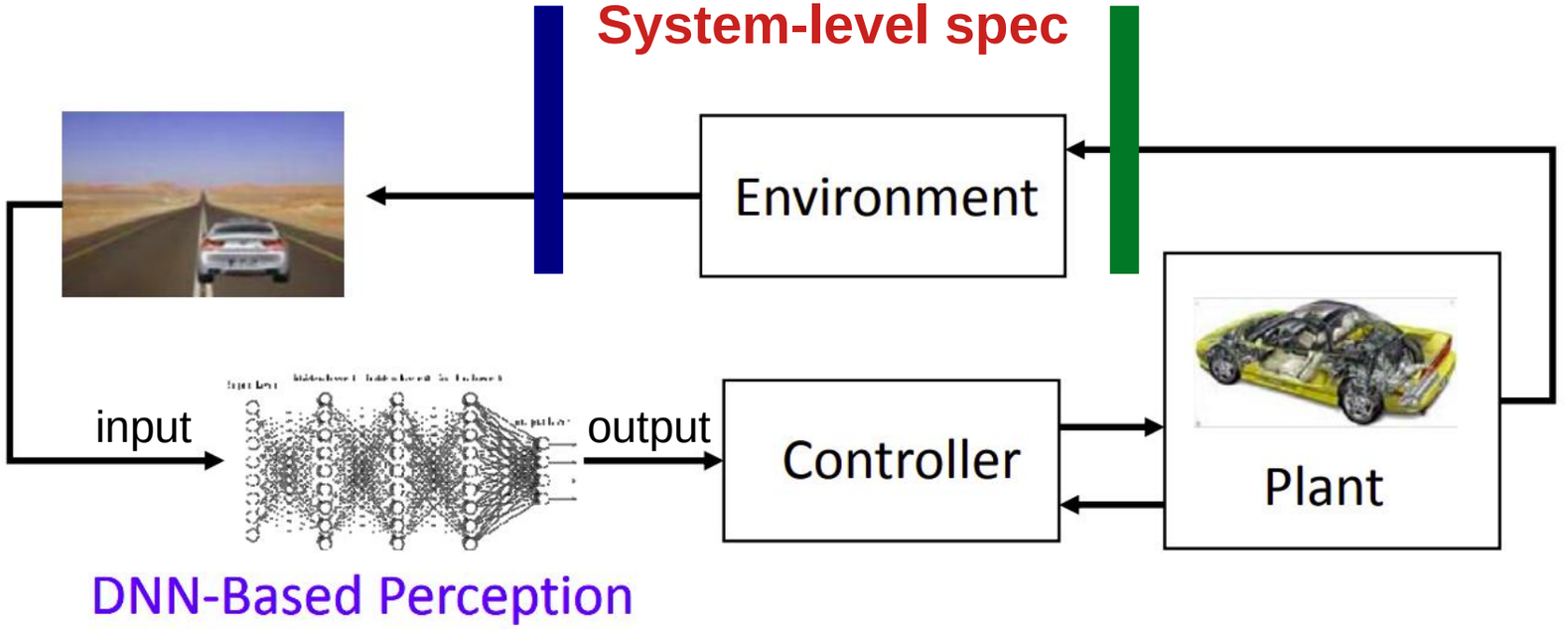
Source: Reluplex: An Efficient SMT Solver for Verifying Deep Neural Networks, by Katz et al, 2017

One Spec vs Multiple Sub-specs

Multiple sub-specs generally preferred over one all-encompassing spec

- Separation of concerns
- Easy understandability
- Proofs often easier
- Modularly build spec over time

Other Ways of Specifying Properties



Source: Seshia et al, Formal Verification of Deep Neural Networks, 2018

{ (own_velocity > 30 km/h) and (road_straight_ahead) and (vehicles_within_5m = 0) }

Model of DNN + Controller + Plant

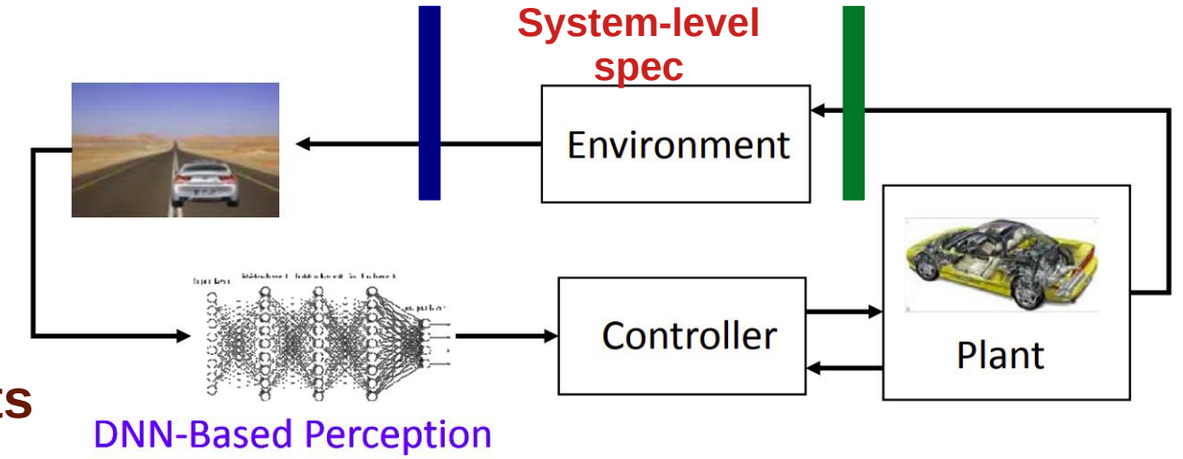
{ Steering = straight }

Other Ways of Specifying Properties

No need for perceptual specs
- **Often easier to specify**

Require models of other components
- **May be harder to verify**

Classification errors of DNN may not translate to system level spec violations



Source: Seshia et al, Formal Verification of Deep Neural Networks, 2018

{ (own_velocity > 30 km/h) and (road_straight_ahead) and (vehicles_within_5m = 0) }

Model of DNN + Controller + Plant

{ Steering = straight }

Specifying Properties of Neural Networks

Pause n Reflect

DNNs are intended to mimic human reasoning

Is ideal human reasoning amenable to formal specification?

There are “boundaries” of acceptable/unacceptable human behaviour

Can we specify these boundaries?

Rules, laws, code of conduct

Do they have unique interpretations?

Do they evolve?

Is there a counterpart for neural networks?