CS375 / Psych 249: Large-Scale Neural Network Models for Neuroscience

Lecture 3: Deep CNNs and the Ventral Visual stream
2017.10.02

Daniel Yamins
Stanford Neurosciences Institute
Stanford Artificial Intelligence Laboratory
Departments of Psychology and Computer Science
Stanford University
Problem: Entity Extraction

"Core object perception" regime

Chance is 50%

All the data I will show you today

Typical primate fixation duration during natural viewing

Basic level categorization

Stimulus Duration (milliseconds)

Accuracy (% correct)

Chance is 50%

Amount of variation (d') Performance

Basic level categorization

Car identification

Face identification

Humans (population pooled)

“Core object perception” regime

Typical primate fixation duration during natural viewing

All the data I will show you today

Amount of variation (d') Performance
Adapted from DiCarlo et al. 2012
GOAL: Predictive model of single-neuron responses throughout the ventral stream to arbitrary image stimuli.
Hierarchical Convolutional Neural Networks

- Linear-Nonlinear neurally-plausible **basic operations** within layer

**Layer components are basic neural-like operations.**

**Linear**
- Filter
  - $\otimes \Phi_1$
  - $\otimes \Phi_2$
  - $\otimes \Phi_k$

**Nonlinear**
- Threshold & Saturate
- Pool
- Normalize

**Data:**
- untangling through dimension expansion
- “AND” operation by limiting dynamic range
- adding robustness by dimension reduction
- put results back into standard range

**Neuro:**
- synaptic weights patterns
- single-unit activations
- complex cells
- competitive inhibition

---

Linear-Nonlinear neurally-plausible basic operations within layer.
Neural Fitting Strategy?

Huge number of parameters consistent with HCNN concept.

i. **architectural** params: (# layers, # filters, receptive field sizes, &c) — “network structure”

ii. **filter** parameters: continuous valued pattern templates — “network contents”

*Q: How to discover the “right” parameters to understand real cortex?*
Models of V1

Where did this come from?

(1) “Hubel and Wiesel’s Intuition”
~1970s and formalized later

can just “see” what the right axes for measuring good tuning curves are, if we’re smart enough

\[
\text{Gaussian tuning curve of V1 simple cell}
\]

adapted from Adrienne Fairhall
Sparse Coding

Foldiak, Olshausen, mid 1990s

→ neurons have to represent their environment, as efficiently as possible

\[ L(x) = |x - O(H(x))|^2 + \lambda \cdot |H(x)| \]
Bottom-Up “Tuning Curve” Approach

V2:

Original photographs
V1-like filters matched: spectrally matched noise
Correlations matched: naturalistic texture

So, maybe a hierarchically-built sparse auto-encoding in a 2-layer model with max pooling?? … but doesn’t really work well in practice.

V4:

V4 Responses to Non-Cartesian Gratings
Gallant et al. 1996

Problem:
No predictions for any other images.
i.e.
is not an “image-computable” model
Bottom-Up “Tuning Curve” Approach

IT:

1 mm

Tsunoda et al.
Bottom-Up “Tuning Curve” Approach

(1) “Hubel and Wiesel’s Intuition”
~1970s and formalized later

→ e.g. there is a “fixed basis set”
that just “makes sense” if we’re
smart enough.

REALLY HARD TO GENERALIZE
TO MULTI-LAYER NETWORKS

(2) Sparse Coding Foldiak, Olshausen,
mid 1990s

→ neurons have to represent their
environment, as efficiently as possible
Neural Fitting Strategy?

Huge number of parameters consistent with HCNN concept.

Obvious alternative strategy: fit parameters to neural data.
Neural Fitting Strategy?

Huge number of parameters consistent with HCNN concept.

...not enough neural data to constrain model class. Gallant (2007); Rust & Movshon (2006)
Neural Fitting Strategy?

Huge number of parameters consistent with HCNN concept.

...not enough neural data to constrain model class. Gallant (2007); Rust & Movshon (2006)

Overfitting.
Optimize for Performance, Test Against Neurons

Visual Recognition Task

Step 1: Optimize for Task
Optimize for Performance, Test Against Neurons

Step 1: Optimize for Task

Spatial Convolution over Image Input

100ms Visual Presentation

LN
LN ...
LN
LN ...

Visual Recognition Task

Step 2: Compare to Neural Data

V1
IT
V2
V4

Optimize for Performance, Test Against Neurons
1. **Performance**: accuracy on a challenging, high-variation visual object categorization task.

2. **Neural predictivity**: the ability of model to predict each individual neural site’s activity.
1. **Performance**: accuracy on a challenging, high-variation visual object categorization task.

2. **Neural predictivity**: the ability of model to predict each individual neural site's activity.

*challenging for neural network engineers, not the animal*
1. **Performance**: accuracy on a challenging, high-variation* visual object categorization task.

2. **Neural predictivity**: the ability of model to predict each individual neural site’s activity.

Our hypothesis: Performance (1) $\rightarrow$ neural predictivity (2).

*challenging for neural network engineers, not the animal*
Multi-array electrophysiology in macaque V4 and IT.
Multi-array Electrophysiology Experiment

5760 images

64 objects

8 categories

uncorrelated photo backgrounds

Animals  Boats  Cars  Chairs  Faces  Fruits  Planes  Tables

Low variation

Medium variation

High variation

Pose, position, scale, and background variation
Multi-array Electrophysiology Experiment

About 300 total sites

Output = Binned spike counts 70ms-170ms post stimulus presentation averaged over 25-50 reps of each image.
Multi-array Electrophysiology Experiment

Responses to 1600 test images of two example units

IT unit 53

Images sorted first by **category**, then **variation level**.
IT Neurons Track Human Performance

V4 loses out at higher variation:

... but humans are much less affected.

... as is the IT neural population.

At **high variation levels**, IT much better than V4 and existing models.
Neural predictivity: the ability of model to predict each individual neural site’s activity.
Neural Response Prediction

Some kind of mapping is necessary.
Here, we use linear regression.

\[ T = M \ast S \]
Neural predictivity: the ability of model to predict each individual neural site’s activity.

Neural site unit ~ sparse linear combination of model units

Linear regression with fixed training images.

Accuracy = goodness-of-fit on held-out testing images (Cross validated)

Neural predictivity = median accuracy over all units.
Initial Validation of Idea

High-throughput experiments to directly test the relationship between performance and IT neural predictivity.

- Random selection of model parameters; measure performance and neural predictivity  
  Pinto et al. (2008, 2009)
Initial Validation of Idea

random

\[ r = 0.55 \pm 0.08 \]

(n=2000)

Yamins* and Hong* et. al. PNAS (2014)

different model
(architectural params)
Initial Validation of Idea

High-throughput experiments to directly test the relationship between neural predictivity and performance.

- Random selection of model parameters; measure performance and neural predictivity  
  Pinto et. al (2008, 2009)

- Optimize parameters for performance; measure neural predictivity.  
  Optimization techniques: Bergstra Yamins & Cox (2013)
Initial Validation of Idea

Yamins* and Hong* et. al. *PNAS* (2014)

performance-optimized

$r = 0.79 \pm 0.05$

(n=2000)
Initial Validation of Idea

High-throughput experiments to directly test the relationship between neural predictivity and performance.

- Random selection of model parameters; measure performance and neural predictivity  
  Pinto et al. (2008, 2009)

- Optimize parameters for performance; measure neural predictivity  
  optimization techniques: Bergstra Yamins & Cox (2013)

- Optimize parameters for neural predictivity; measure performance
Performance vs IT predictivity: Predictivity-Optimized

\[ r = 0.80 \pm 0.04 \]
\[ (n=2000) \]

Yamins* and Hong* et. al. *PNAS* (2014)
Performance vs IT predictivity: Predictivity-Optimized

Performance is a potentially very good driver of neural prediction.

$\text{Yamins}^*$ and Hong$^*$ et. al. \textit{PNAS} (2014)

$r = 0.55 \pm 0.08$

$r = 0.79 \pm 0.05$

$r = 0.80 \pm 0.04$
Performance vs IT predictivity

Yamins* and Hong* et. al. PNAS (2014)
Performance vs IT predictivity

But, not doing that well. Really want to be here:
Optimization Strategy

i. architectural params: (# layers, # filters, receptive field sizes, &c) — “network structure”

→ Automated meta-parameter optimization in high-dimensional discrete parameter spaces  
   Bergstra Yamins & Cox (2013)

→ Ensembles of models chosen through modified boosting  
   Yamins et al (2013, 2014)
Optimization Strategy

i. **Architectural** params: (# layers, # filters, receptive field sizes, &c) — “network structure”

→ Automated meta-parameter optimization in high-dimensional discrete parameter spaces
  Bergstra Yamins & Cox (2013)

→ Ensembles of models chosen through modified boosting  Yamins et. al (2013, 2014)

ii. **Filter** parameters: continuous valued pattern templates — “network contents”

→ GPU-accelerated stochastic gradient descent  Pinto et. al., (2009), Krizhevsky et. al. (2012)

Gradient descent eq: \[ \frac{dp}{dt} = -\lambda(t) \cdot \frac{\partial L}{\partial P} \]
\[ L = \text{loss function} \]
\[ \lambda = \text{learning rate} \]

In current practice:

\[ L = \text{loss computed from large numbers of externally-provided object category labels.} \]
Model Training Regimen

**Model Training Regimen**

**train:** real photos
Model Training Regimen

**train:** real photos

**test:** neural stimuli

generalize?
Model Training Regimen

**train:** real photos

**test:** neural stimuli

removed categories of photos that appeared in the test stimuli

(animals, boats, cars, chairs, faces, fruits, planes, tables)
**Model Training Regimen**

**train:** real photos

**test:** neural stimuli

![Tree map visualization of ImageNet](image)

- removed categories of photos that appeared in the test stimuli (animals, boats, cars, chairs, faces, fruits, planes, tables)

→ Specific 4-layer model that achieved high recognition performance.
Performance Generalization

Basic categorization

Clouds

Train Set

ImageNet 1K Test Split

Samples

Images

Images
At high variation levels, IT much better than V4 and existing models

Performance Comparison

Yamins* and Hong* et. al. *PNAS* (2014)
Performance Comparison

At high variation levels, IT much better than V4 and existing models

New model comparable to IT / human performance levels.

Yamins* and Hong* et. al. *PNAS* (2014)
Performance

Low Variation Tasks
- Pixels
- SIFT
- V1-like
- V2-like
- HMAX
- PLOS09
- HMO
- V4 Neurons
- IT Neurons
- Human

Medium Variation Tasks
- Basic categorization
- Car identification
- Face identification

High Variation Tasks

Performance Comparison

Yamins* and Hong* et. al. *PNAS* (2014)
Does it predict neurons better?
Does it predict neurons better?

Images sorted first by category, then variation level.

Yamins* and Hong* et. al. *PNAS* (2014)
Does it predict neurons better?

unit 53

Yamins* and Hong* et. al. PNAS (2014)

Images sorted first by category, then variation level.

Response Magnitude

Animals  Boats  Cars  Chairs  Faces  Fruits  Planes  Tables

neural data

model prediction

r^2=0.55
Predicting IT Neural Responses

Images sorted first by **category**, then **variation level**.

- **IT Site 150**
- **IT Site 56**
- **IT Site 42**

Response Magnitude

- **Neural data**
- **Model prediction**
Key Underlying Principle

$\rho = 0.87 \pm 0.15$

Yamins* and Hong* et. al. *PNAS* (2014)
Predicting IT Neural Responses

What about intermediate layers?

i. compare intermediate model layers to IT neural data

ii. compare all model layers to intermediate visual areas (V4)
Captures low variation image response patterns …
Captures low variation image response patterns …
Layer 1

Neural data
Model prediction

... but fails to capture higher variation response patterns.
Building tolerance while maintaining selectivity
Predicting IT Neural Responses

Yamins* and Hong* et. al. PNAS (2014)
Predicting IT Neural Responses

Yamins* and Hong* et. al. *PNAS* (2014)
Predicting IT Neural Responses

Yamins* and Hong* et. al. *PNAS* (2014)
Predicting IT Neural Responses

Yamins* and Hong* et. al. PNAS (2014)

IT Explained Variance (%)

Performance constraints

Architectural constraints

IT Explained Variance (%)

Ideal Observers

Control Models

HMO Layers

Category

All Variables

Architectural constraints

predicting

Performance constraints

IT Explained Variance (%)
Predicting IT Neural Responses

Performance constraints + architectural constraints → better neural prediction
Predicting IT Neural Responses

What about intermediate layers?

i. compare intermediate model layers to IT neural data

ii. compare all model layers to intermediate visual areas (V4)
Predicting V4 Neural Responses

V4 unit 60

Animals  Boats  Cars  Chairs  Faces  Fruits  Planes  Tables
Predicting V4 Neural Responses

V4 unit 60

Top Layer

Layer 1

Animals  Boats  Cars  Chairs  Faces  Fruits  Planes  Tables

Neural data  Model prediction
Predicting V4 Neural Responses

- Top Layer
- Layer 3
- Layer 2
- Layer 1

Neural data
Model prediction

Animals  Boats  Cars  Chairs  Faces  Fruits  Planes  Tables
Predicting V4 Neural Responses

Yamins* and Hong* et. al. PNAS (2014)
Investigating fits as a function of model layer:

IT fit increases at each layer. In contrast, V4 fit peaks and then goes down.

![Graph showing IT and V4 predictivity across model layers.](image)

Yamins* and Hong* et. al. *PNAS* (2014)
Model output at lowest layer resembles Gabor wavelets:

Layer 1 Filters

In submission: model lowest layer is best explanation of imaging data in V1. (with Darren Seibert and Justin Gardner)

Low $M_{ij}$ (blue) means neurons think the stimuli are similar.

High $M_{ij}$ (red) means neurons think the two stimuli are different.

$$M_{ij} = 1 - \text{correlation}(r_i, r_j)$$
RDMs allow comparison of different neural representations on a common stimulus set.

Their structure echoes key features of the functionality of the population code.
Their structure echoes key features of the functionality of the population code.
The Population Code

Model captures diagonal and off-diagonal structure effectively.
The Population Code

Perspective on hierarchical processing in the ventral stream.

Pixels  V1-like  V2-like  V4  IT

HMO Model
The Population Code

HMO

IT Neurons

Image Generalization

Object Generalization

Category Generalization

Animals (8)
Boats (8)
Cars (8)
Chairs (8)
Faces (8)
Fruits (8)
Planes (8)
Tables (8)

Animals (4)
Boats (4)
Cars (4)
Chairs (4)
Faces (4)
Fruits (4)
Planes (4)
Tables (4)

Faces (8)
Fruits (8)
Planes (8)
Tables (8)

Similarity to IT at High Variation

Spearman r of RDMs

image generalization
object generalization
category generalization

Controls
HMO
V4 Neurons
IT Neuron Split-Half
Behavioral “Top-Down” constraints

Complement standard “from below” approach ...
Behavioral “Top-Down” constraints

Complement standard “from below” approach …
Behavioral “Top-Down” constraints

Complement standard “from below” approach …
Behavioral “Top-Down” constraints

Complement standard “from below” approach …
Behavioral “Top-Down” constraints

Complement standard “from below” approach … with behavioral constraints
Behavioral “Top-Down” constraints

Complement standard “from below” approach … with behavioral constraints
Behavioral “Top-Down” constraints

Complement standard “from below” approach … with behavioral constraints
Complement standard “from below” approach ... with behavioral constraints
Beyond categorization

**Category**: plane

**Identity**: f16
Beyond categorization

Position
Beyond categorization
Beyond categorization

Aspect Ratio and Angle
Beyond categorization

We can quickly assess the scene as a whole.
Where and how are all these properties coded neurally?
Beyond categorization

“Standard word model” predicts: **not at the top of the ventral stream.**

Aggregation over identity-preserving transformations, e.g. translation.
Beyond categorization

“Standard word model” predicts: **not at the top of the ventral stream.**

Aggregation over identity-preserving transformations, e.g. translation.
“Standard word model” predicts: not at the top of the ventral stream.

Aggregation over identity-preserving transformations, e.g. translation.

- Receptive Field Size $\uparrow$
- Category Invariance $\uparrow$
- (e.g.) Position Sensitivity $\downarrow$
Where and how are all these properties coded neurally?
Somewhat newish ideas about IT?

State of knowledge from previous studies . . .

Population Decode Performance
(relative to human performance)

Depth Along Ventral Stream
(increasing receptive field size →)

Categorization
(known)

Orthogonal Properties

Multiple hypotheses consistent with the existing data . . . (known)
Somewhat newish ideas about IT?

State of knowledge from previous studies . . .

Categorization

Orthogonal Properties

Multiple hypotheses consistent with the existing data . . .

H1: Tolerance / sensitivity tradeoff?

Depth Along Ventral Stream
(increasing receptive field size →)

Population Decode Performance
(relative to human performance)
Somewhat newish ideas about IT?

State of knowledge from previous studies . . .

Multiple hypotheses consistent with the existing data . . .

H1

H2

H3: Information preservation?

Population Decode Performance

Depth Along Ventral Stream

(increasing receptive field size →)

Orthogonal Properties

(known)

Orthogonal Properties

??

Orthogonal Properties

??

Orthogonal Properties

??

Orthogonal Properties

??

Orthogonal Properties

(known)
Beyond categorization

Unexpected observation:

Training on categorization task 

Increased performance on position estimation task.

even though the goal was to become INVARIANT to position

Beyond categorization

Category optimization $\rightarrow$ improved performance on non-categorical tasks.

$\text{Hong}^*, \text{Yamins}^*, \text{Majaj} \& \text{DiCarlo. Nat. Neuro. (in press)}$
Beyond categorization

Unexpected observation #2:

Test Performance

Categorization

Model Layers

Layer 1
Layer 2
Layer 3
Layer 4
Layer 5
Layer 6

Unexpected observation #2:

Increased performance on position estimation task at each model layer.

Beyond categorization

For all tasks of visual interest we could measure in our test dataset:

- **Categorization**
- **X-Axis Position**
- **3-D Object Scale**
- **Z-Axis Rotation**

Performance on non-categorical tasks increases at each layer.
Beyond categorization

What do the data say?
Population Decoding

*category: plane
identity: f16*

Categorization

Identification

IT cortex
V1-like model
V4 cortex
pixel control

Population Decoding

category: plane
identity: f16

Categorization
Identification

X-axis Position
Y-axis Position


IT cortex
V4 cortex
V1-like model
pixel control
Population Decoding

**IT > V4, V1** for all tasks

**V4 > V1** for most tasks

---

**V4 > V1**

- **IT cortex**
- **V1-like model**
- **V4 cortex**
- **pixel control**

---

Single Site Responses

Best single position-encoding sites.

heat map value at $x, y =$ response averaged over all images where object center is in position $x, y$
Single Site Responses

Best single position-encoding sites.

heat map value at $x, y =$ response averaged over all images where object center is in position $x, y$
Population Decoding

“Standard” receptive field-mapping stimuli w/ position and orientation variation:

X-position

Y-position

Orientation
Population Decoding

V1 > V4, IT for “standard” tasks

Human Psychophysical Measurements

Click where the **boat** was!

6 learning trial(s) left after this.
Monkey Neurons vs Humans

performance \sim k \times \log(N)

Basic Categorization

Subordinate Identification

## Monkey Neurons vs Humans

<table>
<thead>
<tr>
<th></th>
<th>IT</th>
<th>V4</th>
<th>V1</th>
<th>Pix</th>
</tr>
</thead>
<tbody>
<tr>
<td>Basic Categorization</td>
<td>773 ± 185</td>
<td>2.2 × 10⁶</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Subordinate Identification</td>
<td>496 ± 93</td>
<td>4.4 × 10⁶</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>X-axis Position</td>
<td>1414 ± 403</td>
<td>5.2 × 10⁵</td>
<td>3.0 × 10⁷</td>
<td>—</td>
</tr>
<tr>
<td>Y-axis Position</td>
<td>918 ± 309</td>
<td>2.5 × 10⁴</td>
<td>8.7 × 10⁵</td>
<td>—</td>
</tr>
<tr>
<td>Bounding Box Size</td>
<td>322 ± 90</td>
<td>1.7 × 10⁴</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>X-axis Size</td>
<td>256 ± 87</td>
<td>9.8 × 10³</td>
<td>3.4 × 10⁷</td>
<td>—</td>
</tr>
<tr>
<td>Y-axis Size</td>
<td>237 ± 87</td>
<td>3.8 × 10³</td>
<td>9.5 × 10⁶</td>
<td>—</td>
</tr>
<tr>
<td>3-D Object Scale</td>
<td>401 ± 90</td>
<td>3.2 × 10⁴</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Major Axis Length</td>
<td>201 ± 70</td>
<td>1.1 × 10⁴</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Aspect Ratio</td>
<td>163 ± 61</td>
<td>951 ± 59</td>
<td>6.5 × 10³</td>
<td>—</td>
</tr>
<tr>
<td>Major Axis Angle</td>
<td>804 ± 136</td>
<td>3.2 × 10⁶</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Z-axis Rotation</td>
<td>1932 ± 1061</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Y-axis Rotation</td>
<td>369 ± 115</td>
<td>2.8 × 10⁵</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>X-axis Rotation</td>
<td>1570 ± 530</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
</tbody>
</table>

— = more than 10 billion sites required

Mean over tasks, human-parity for IT is at ~700 multi-unit trial-averaged sites.

## Monkey Neurons vs Humans

<table>
<thead>
<tr>
<th></th>
<th>IT (± SD)</th>
<th>V4 (× 10^6)</th>
<th>V1 (× 10^6)</th>
<th>Pix</th>
</tr>
</thead>
<tbody>
<tr>
<td>Basic Categorization</td>
<td>773 ± 185</td>
<td>2.2 ± 185</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Subordinate Identification</td>
<td>496 ± 93</td>
<td>4.4 ± 93</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>X-axis Position</td>
<td>1414 ± 403</td>
<td>5.2 ± 403</td>
<td>3.0 ± 403</td>
<td>—</td>
</tr>
<tr>
<td>Y-axis Position</td>
<td>918 ± 309</td>
<td>2.5 ± 309</td>
<td>8.7 ± 309</td>
<td>—</td>
</tr>
<tr>
<td>Bounding Box Size</td>
<td>322 ± 90</td>
<td>1.7 ± 90</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>X-axis Size</td>
<td>256 ± 87</td>
<td>9.8 ± 87</td>
<td>3.4 ± 87</td>
<td>—</td>
</tr>
<tr>
<td>Y-axis Size</td>
<td>237 ± 87</td>
<td>3.8 ± 87</td>
<td>9.5 ± 87</td>
<td>—</td>
</tr>
<tr>
<td>3-D Object Scale</td>
<td>401 ± 90</td>
<td>3.2 ± 90</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Major Axis Length</td>
<td>201 ± 70</td>
<td>1.1 ± 70</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Aspect Ratio</td>
<td>163 ± 61</td>
<td>951 ± 59</td>
<td>6.5 ± 59</td>
<td>—</td>
</tr>
<tr>
<td>Major Axis Angle</td>
<td>804 ± 136</td>
<td>3.2 ± 136</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Z-axis Rotation</td>
<td>1932 ± 1061</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Y-axis Rotation</td>
<td>369 ± 115</td>
<td>2.8 ± 115</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>X-axis Rotation</td>
<td>1570 ± 530</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
</tbody>
</table>

— = more than 10 billion sites required

Mean over tasks, human-parity for IT is at $\sim$350000 single-unit single-trial neurons.

Somewhat newish ideas about IT?

**State of knowledge from previous studies . . .**

- **Categorization**
  - (known)

- **Orthogonal Properties**
  - ??

**Multiple hypotheses consistent with the existing data . . .**

- **H1**

- **H2**

- **H3**

- **H4**: Simultaneous build-up of encoding

**Depth Along Ventral Stream**

*increasing receptive field size →*

Population Decode Performance

(relative to human performance)
Somewhat newish ideas about IT?

1. IT is *NOT* invariant. Strict generalization of simple-to-complex cells: **no**.

2. “Lower-level” properties are not that low-level — at least, with complex objects and backgrounds.

3. Categorization and non-categorical properties “go together” — *not* just that “not all (e.g.) position information is lost” (MacEvoy 2013, DiCarlo 2003)

Provides support to a hypothesis for what IT does:

“Inverting the generative model of the scene”
Principle of “Goal-Driven Modeling”
Principle of “Goal-Driven Modeling”

Heuristic of “Goal-Driven Modeling”
Heuristic of “Goal-Driven Modeling”

“Mercedes behind Lamborghini, on a field in front of mountains.”

“Hannah is good at compromising”
“Mercedes behind Lamborghini, on a field in front of mountains.”

“Hannah is good at compromising”
But what type of understanding is this?
But what type of understanding is this?

not saying this type of understanding is impossible …
> Formulate comprehensive model class (\textbf{CNNs})

> Formulate comprehensive model class (\textbf{CNNs})

> Choose challenging, ethologically-valid tasks (\textit{categorization})
> Formulate comprehensive model class (**CNNs**)

> Choose challenging, ethologically-valid tasks (**categorization**)

> Implement generic learning rules (**gradient descent**)

> Formulate comprehensive model class (**CNNs**)

> Choose challenging, ethologically-valid tasks (**categorization**)

> Implement generic learning rules (**gradient descent**)

> Map to brain data. (**ventral stream**)

\[
\text{argmin}_{a \in \mathcal{A}} [L(p^*_a)]
\]

where \( p^* \) is result of

\[
\frac{dp_a}{dt} = -\lambda(t) \cdot \langle \nabla_{p_a} L(x) \rangle_{x \in \mathcal{D}}
\]

- **A** = architecture class
- **L** = loss function
- **D** = dataset
3. \[
\text{argmin}_{a \in \mathcal{A}} [L(p^*_a)] \\
\text{where } p^* \text{ is result of }
\]
\[
\frac{dp_a}{dt} = -\lambda(t) \cdot \langle \nabla_{p_a} L(x) \rangle_{x \in \mathcal{D}}
\]

“learning rule”

2. \[
L = \text{loss function} \quad D = \text{dataset}
\]

“task”

1. \[
A = \text{architecture class}
\]
Principle of “Goal-Driven Modeling”

Heuristic of “Goal-Driven Modeling”

… after all at some point, for any given task, you’ll probably “go over the hump” … perhaps when you exceed human performance or overfit on that task.
> Formulate comprehensive model class (CNNs)

> Choose challenging, ethologically-valid tasks (categorization)

> Implement generic learning rules (gradient descent)

> Map to brain data. (ventral stream)

> Formulate comprehensive model class (RNNs)

> Choose challenging, ethologically-valid tasks (task switching/ memory)

> Implement generic learning rules (??)

> Map to brain data. (Parietal cortex, PFC)