Recognition: Past, present, future?

Benozzo Gozzoli, Journey of the Magi, c. 1459

Straightforward image classification

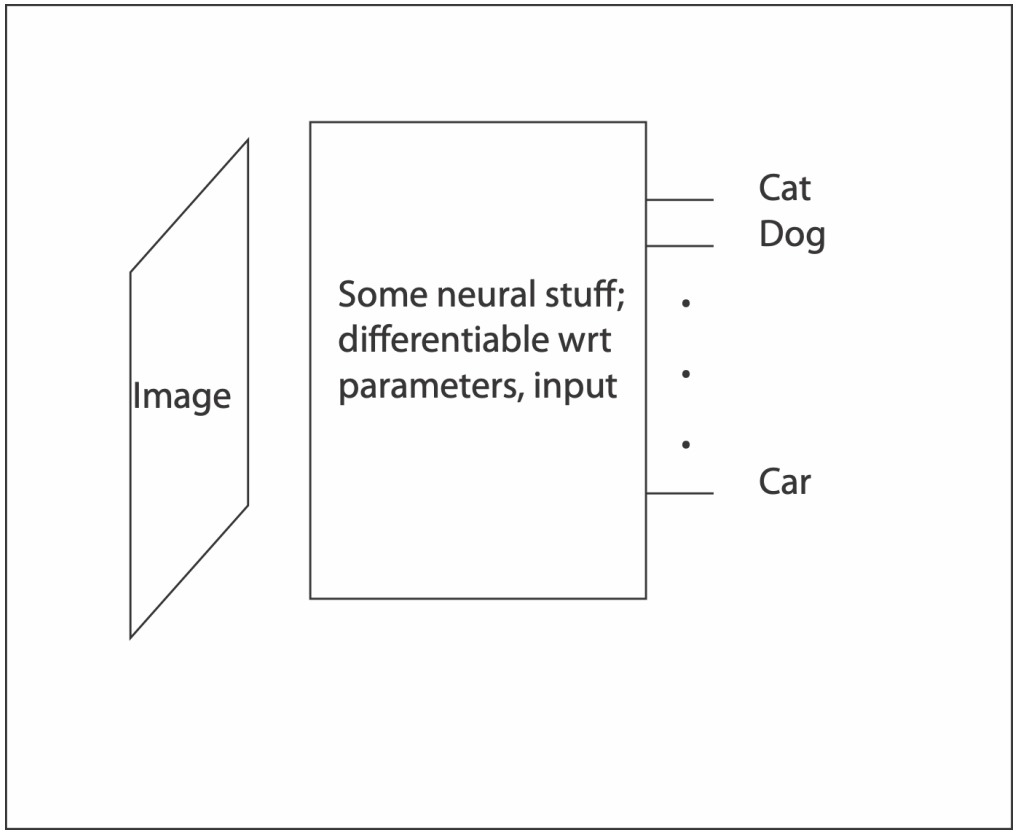
We know basics for two class classification

- Image encoder + Logistic regression
  - train encoder parameters, lr parameters with training data
  - evaluate with test data

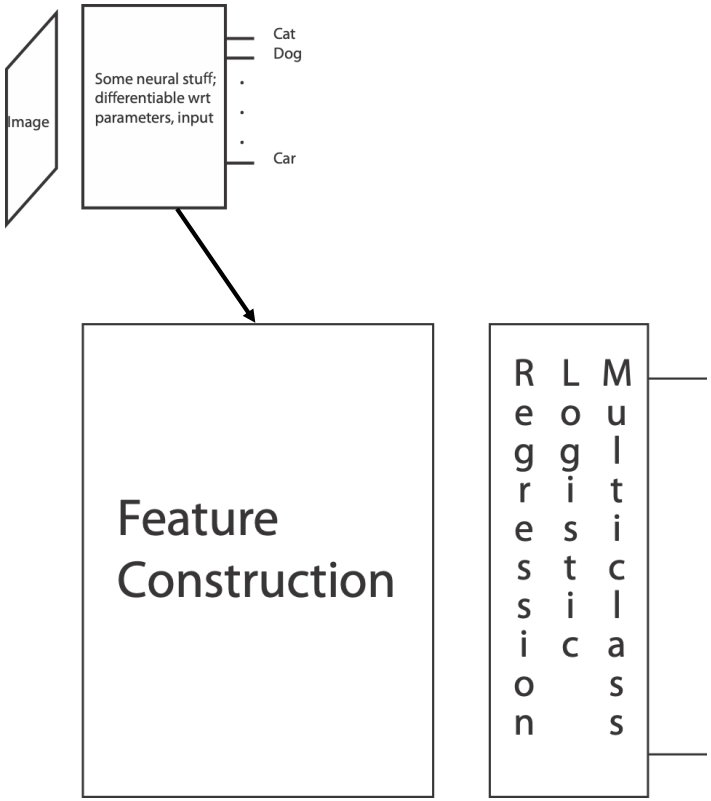
Open:

- more than two classes
- best encoder

Image classification



Under the hood



Multiclass logistic regression

For classes 1, .., C

Given a feature vector

$$\mathbf{x}$$

Form

$$\mathbf{w}_i^T \mathbf{x}$$

Interpret by

$$P(\text{example is of class i}) = \frac{e^{\mathbf{w}_i^T \mathbf{x}}}{\sum_k e^{\mathbf{w}_k^T \mathbf{x}}}$$

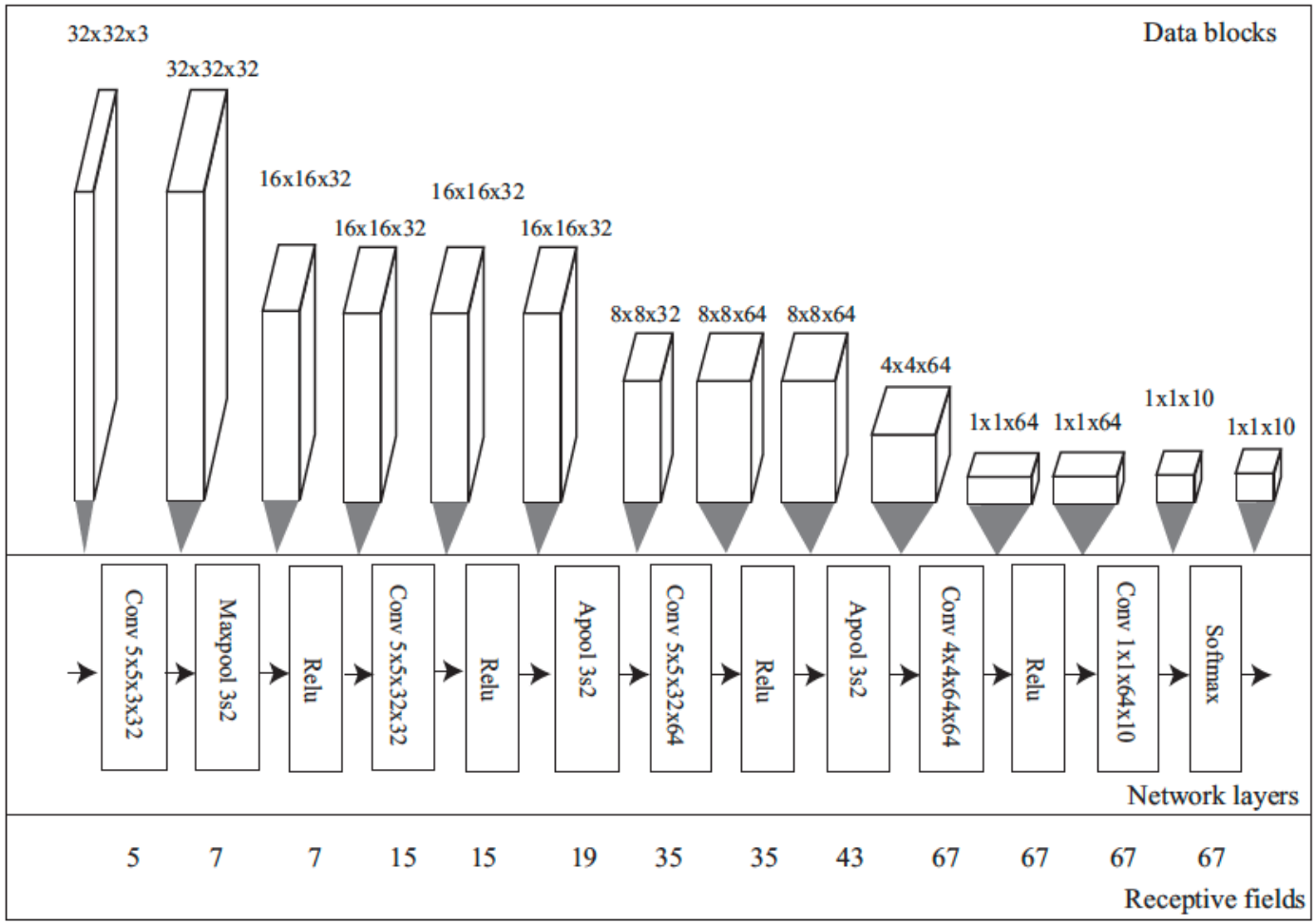


FIGURE 7.3: *Three different representations of the simple network used to classify CIFAR-10 images for this example. Details in the text.*

FIGURE 7.7: *Visualizing the patterns that the final stage ReLU's respond to for the simple CIFAR example. Each block of images shows the images that get the largest output for each of 10 ReLU's (the ReLU's were chosen at random from the 64 available in the top ReLU layer). Notice that these ReLU outputs don't correspond to class – these outputs go through a fully connected layer before classification – but each ReLU are clearly responds to a pattern, and different ReLU's respond more strongly to different patterns.*

Classification variants

Predict more labels with complex semantics

Predict a cost function from the image

- report the minimum

This allows

- Visual question answering
  – function accepts question, offered answers and takes min at best
- Writing sentences
  – choose sentence that minimizes cost

Situations

CLIPPING			
ROLE	VALUE	ROLE	VALUE
AGENT	MAN	AGENT	VET
SOURCE	SHEEP	SOURCE	DOG
TOOL	SHEARS	TOOL	CLIPPER
ITEM	WOOL	ITEM	CLAW
PLACE	FIELD	PLACE	ROOM

JUMPING			
ROLE	VALUE	ROLE	VALUE
AGENT	BOY	AGENT	BEAR
SOURCE	CLIFF	SOURCE	ICEBERG
OBSTACLE	-	OBSTACLE	WATER
DESTINATION	WATER	DESTINATION	ICEBERG
PLACE	LAKE	PLACE	OUTDOOR

SPRAYING			
ROLE	VALUE	ROLE	VALUE
AGENT	MAN	AGENT	FIREMAN
SOURCE	SPRAY CAN	SOURCE	HOSE
SUBSTANCE	PAINT	SUBSTANCE	WATER
DESTINATION	WALL	DESTINATION	FIRE
PLACE	ALLEYWAY	PLACE	OUTSIDE

Yatskar+Zettlemoyer+Farhadi 2016

Visual Question Answering

Figure 1.22 Visual question answering systems produce natural language answers to questions about images. It is difficult for a VQA system to hide ignorance in the way that a captioning system can. Here the system is producing quite sensible answers to rather difficult questions about the image (answers are typically chosen from a multiple choice set). Figure courtesy of Devi Parikh, produced by a system described in "Making the V in VQA Matter: Elevating the Role of Image Understanding in Visual Question Answering" by Goyal, Khot, Summers-Stay, Batra, and Parikh and published in CVPR 2017.

doesn’t always work…

Q. How many holes are in the pizza?
A. 8

Q. What color is the front right leg?
A. Brown

Q. What letter is on the racket?
A. w

Q. Why is the sign bent?
A. It’s not

Figure 1.23 Because it is difficult for a VQA system to hide ignorance in the way that a captioning system can, the mistakes can be informative and highlight how difficult it is to produce accurate visual representations. For example, the system is guessing about the number of holes in a pizza, because it doesn’t understand the conventions about what holes are worth talking about, and it has real difficulty counting. Similarly, the system is describing the cat’s leg as brown because it can’t localize the leg properly. Figure courtesy of Devi Parikh, produced by a system described in “Making the V in VQA Matter: Elevating the Role of Image Understanding in Visual Question Answering” by Goyal, Khot, Summers-Stay, Batra, and Parikh and published in CVPR 2017.

Sentence generation

Decode features into sentence (with LSTM, etc)

- essentially classification with funky taxonomy

A baby eating a piece of food in his mouth.

A young boy eating a piece of cake

Aneja et al, 2018

doesn’t always work…

And scoring system is easily subverted!

A small bird is perched on a branch

A small brown bear is sitting in the grass

Aneja et al, 2018

Can train encoder *without labels*

Encoder yields embedding of the image

Exploit data augmentation

- take image and
  - crop+resize; adjust colormap; etc

Strategy: Contrastive learning

- Adjust embedding so that
  - A and Augment(A) should be close
  - A and B should be far

Then multiclass logistic regression when you have labels

SOA - rough summary

Very high accuracy with 1000's of classes

- Using
  - very deep residual networks
- clever trick to improve training convergence
  - alternative feature construction methods

Classification wrt

- Object present
- Scene type
- Etc

Challenges

- tough with little training data (but encoders are somewhat interchangeable)
- change in dataset presents problems

Open questions

Rules of machine learning

- It all works when test data is “like” training data
  - IID samples from the same distribution
- All bets are off otherwise; very little theoretical support

Practice in computer vision

- It is tough to tell when this condition occurs
- Mostly, it isn’t imposed
  - instead, we say that there was a generalization failure when classifier doesn’t work

Q: Why don’t we get in trouble when we break the rules?

Q: Tell when datasets A, B are “compatible”

- In a crisp, formal way (rather than try and see)

Exploiting registration and classification

Use a classifier to tell:

- how far to the next intersection?
- what is it like?
- is there a bike lane?
- etc.

Pred = 18.5 m

Road layout maps

Potential cues

- streetview
- openmaps

Partially supervised cues

Open Street Maps (OSM)

Map data: OpenStreetMap is an open-source mapping project covering over 21 million miles of road. Unlike proprietary maps, the underlying road coordinates and metadata are freely available for download. Accuracy and overlap with Google Maps is very high, though some inevitable noise is present as information is contributed by individual volunteers or automatically extracted from users' GPS trajectories. For example, roads in smaller cities may lack detailed annotations (e.g., the number of lanes may be unmarked). These inconsistencies result in varying-sized subsets of the data being applicable for different attributes.

Seff+Xiao

Fig. 3. Intersection detection heatmap. Images are cropped from test set GSV panoramas in the direction of travel indicated by the black arrow. The probabilities of "approaching" an intersection output by the trained ConvNet are overlaid on the road. (The images are from the ground level road, not the bridge.)

Seff+Xiao

Partially supervised cues

Google street view

Image collection: Google Street View contains panoramic images of street scenes covering 5 million miles of road across 3,000 cities. Each panorama has a corresponding metadata file storing the panorama's unique "pano_id", geographic location, azimuth orientation, and the pano_ids of adjacent panoramas. Beginning from an initial seed panorama, we collect street view images by running a bread-first search, downloading each image and its associated metadata along the way. Thus far, our dataset contains one million GSV panoramas from the San Francisco Bay Area. GSV panoramas can be downloaded at several different resolutions (marked as "zoom levels"). Finding the higher zoom levels unnecessary for our purposes, we elected to download at a zoom level of 1, where each panorama has a size of 832×416 pixels.

Seff+Xiao

Labelling - I

Match panoramas to roads

- panorama center location, orientation is known
- (essentially) project to plane
- thresholded nearest neighbor to road center polyline
  - thresholding removes panoramas inside buildings, etc.
- some noise
  - under bridges, etc.

Annotations

- Intersections
- Drivable heading
- Heading angle
- Bike lane
- Speed limit, wrong way, etc.

Pred = 0.1 m
True = 1.9 m

Pred = 18.5 m
True = 19.2 m

Pred = 22.9 m
True = 22.4 m

Fig. 4. Distance to intersection estimation. For images within 30 m of true intersections, our model is trained to estimate the distance from the host car to the center of the intersection across a variety of road types.

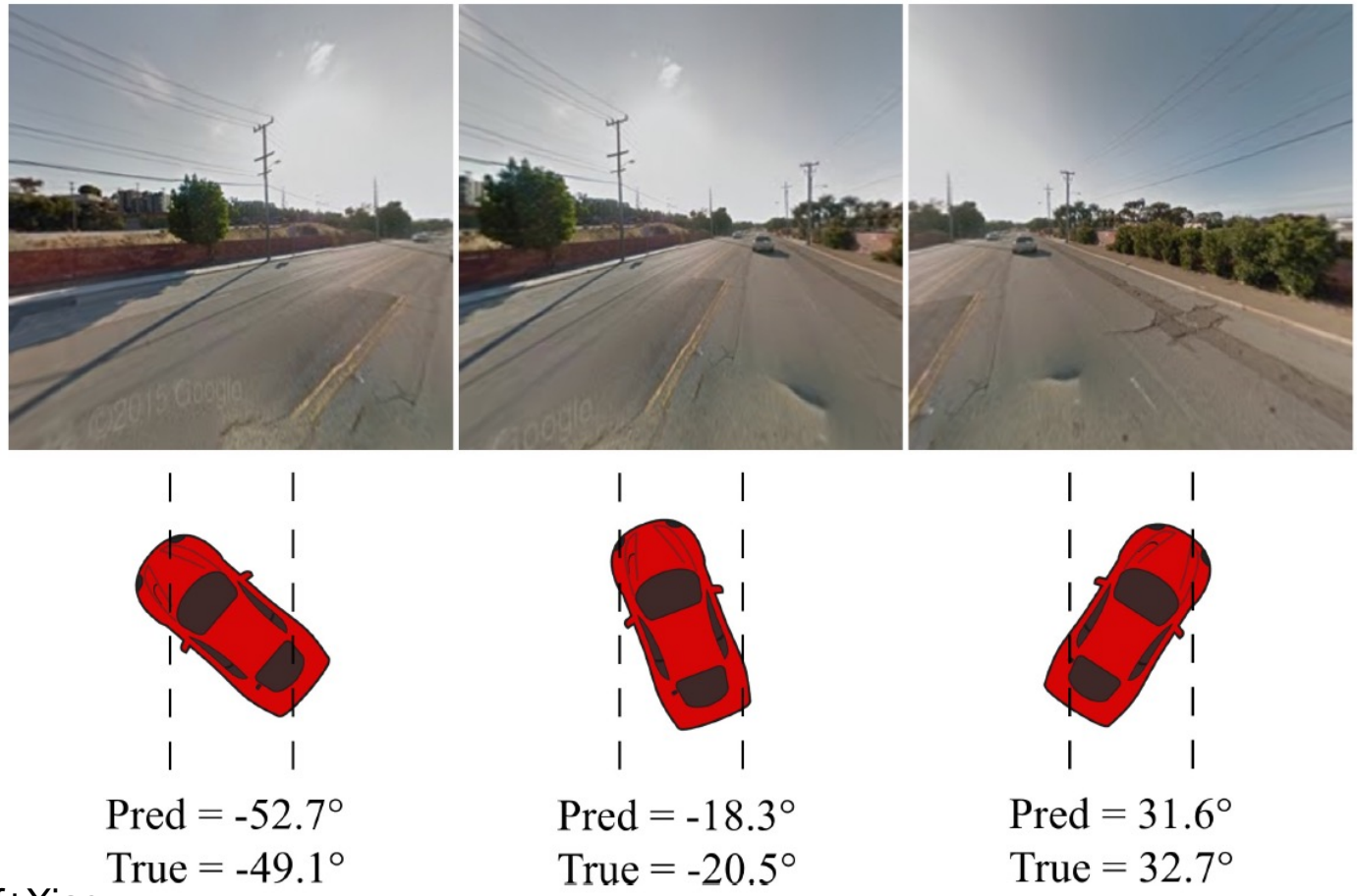
Seff+Xiao

Fig. 5. Intersection topology is one of several attributes our model learns to infer from an input GSV panorama. The blue circles on the Google Maps extracts to the left show the locations of the input panoramas. The pie charts display the probabilities output by the trained ConvNet of each heading angle being on a driveable path (see Figure 3 for colormap legend).

Seff+Xiao

Fig. 6. Driveable headings. A ConvNet is trained to distinguish between non-drivable headings (left) and drivable headings aligned with the road (right). The ConvNet weakly classifies the middle example as drivable because the host car's heading is facing the alleyway between the buildings.

Seff+Xiao



Seff+Xiao

Fig. 7. Heading angle regression. The network learns to predict the relative angle between the street and host vehicle heading given a single image cropped from a GSV panorama. Below each GSV image, the graphic visualizes the ground truth heading angle.

p(bike lane) = 0.043 p(bike lane) = 0.604 p(bike lane) = 0.988

Fig. 8. The ConvNet learns to detect bike lanes adjacent to the vehicle. The GSV images are arranged from left to right in increasing order of probability output by the ConvNet of a bike lane being present (ground truth labels from left to right are negative, negative, positive). The middle example contains a taxi lane, resulting in a weak false positive.

Seff+Xiao

Pred = 26.1 mph
True = 30 mph

Pred = 30.0 mph
True = 50 mph

Pred = 54.3 mph
True = 50 mph

Fig. 9. Speed limit regression. The network learns to predict speed limits given a GSV image of road scene. The model significantly underestimates the speed limit in the middle example as this type of two-way road with a single lane in each direction would generally not have a speed limit as high as 50 mph.

Seff+Xiao

p(one-way) = 0.207 p(one-way) = 0.226 p(one-way) = 0.848

Fig. 10. One-way vs. two-way road classification. The probability output by the ConvNet of each GSV scene being on a one-way road is shown. From left to right the ground truth labels are two-way, two-way, and one-way. The image on the left is correctly classified as two-way despite the absence of the signature double yellow lines.

Seff+Xiao

p(wrong way) = 0.555 p(wrong way) = 0.042 p(wrong way) = 0.729

Fig. 11. Wrong way detection. The probability output by the ConvNet of each GSV image facing the wrong way on the road is displayed. From left to right the ground truth labels are wrong way, right way, and right way. For two-way roads with no lane markings (left), this is an especially difficult problem as it amounts to estimating the horizontal position of the host car. The problem can also be quite ill-defined if there are no context clues as is the case with the rightmost image.

Seff+Xiao

Pred = 2	Pred = 2	Pred = 3
True = 1	True = 2	True = 2

Fig. 12. Number of lanes estimation. The predicted and true number of lanes for three roads are displayed along with the corresponding GSV images. For streets without clearly visible lane markings (left), this is especially challenging. Although the ground truth for the rightmost image is two lanes, there is a third lane that merges just ahead.

Seff+Xiao

At this point

I can tell from an image whether

- I'm pointing in the right direction
- going the right way
- facing an intersection
- available turns, etc.
- what and where street signs are
- …

Can I build a reliable controller?

BIG GOOD QUESTIONS

Mashup of openmaps and street view

- it could predict drivable directions, steering directions, lanes, signs, etc.

Q: WHY IS THIS NOT DRIVING AROUND NOW?

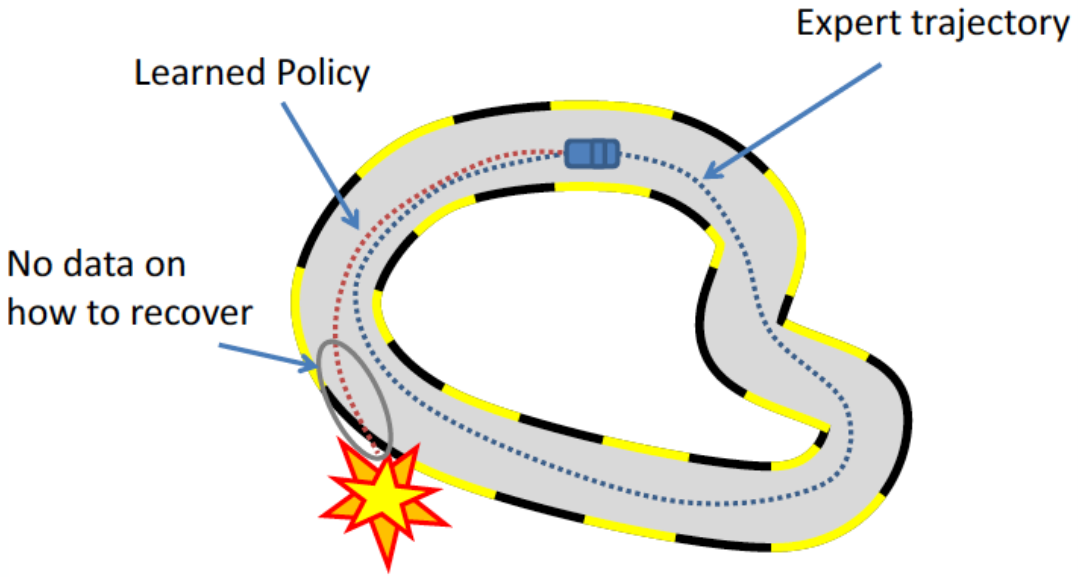
- A: (pretty obviously) because it doesn't work

Q: WHY NOT?

- A: interesting

Data Distribution Mismatch!

$$p_{\pi^*}(o_t) \neq p_{\pi_\theta}(o_t)$$

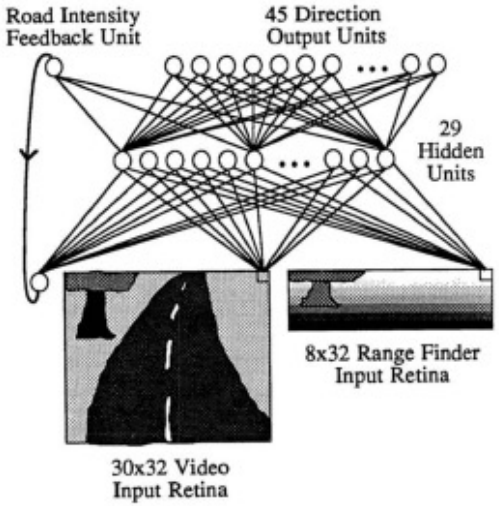


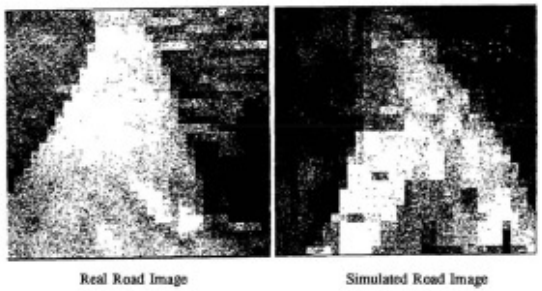
Fragkiadaki, ND

Imitation learning

Approaches

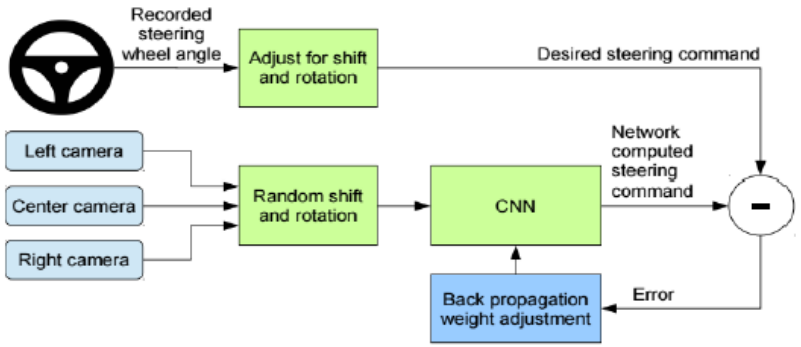
- Imitation learning:
  – Train a policy that does “the same thing” as an expert

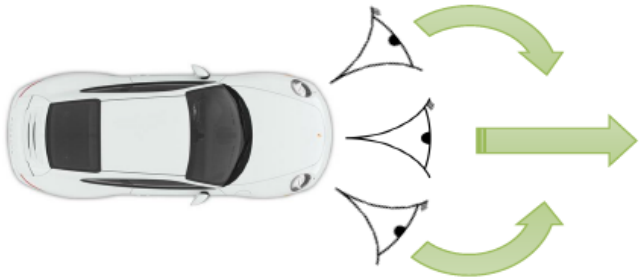




“In addition, the network must not solely be shown examples of accurate driving, but also how to recover (i.e. return to the road center) once a mistake has been made. Partial initial training on a variety of simulated road images should help eliminate these difficulties and facilitate better performance.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

Demonstration Augmentation: NVIDIA 2016





Additional, left and right cameras with automatic grant-truth labels to recover from mistakes

"*DAVE-2 was inspired by the pioneering work of Pomerleau [6] who in 1989 built the Autonomous Land Vehicle in a Neural Network (ALVINN) system*. Training with data from only the human driver is not sufficient. The network must learn how to recover from mistakes. …",

End to End Learning for Self-Driving Cars , Bojarski et al. 2016

Fragkiadaki, ND