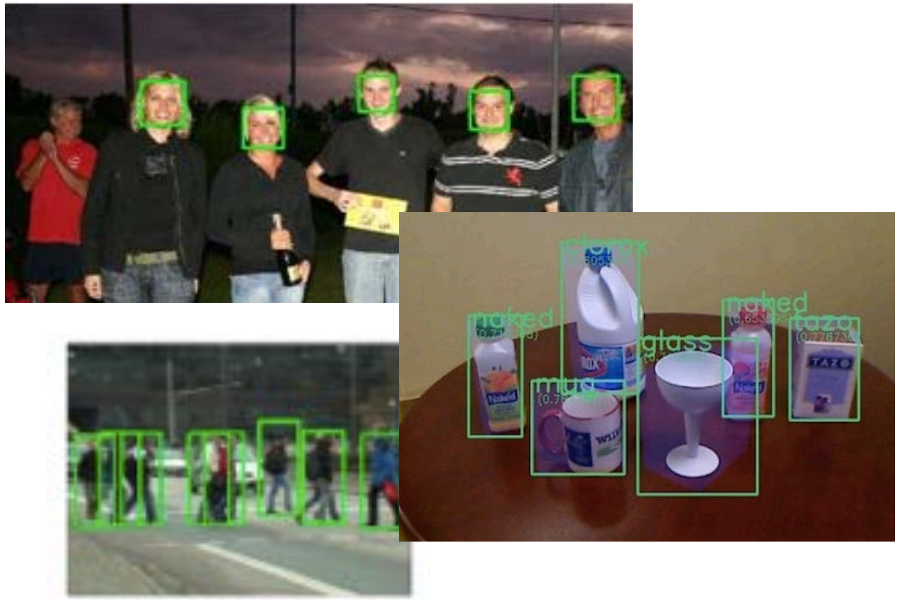
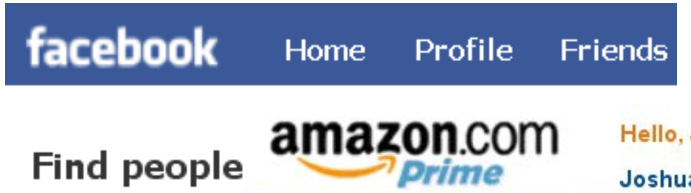
# Reverse engineering common sense

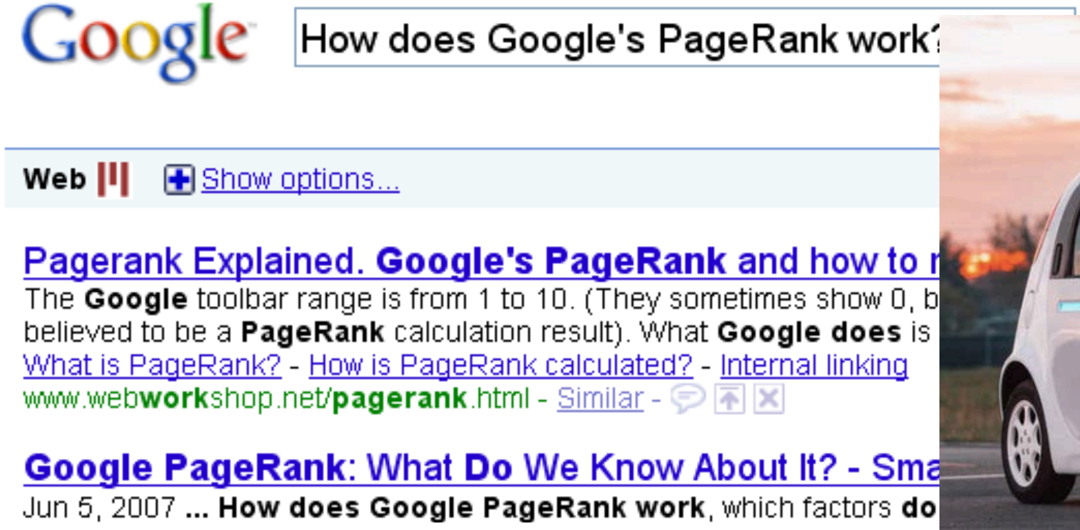
Josh Tenenbaum

# AI Technologies... but no real AI









# Where's the gap?

- Intelligence is not just about *pattern recognition* and *function approximation*.
- It is about *modeling the world*…
  - *explaining* and *understanding* what we see.
  - *imagining* things we could see but haven't yet.
  - *planning* actions and *solving problems* to make these things real.
  - *building new models* as we learn more about the world.

To read more: Lake, Ullman, Tenenbaum & Gershman, "Building machines that learn and think like people", on arXiv and *Behavioral and Brain Sciences* (2017).


PIANO 100
MINOSO
BLOKUS30
HUMAN30
CARDS8
CRAYOLA3
196.00

MIND

A QUARTERLY REVIEW

[October, 1950

# COMPUTING MACHINERY AND INTELLIGENCE

By A. M. Turing

In the process of trying to imitate an adult human mind we are bound to think a good deal about the process which has brought it to the state that it is in. We may notice three components,

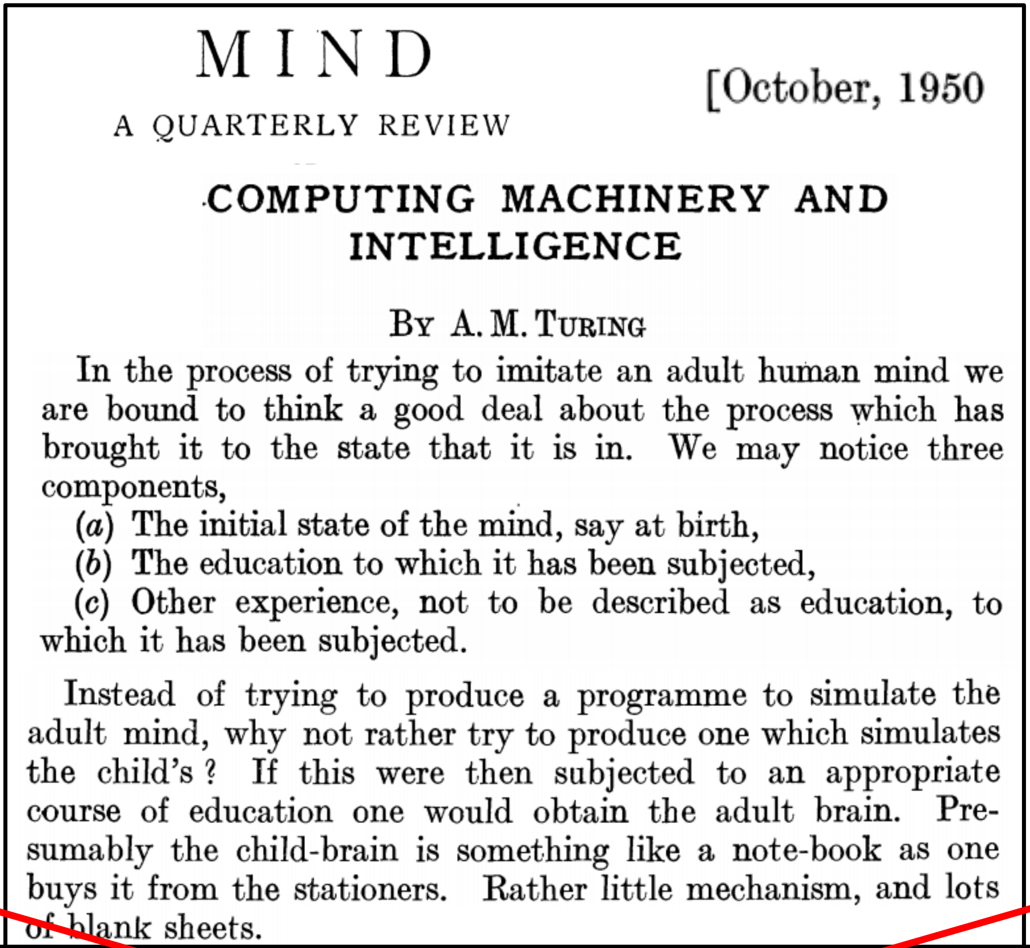
(*a*) The initial state of the mind, say at birth,

(*b*) The education to which it has been subjected,

(*c*) Other experience, not to be described as education, to which it has been subjected.

Instead of trying to produce a programme to simulate the adult mind, why not rather try to produce one which simulates

"Presumably the child-brain is something like a note-book as one buys it from the stationers. Rather little mechanism, and lots of blank sheets."



MIND

A QUARTERLY REVIEW

[October, 1950

COMPUTING MACHINERY AND INTELLIGENCE

By A. M. Turing

In the process of trying to imitate an adult human mind we are bound to think a good deal about the process which has brought it to the state that it is in. We may notice three components,

(*a*) The initial state of the mind, say at birth,
(*b*) The education to which it has been subjected,
(*c*) Other experience, not to be described as education, to which it has been subjected.

Instead of trying to produce a programme to simulate the adult mind, why not rather try to produce one which simulates the child's? If this were then subjected to an appropriate course of education one would obtain the adult brain. Presumably the child-brain is something like a note-book as one buys it from the stationers. Rather little mechanism, and lots of blank sheets.

"Presumably the child-brain is something like a note-book as one buys it from the stationers. Rather little mechanism, and lots of blank sheets."

**What is the the starting state (inductive bias)?**

More content than we might have thought, some of it very structured:

*"Core cognition"*

*"The game engine in your head"*

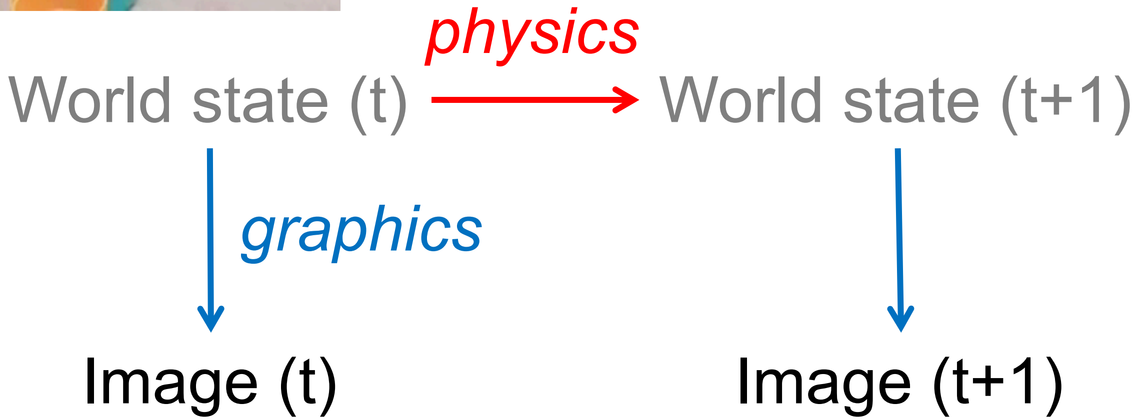
**What are the learning procedures?**

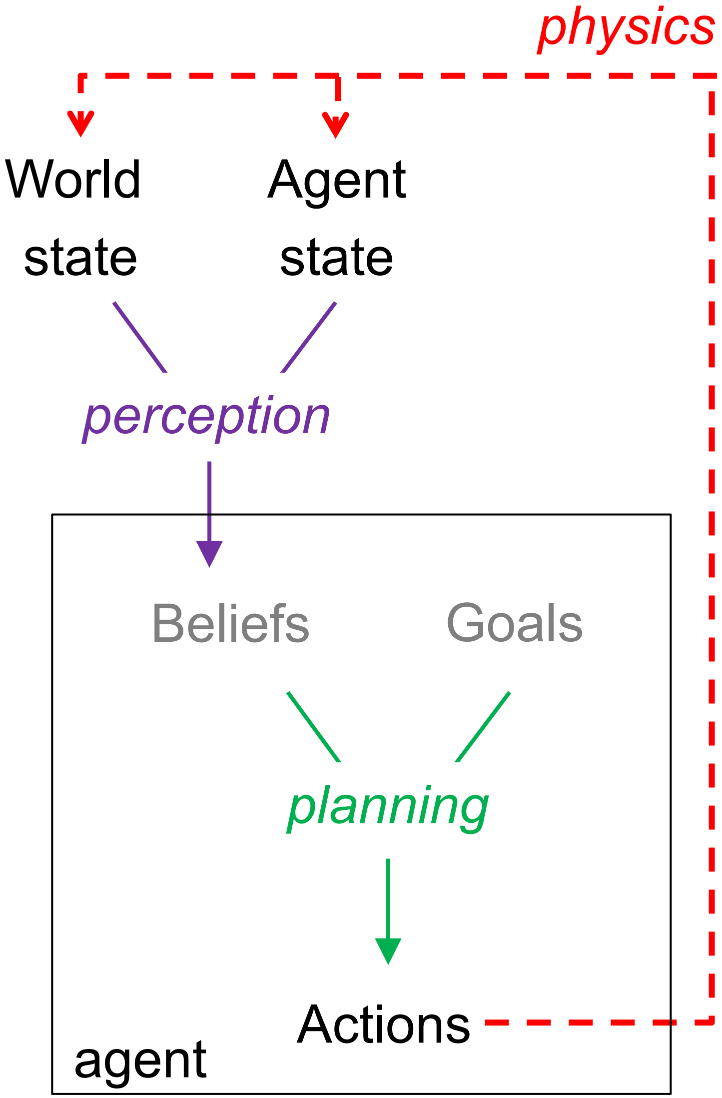
More mechanisms than we might have thought, some of them very smart:

*"The child as scientist"*

*"The child as coder"*

# Reverse-engineering the "Common Sense Core": Intuitive Physics, Intuitive Psychology

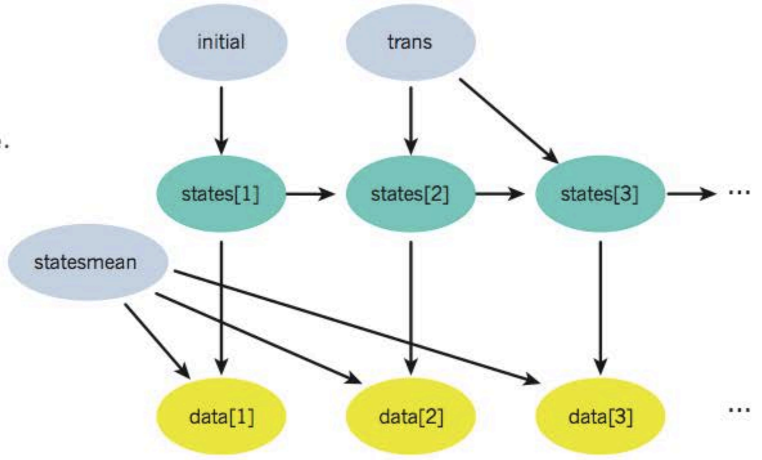
# How do we build this architecture?

*Probabilistic programs* integrate our best ideas on intelligence, across three different kinds of mathematics:

- **Symbolic manipulation (algebra, logic)** for representing and reasoning with abstract knowledge.
- **Bayesian inference (probability)** for reasoning about unobserved causes from sparse, uncertain data.
- **Neural networks (calculus)** for pattern recognition and function approximation.

```
statesmean = [-1, 1, 0]  # Emission parameters.
initial    = Categorical([1.0/3, 1.0/3, 1.0/3]) # Prob distr of state[1].
trans      = [Categorical([0.1, 0.5, 0.4]), Categorical([0.2, 0.2, 0.6]),
              Categorical([0.15, 0.15, 0.7])]   # Trans distr for each state.
data       = [Nil, 0.9, 0.8, 0.7, 0, -0.025, -5, -2, -0.1, 0, 0.13]

@model hmm begin # Define a model hmm.
 states = Array(Int, length(data))
 @assume(states[1] ~ initial)
 for i = 2:length(data)
   @assume(states[i] ~ trans[states[i-1]])
   @observe(data[i]  ~ Normal(statesmean[states[i]], 0.4))
 end
 @predict states
end
```

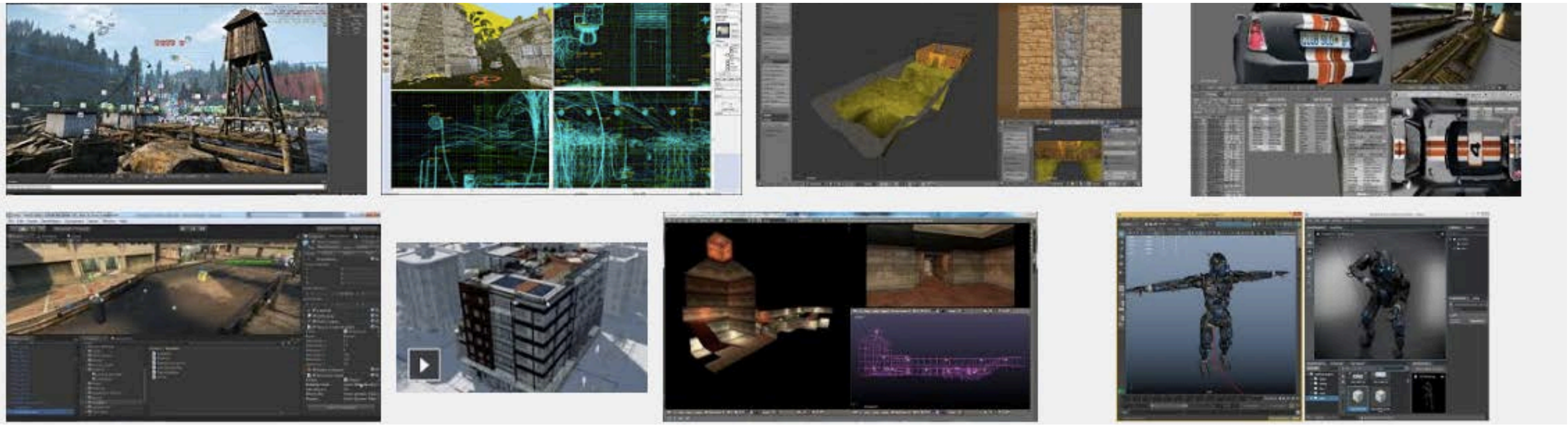


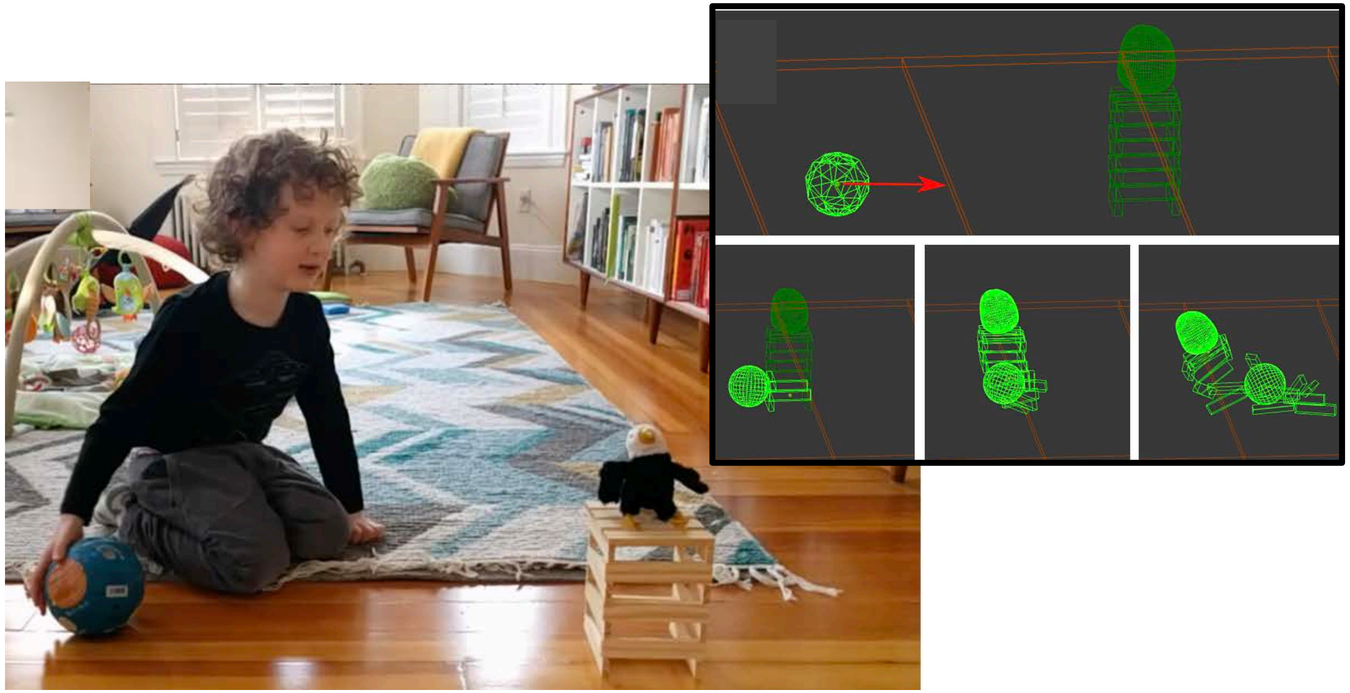

*Probabilistic Programming Languages:* Stan, Alchemy, BLOG, Church, Anglican, Edward, **Pyro, TensorFlow Probability, Gen**, ...

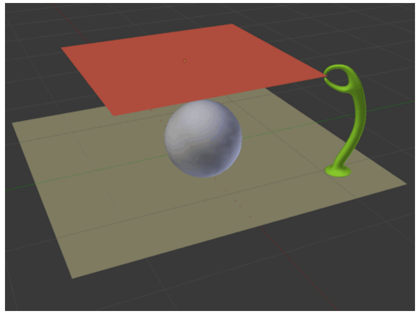
**See https://probmods.org**

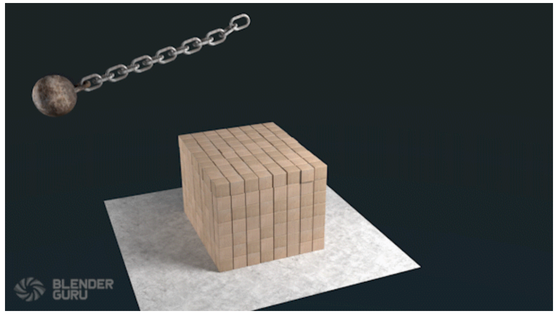
# How do we build this architecture?

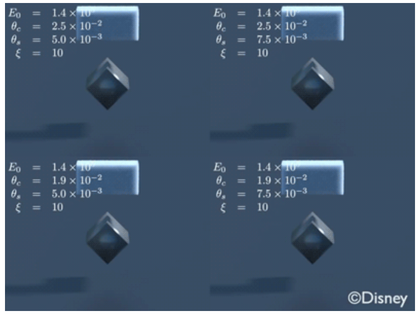
"The game engine in your head": Very fast, approximate programs for simulating graphics, physics, planning …





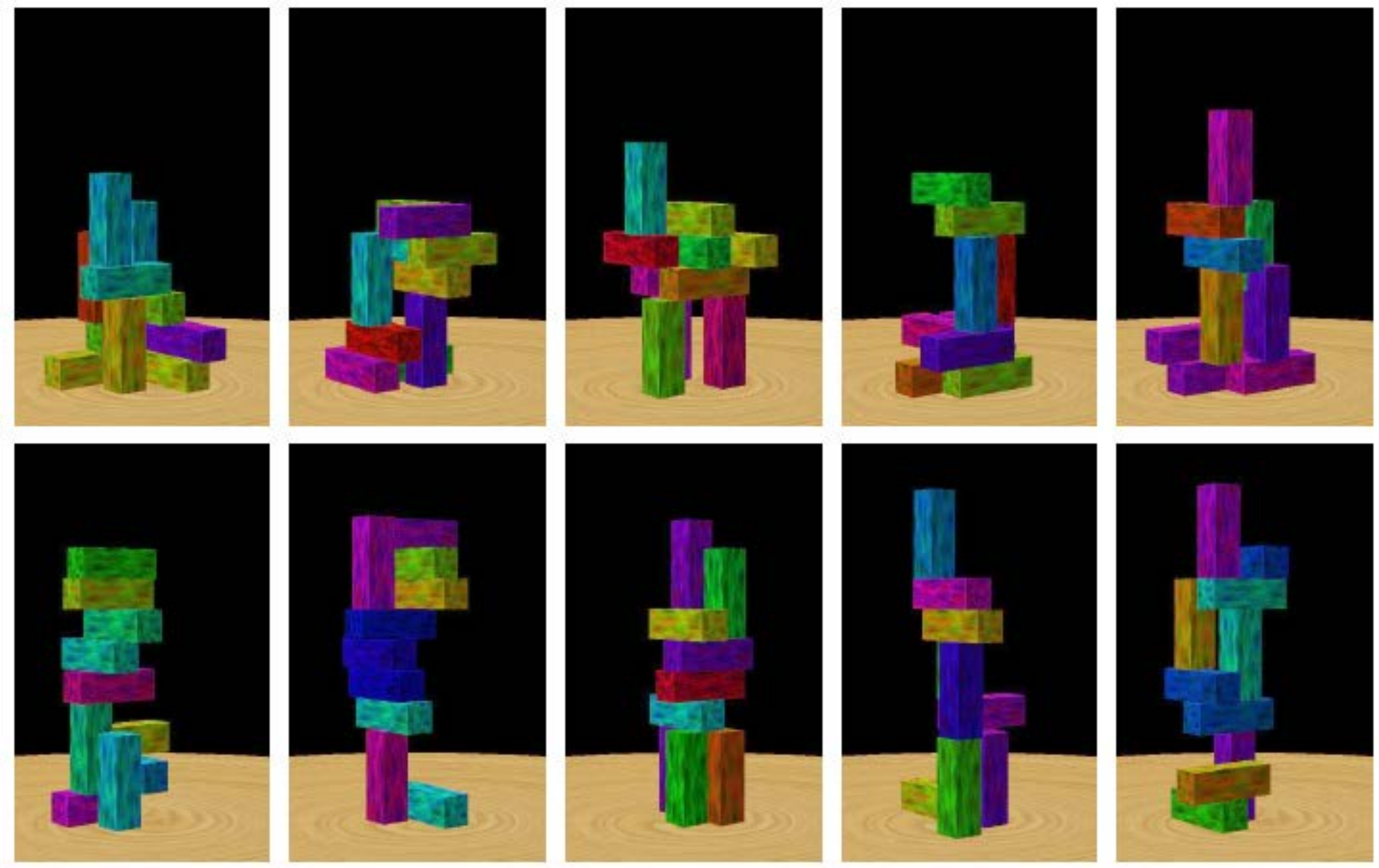




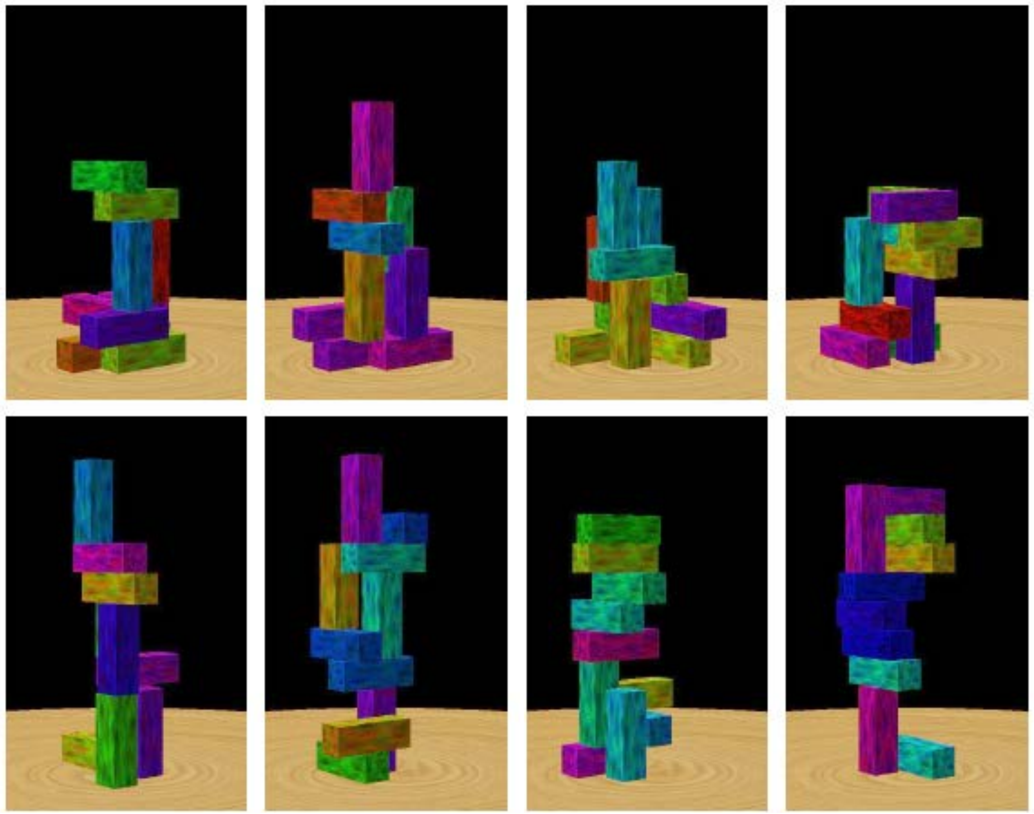
# The intuitive physics engine

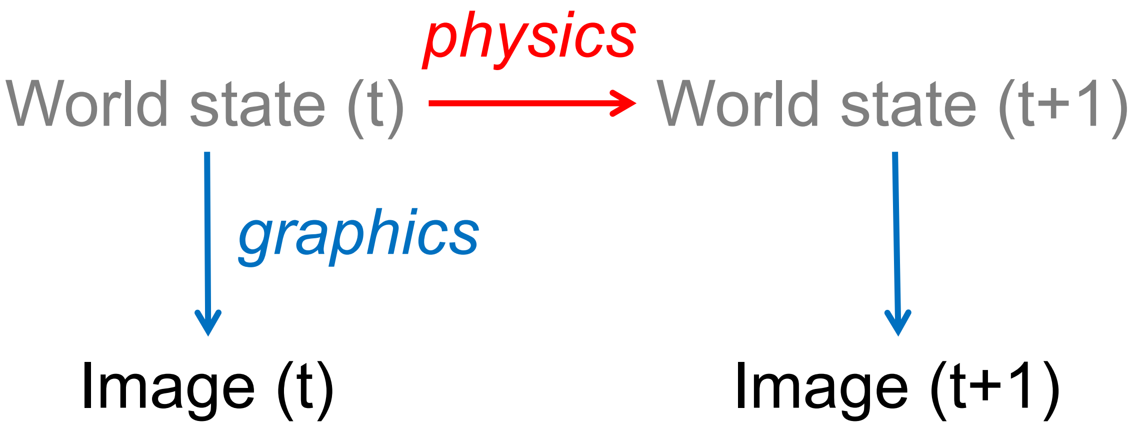
(Battaglia et al., *PNAS* 2013; Hamrick et al., *Cognition* 2016)



# The intuitive physics engine

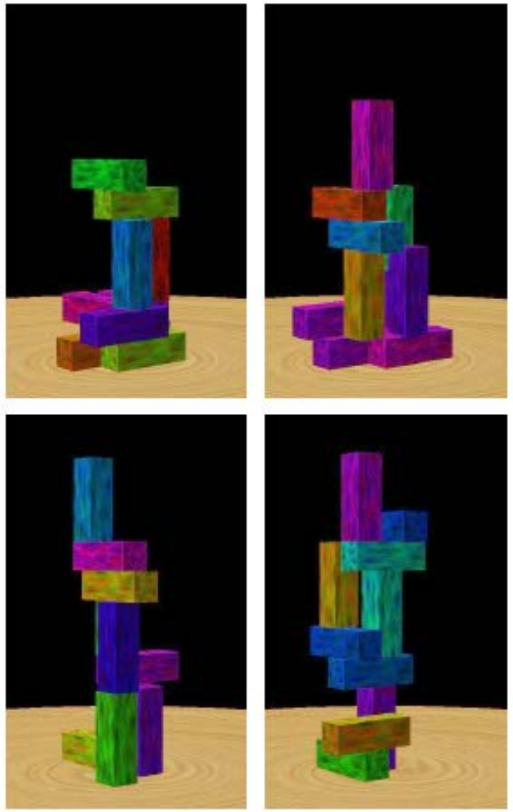
(Battaglia et al., *PNAS* 2013; Hamrick et al., *Cognition* 2016)

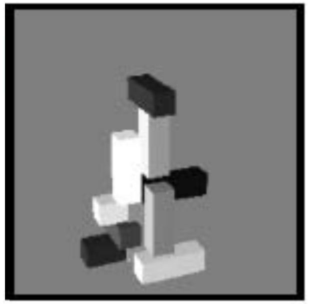




# Vision as inverse graphics

(Mansinghka, Kulkarni, Perov, Tenenbaum, NIPS 2013; Kulkarni et al., CVPR 2015)

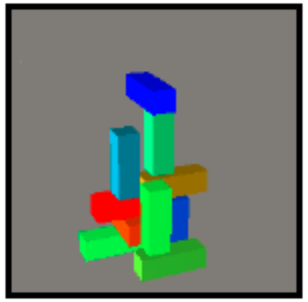




World state (t)

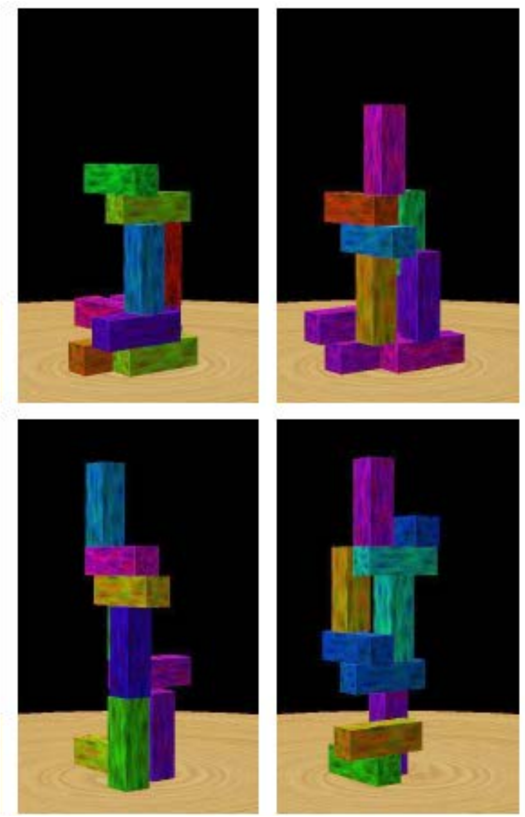
Prob. approx. rendering

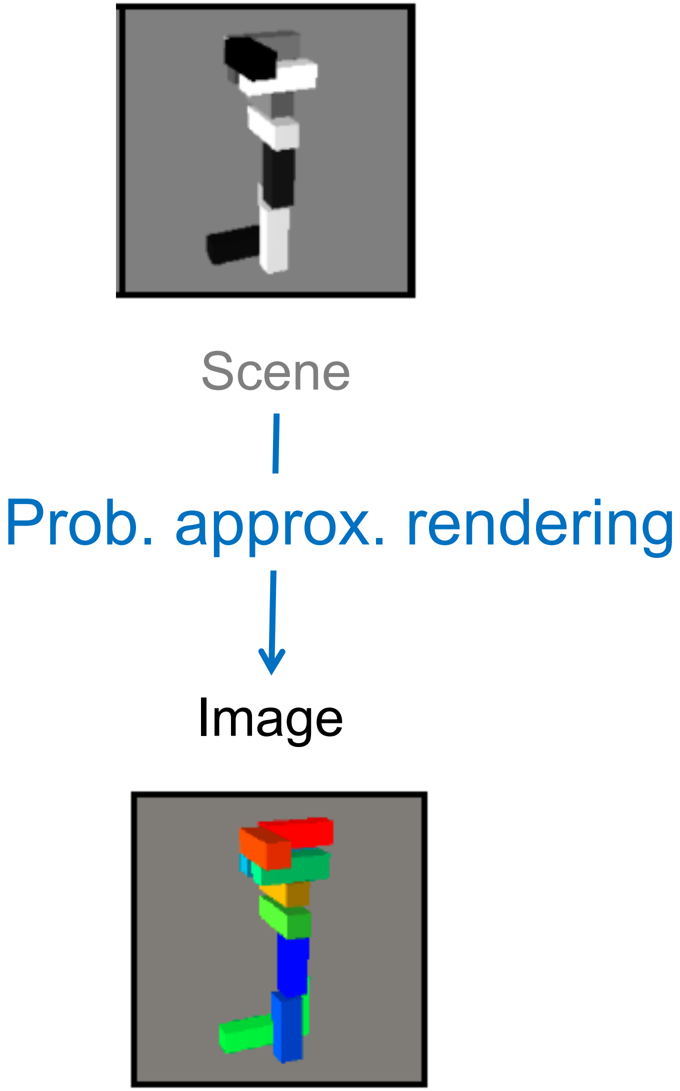
Image (t)



# Vision as inverse graphics

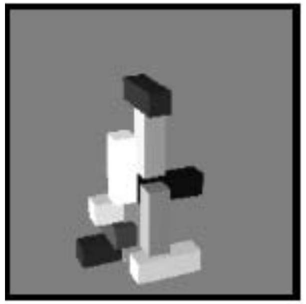
(Mansinghka, Kulkarni, Perov, Tenenbaum, NIPS 2013; Kulkarni et al., CVPR 2015)





# Vision as inverse graphics

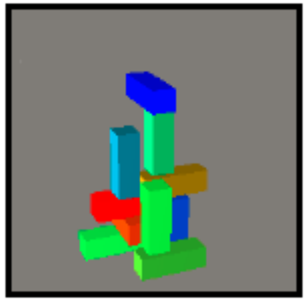
(Mansinghka, Kulkarni, Perov, Tenenbaum, NIPS 2013; Kulkarni et al., CVPR 2015)



Scene

Prob. approx. rendering

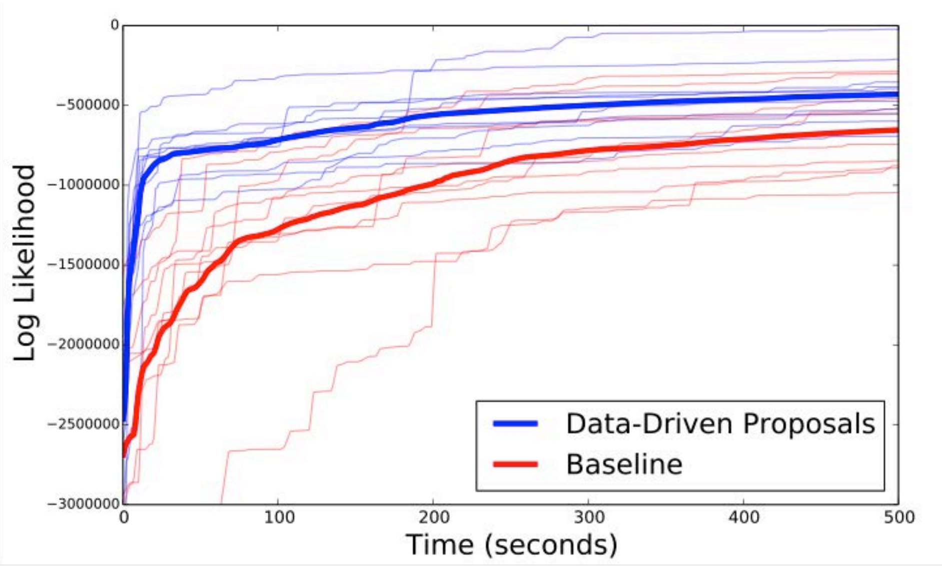
Image

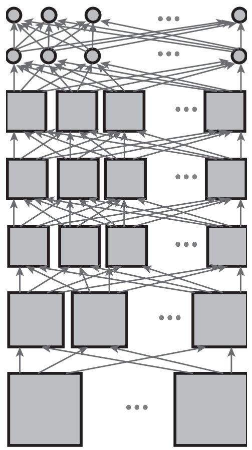


**Bayesian (probabilistic causal) inference, fast and slow**

*Slow*: Top-down sampling by reverse simulation. (e.g., markov chain monte carlo)

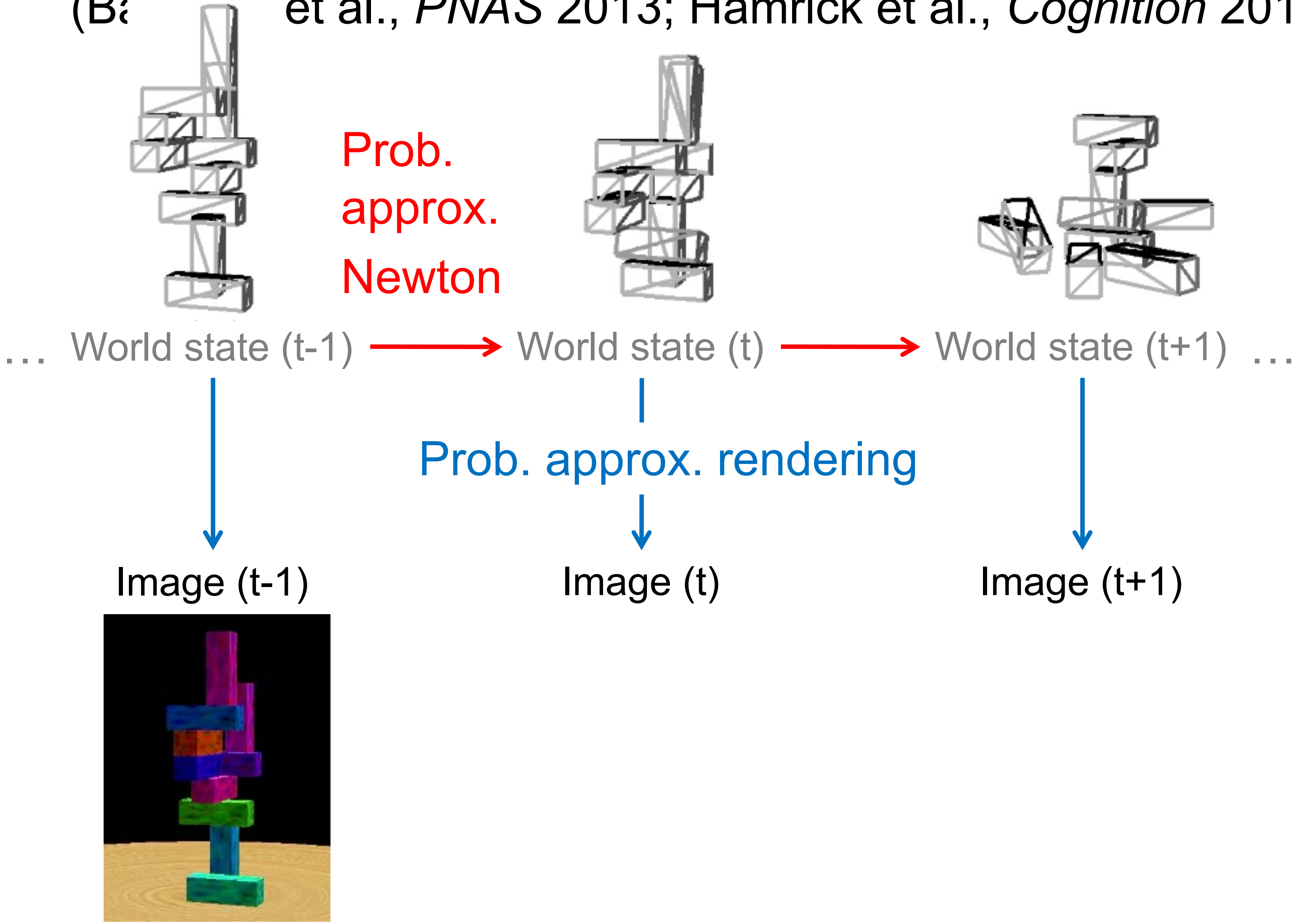
*Fast:* Bottom-up guesses about object shapes and locations, based on pattern recognition and function approximation (e.g., neural networks)



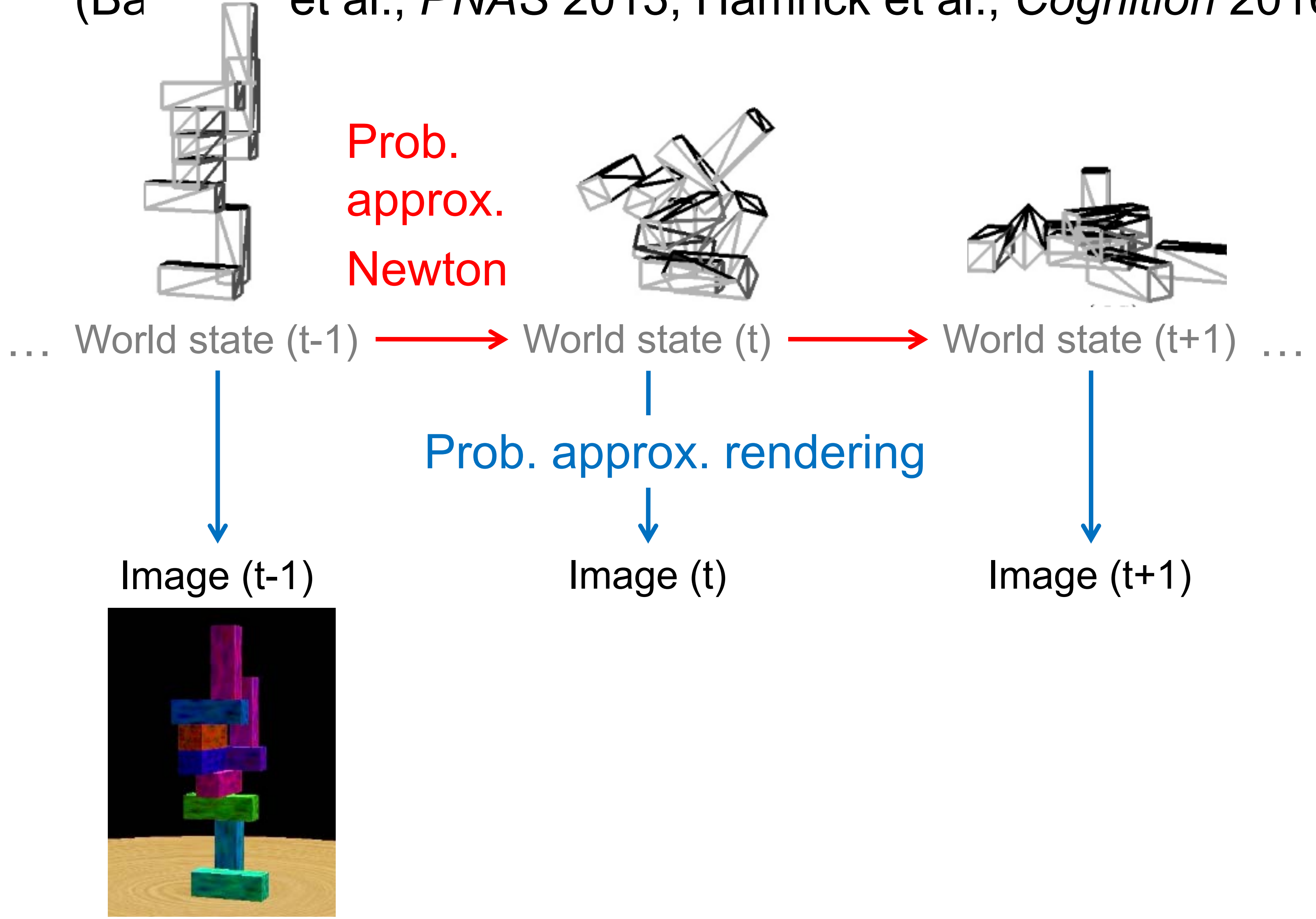
# The intuitive physics engine

(Ba[illegible] et al., *PNAS* 2013; Hamrick et al., *Cognition* 2016)



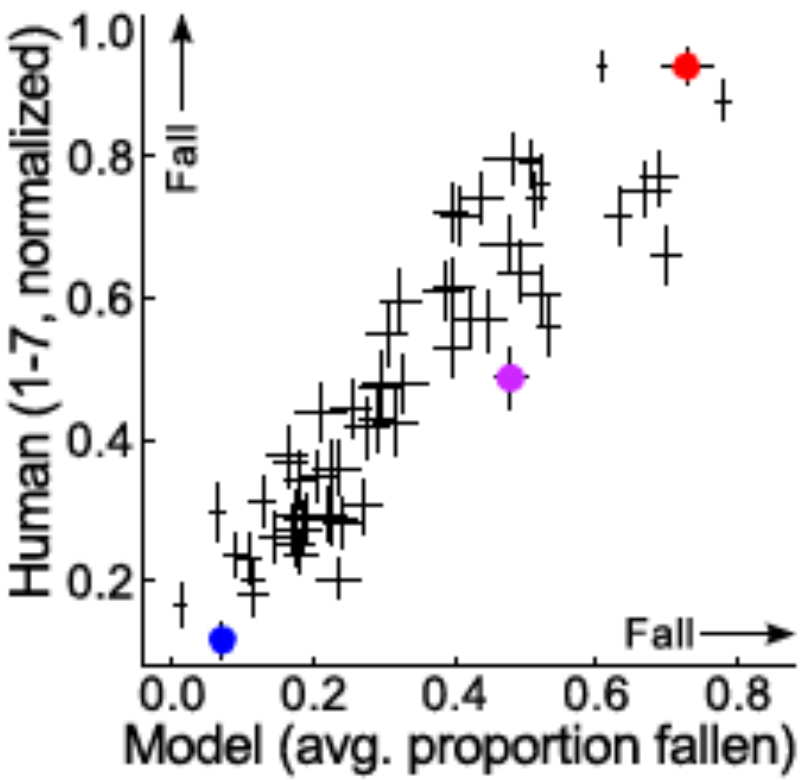
# The intuitive physics engine

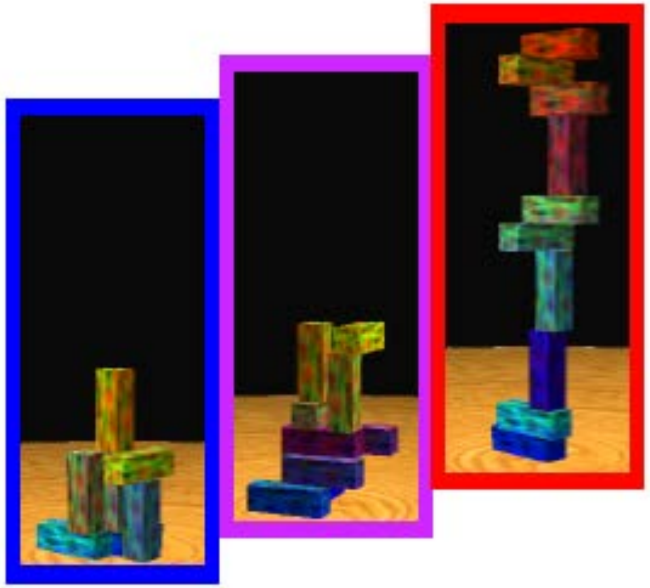
(Battaglia et al., *PNAS* 2013; Hamrick et al., *Cognition* 2016)

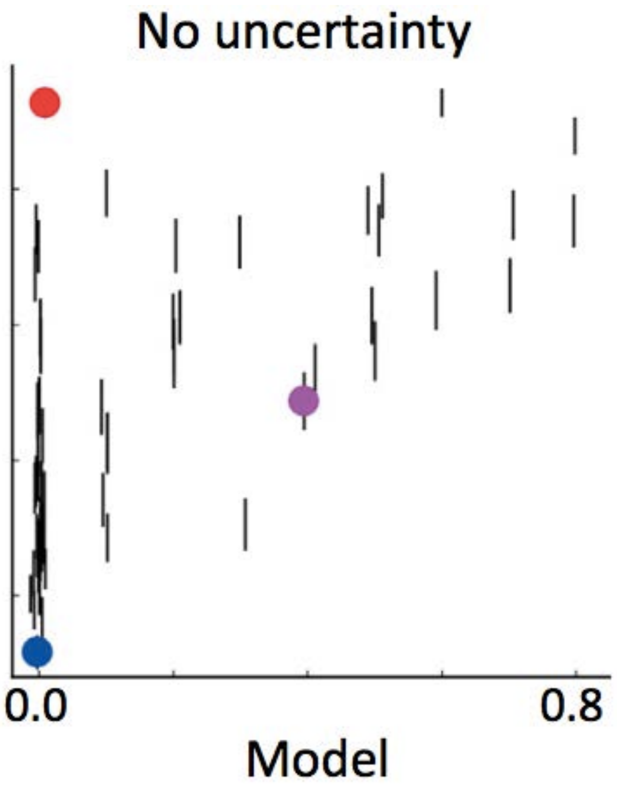


# The intuitive physics engine

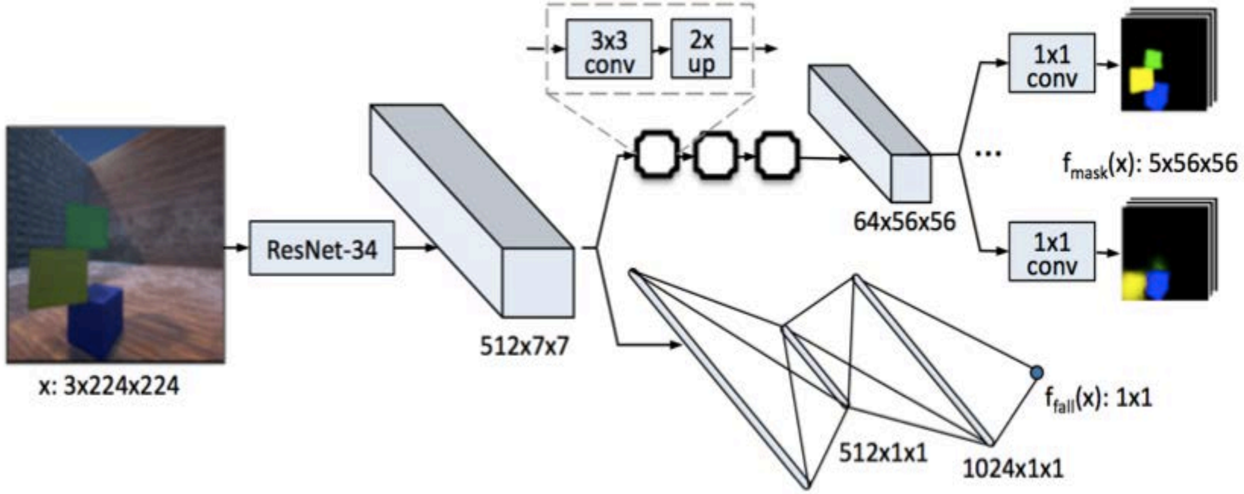
(Battaglia et al., *PNAS* 2013; Hamrick et al., *Cognition* 2016)



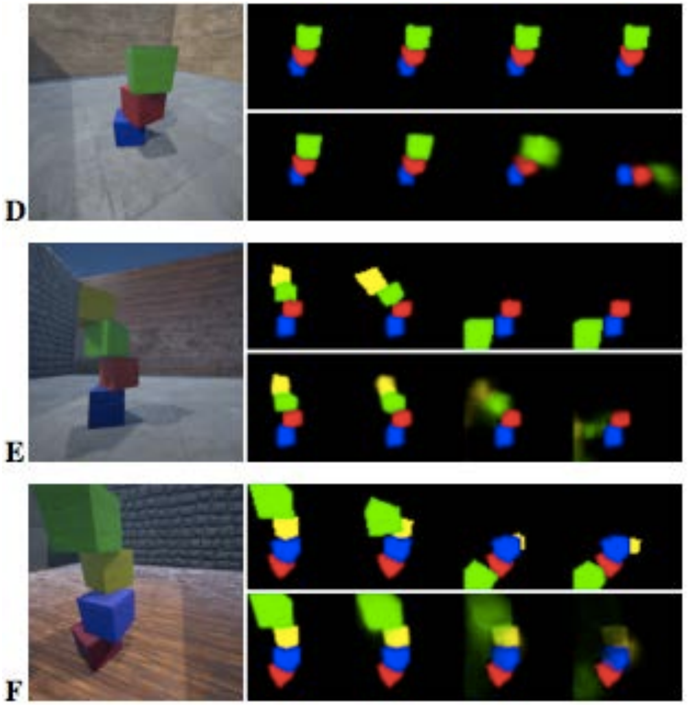


# An alternative to simulation: neural networks?




Can we treat intuitive physics as a pattern recognition task?

PhysNet (Facebook AI; Lerer et al 2016)




Requires much more training than people get (200K for 2-4 cubes), and doesn't generalize in all the ways that people do.

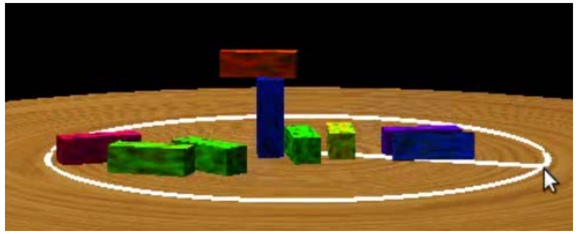
Without explicit representations of objects and their interactions, probably not compositional enough to capture underlying causal structure.

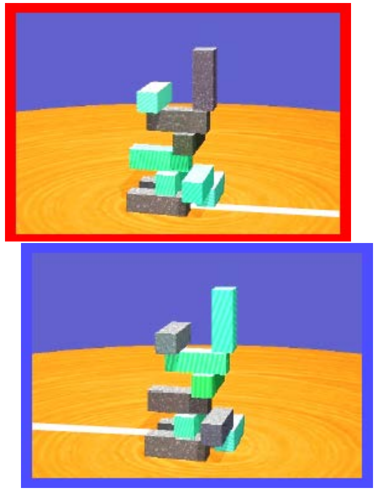
# The intuitive physics engine

(Battaglia et al., *PNAS* 2013; Hamrick et al., *Cognition* 2016)

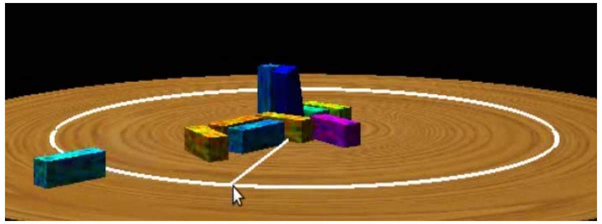
Will this stack of blocks fall?

Which way will they fall?

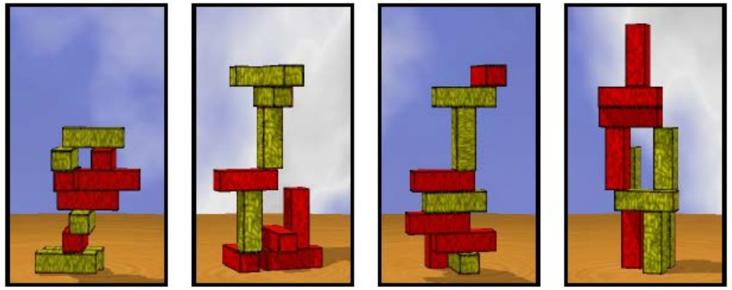




How far will they fall?



Is red or yellow heavier?



What if grey is much heavier than green?

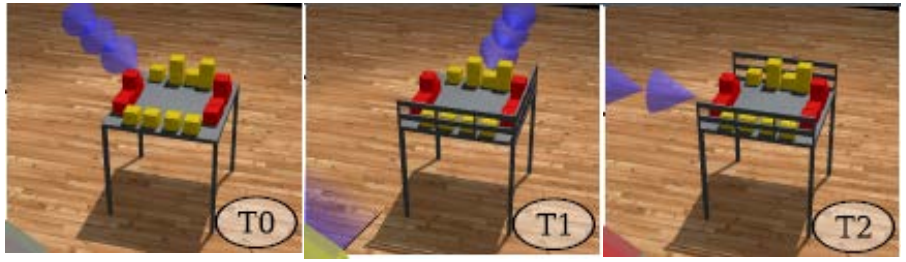
*physics*

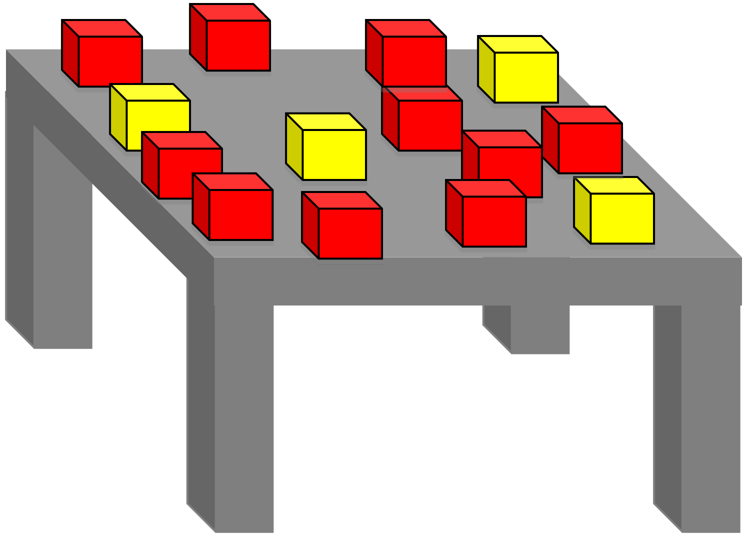
World state (t) → World state (t+1)

*graphics*

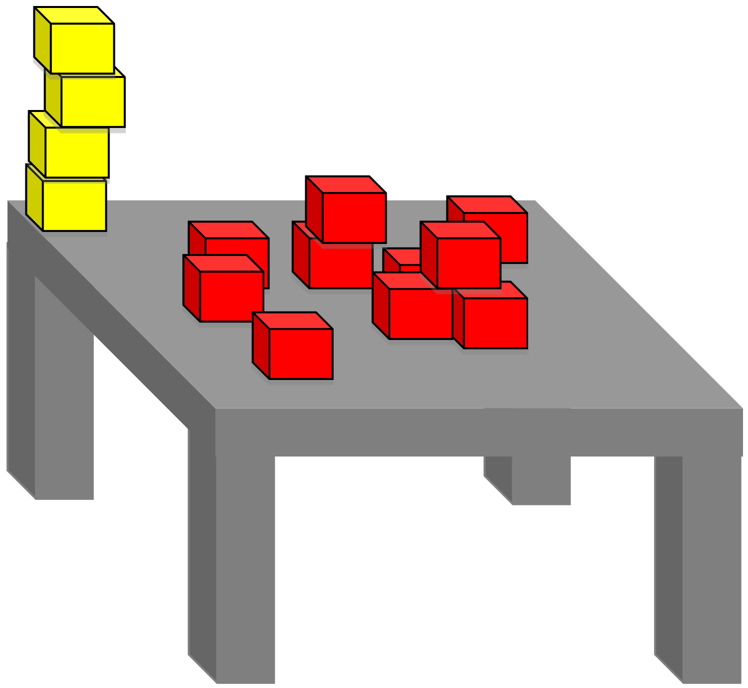
Image (t)    Image (t+1)

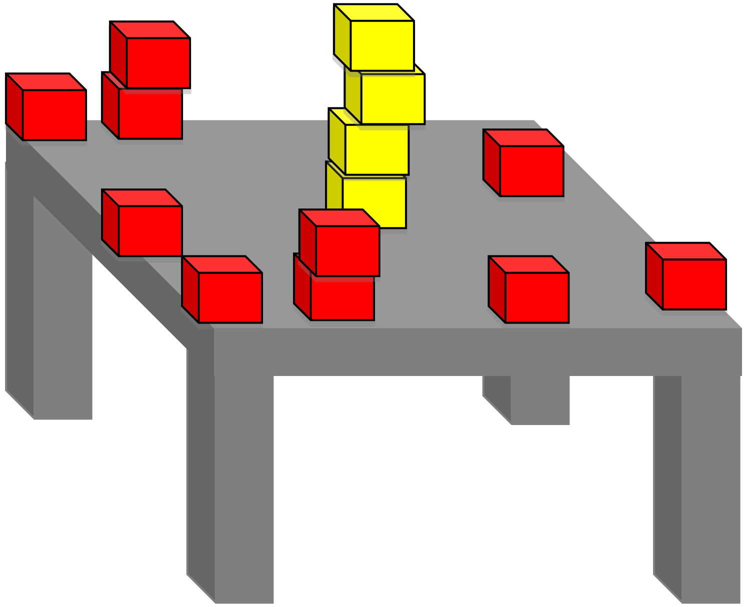
What will happen if you bump the table …?

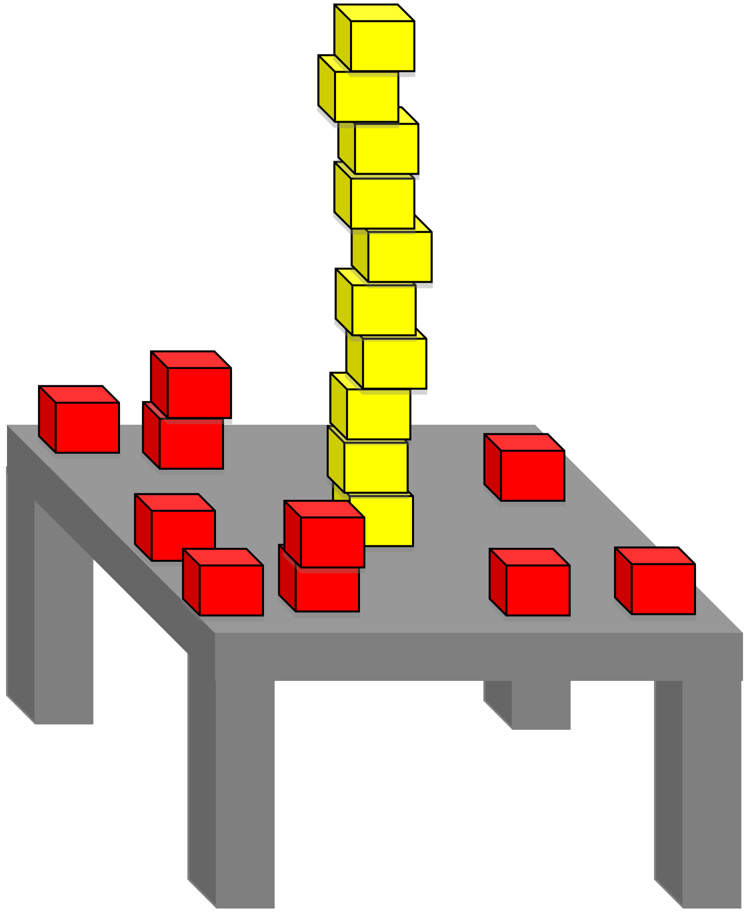


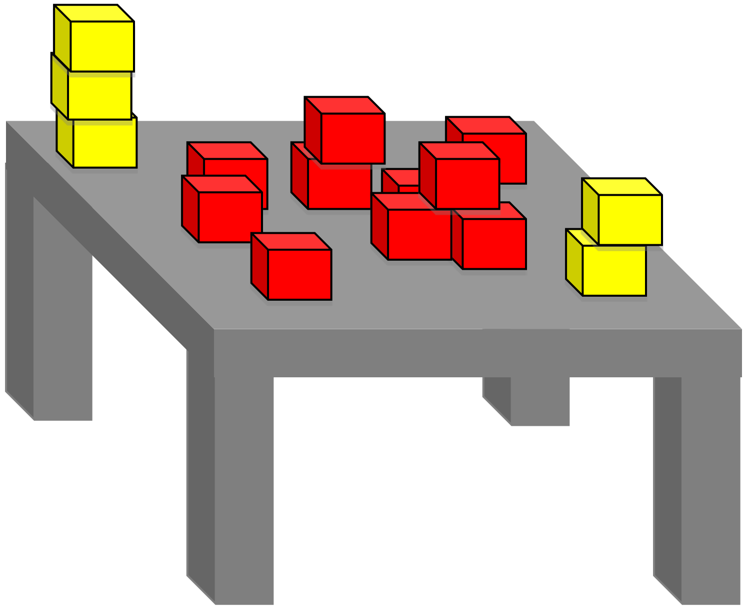


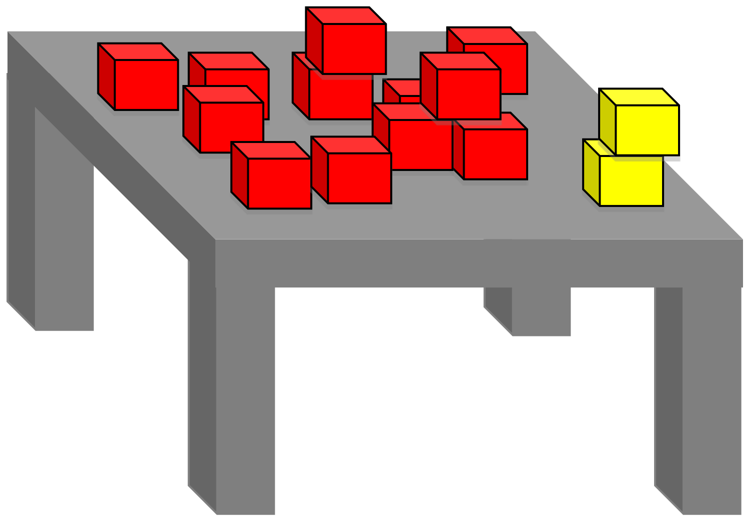
What if the table is bumped hard enough to knock some of the blocks onto the floor, is it more likely to be red blocks or yellow blocks?

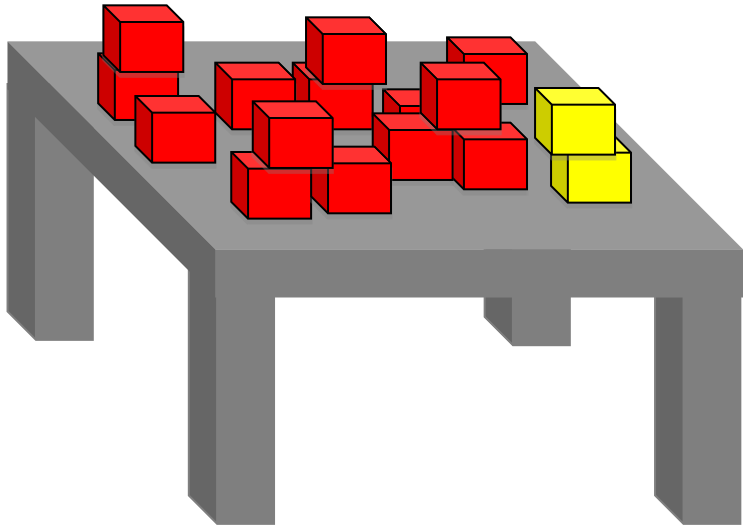


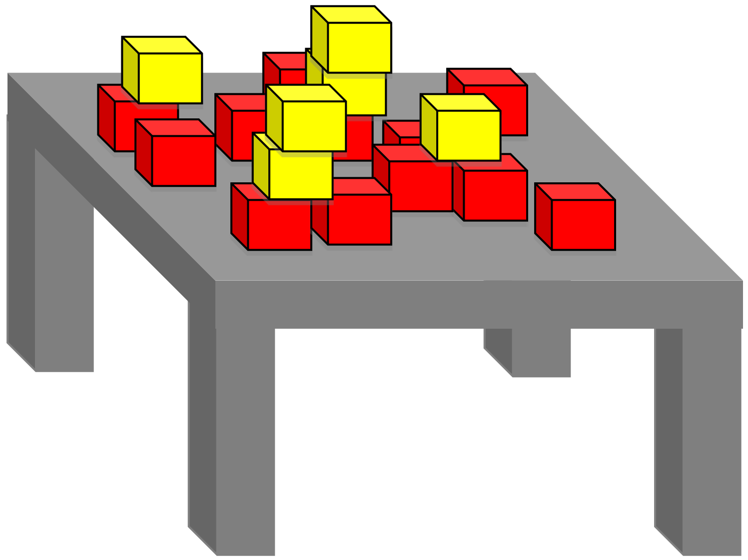




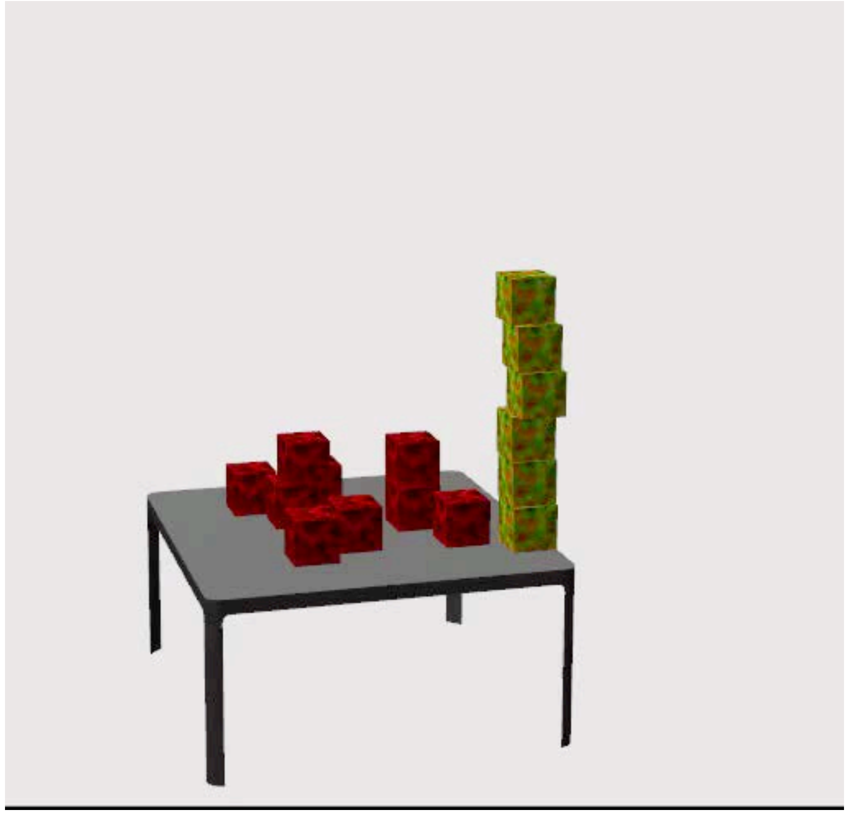






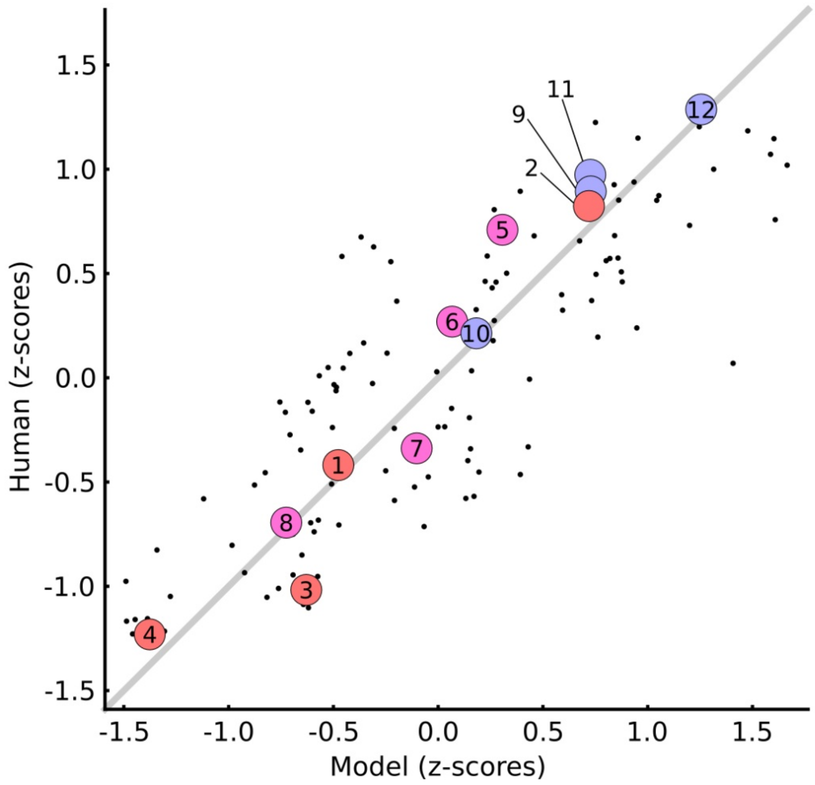


# Prediction by simulation



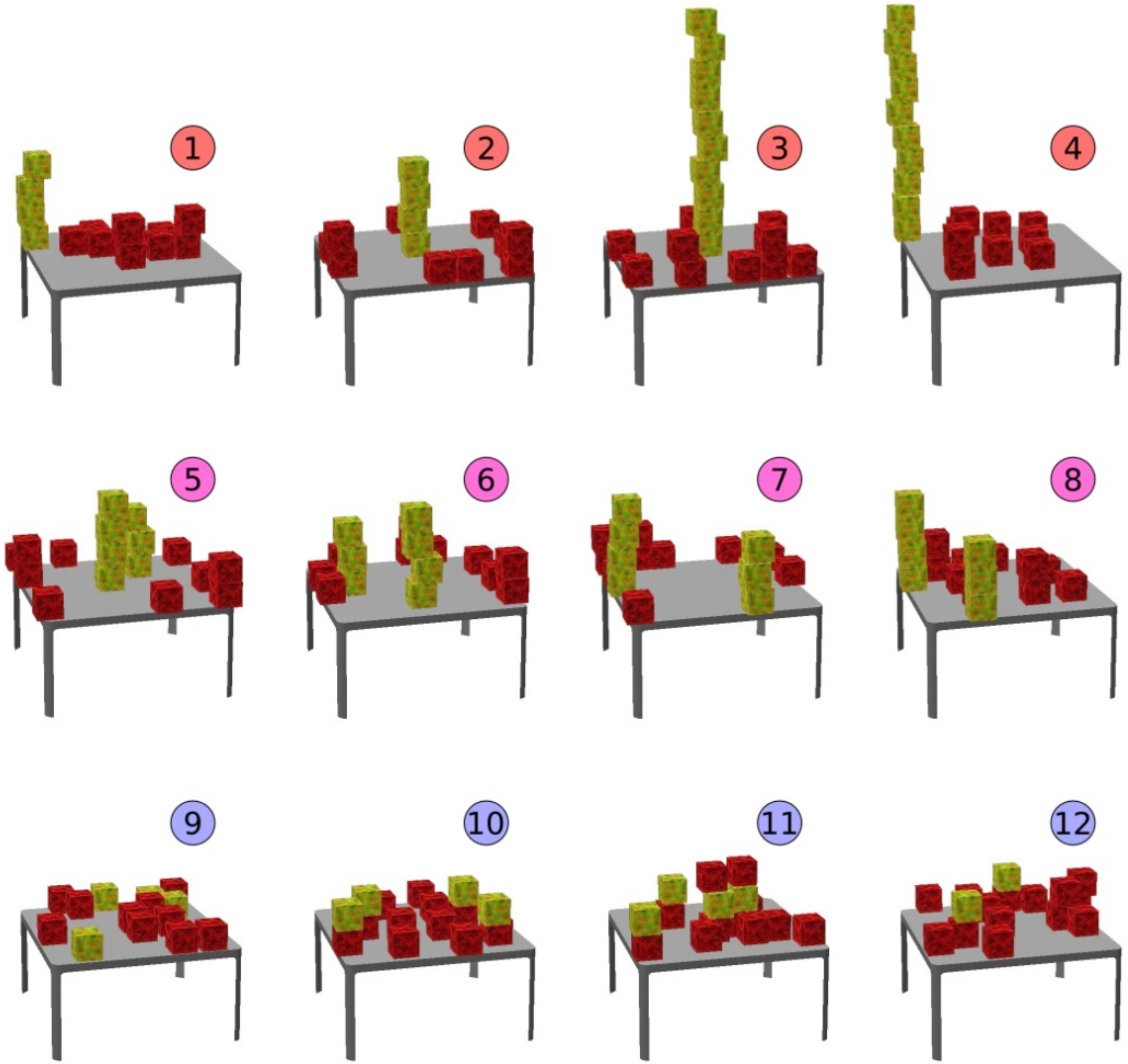
# What will happen if…?

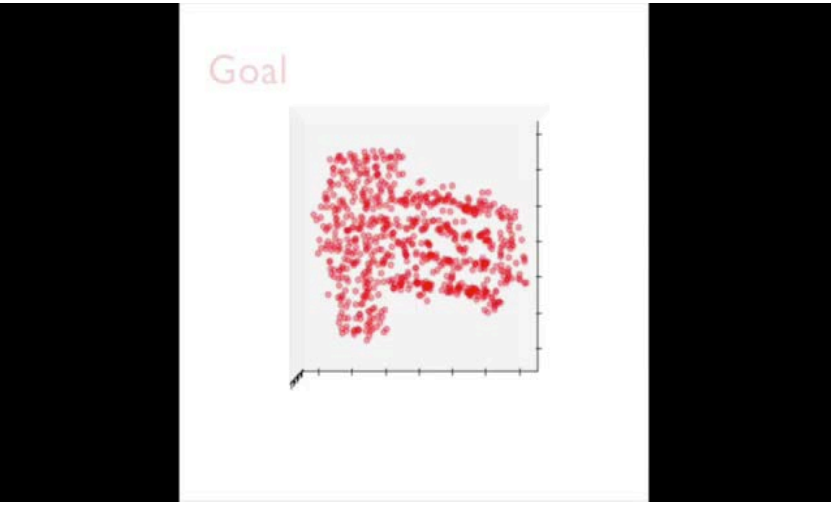
… you bump the table hard enough to knock some blocks onto the floor? Will you knock off more red, or yellow?

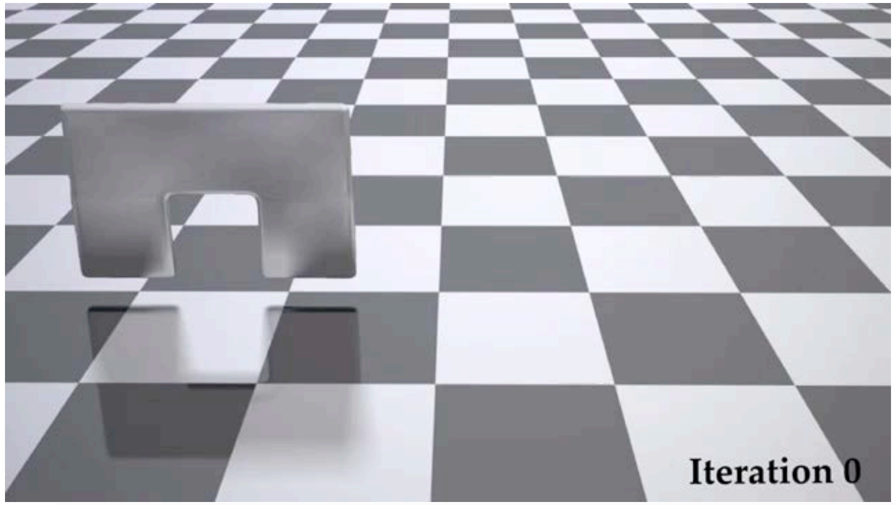


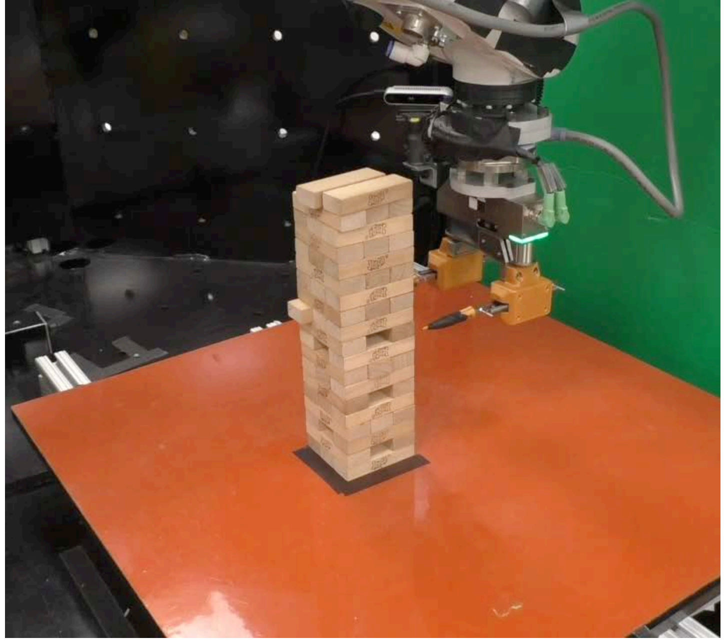

100% yellow

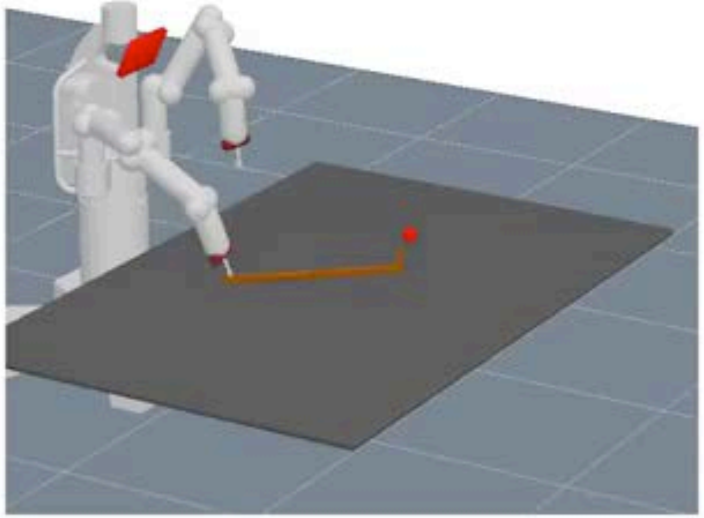
100% red

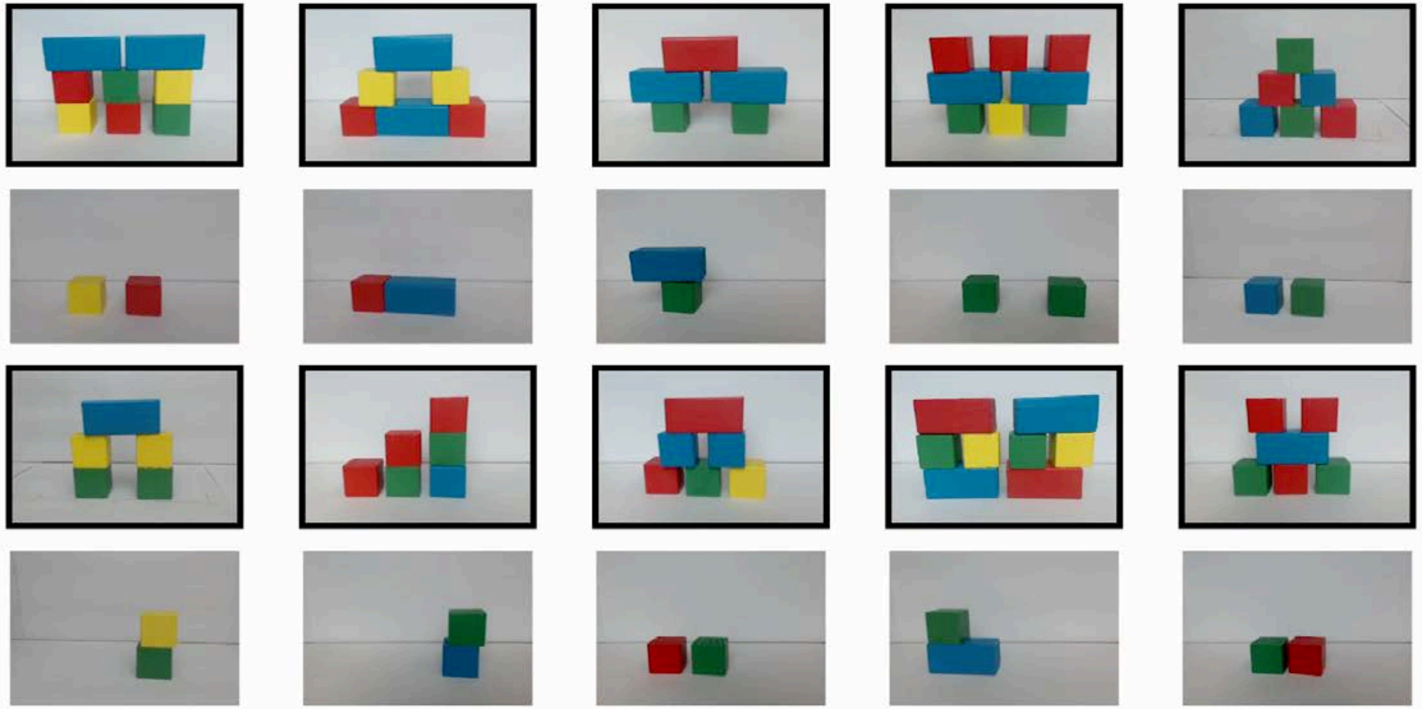



Goal

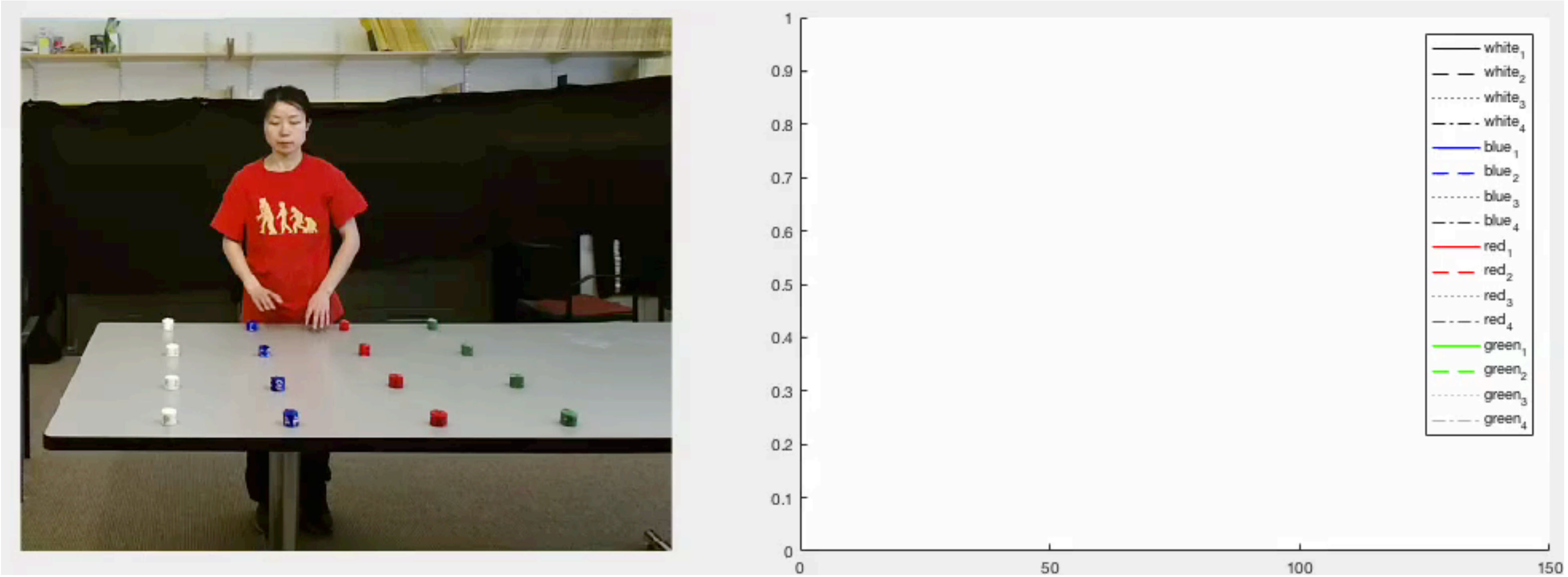

Iteration 0

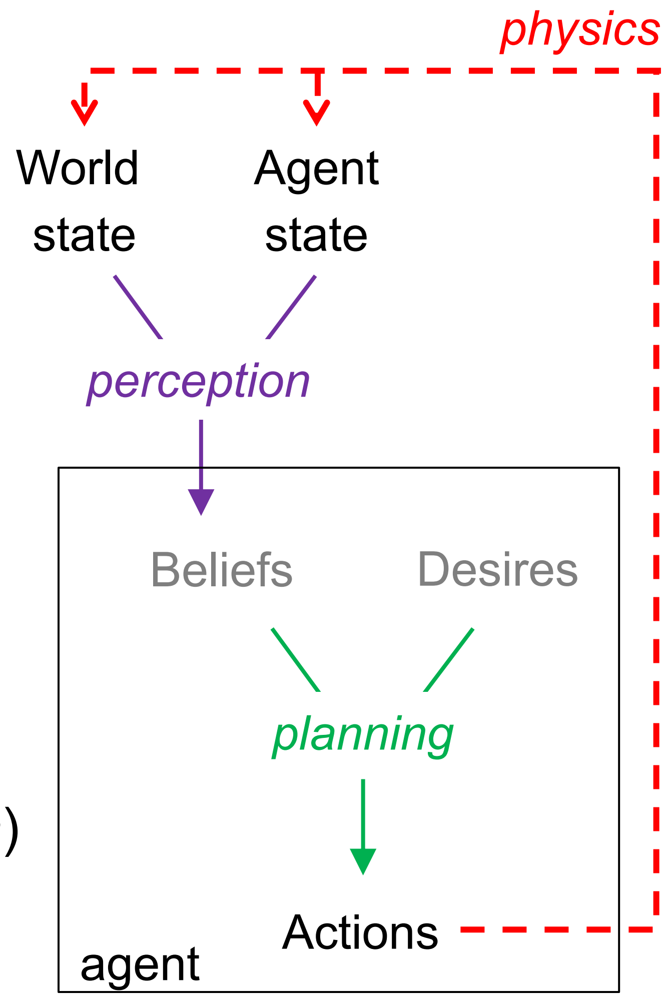


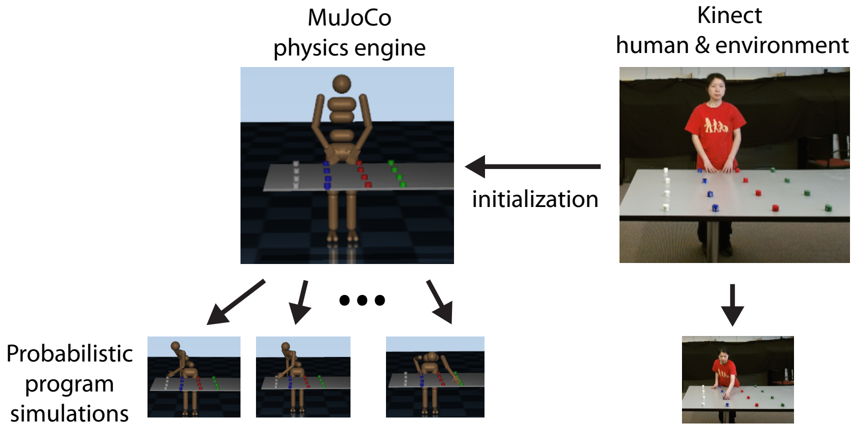




# Building intuitive psychology on physics for grounded action understanding




**Planning efficient actions:**

$$\underset{a}{\arg\max}\ U(a,s) = R(s) - C(a)$$

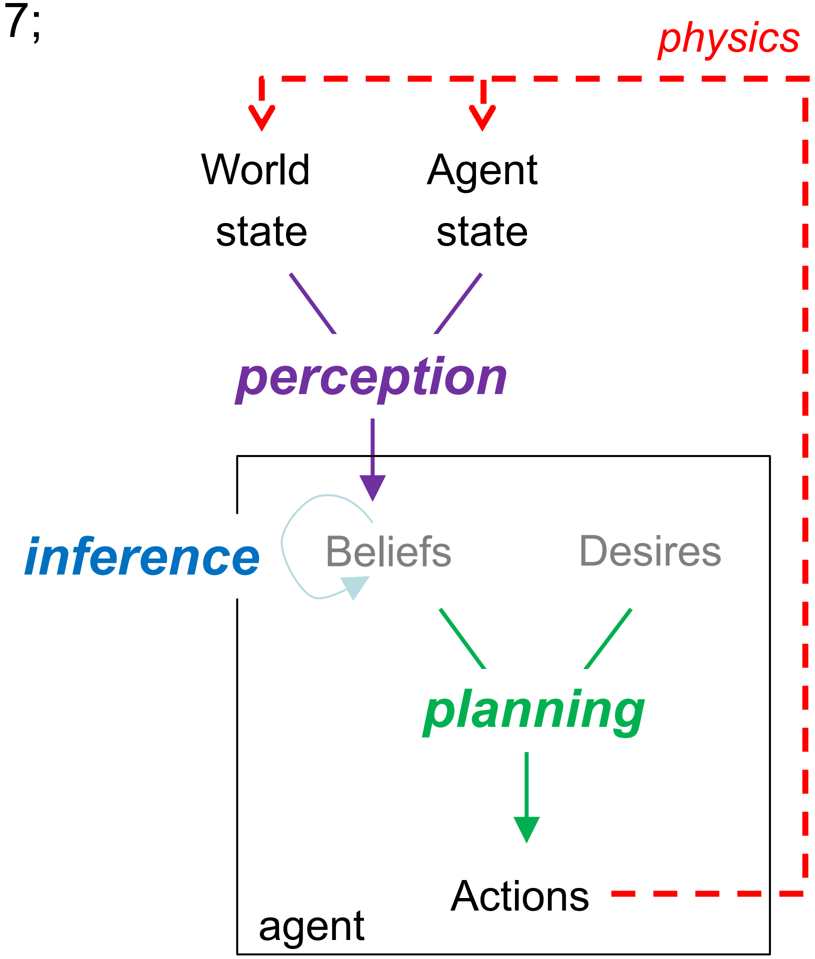
$R(s)$: reward for achieving goal state

$C(a)$: cost per action based on physical work done

# Joint inference of beliefs and desires

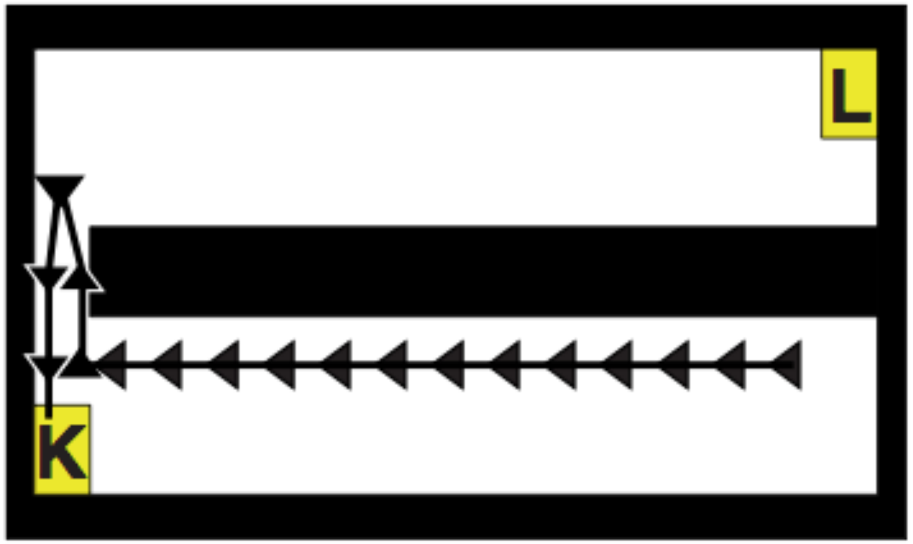
(Baker, Jara-Ettinger, Saxe, Tenenbaum, *Nature Human Behavior*, 2017;
Jara-Ettinger et al., *Trends in Cognitive Sciences*, 2016)

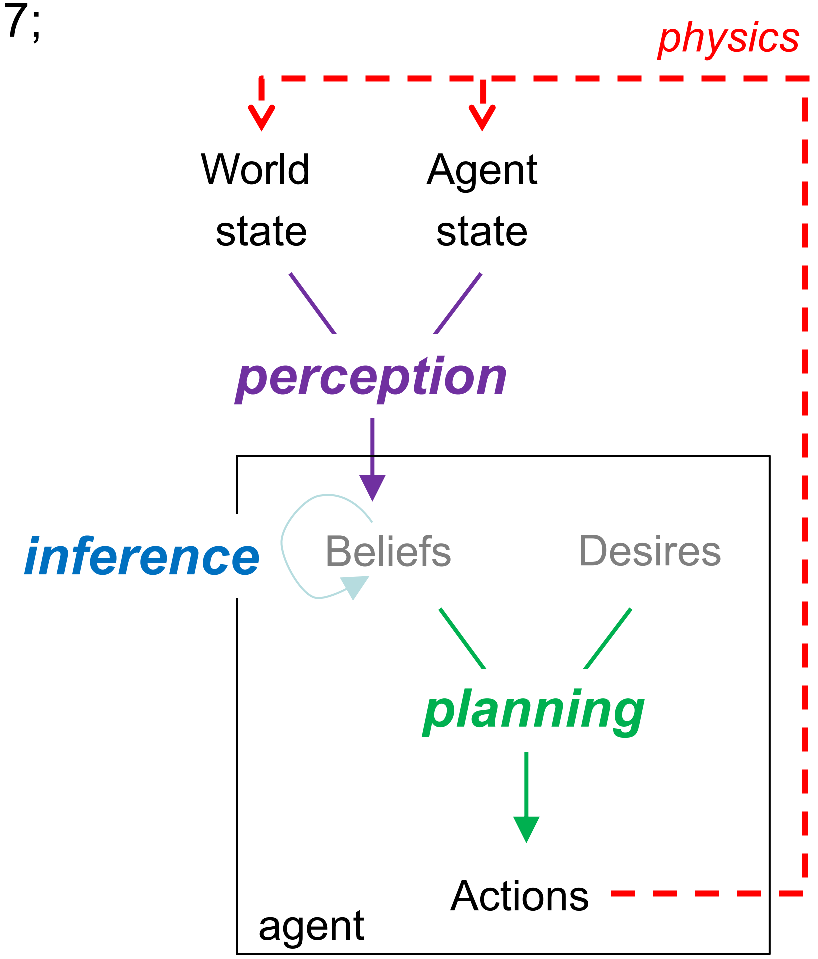

Three trucks come to campus on different days:
Korean (K), Lebanese (L) and Mexican (M)



# Joint inference of beliefs and desires

(Baker, Jara-Ettinger, Saxe, Tenenbaum, *Nature Human Behavior*, 2017;
Jara-Ettinger et al., *Trends in Cognitive Sciences*, 2016)

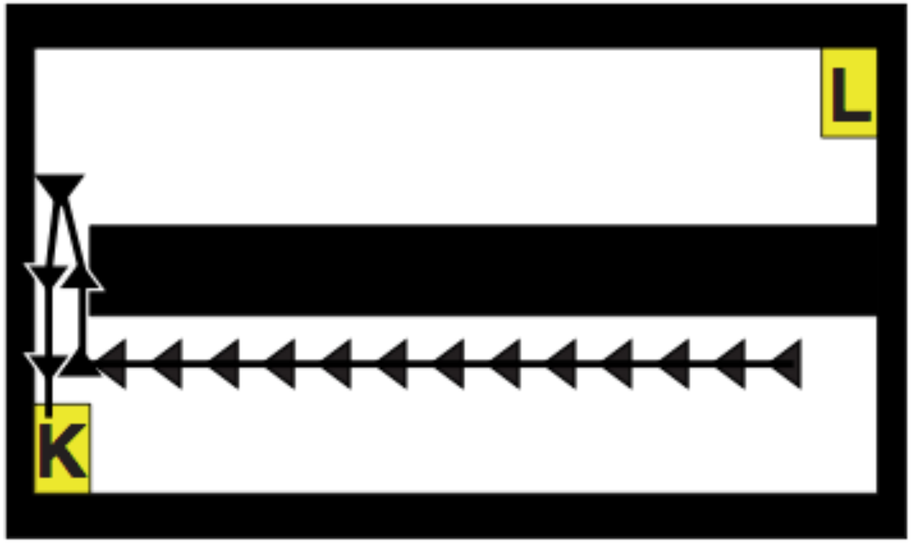


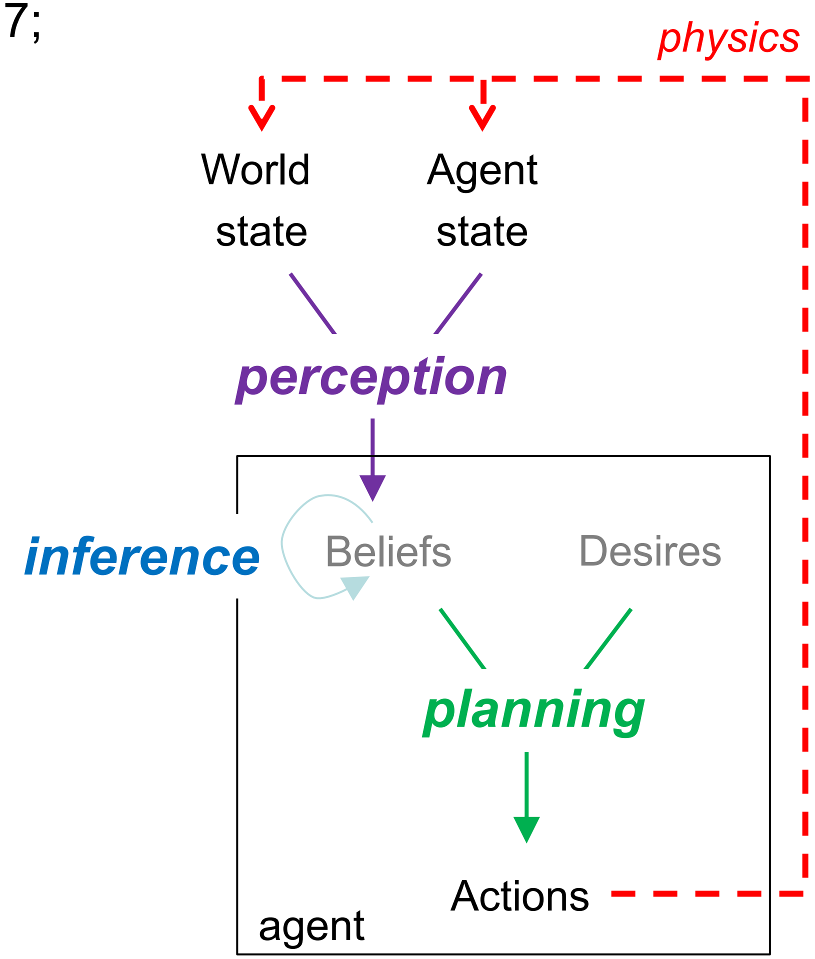




Three trucks come to campus on different days:
Korean (K), Lebanese (L) and Mexican (M)

# Joint inference of beliefs and desires

(Baker, Jara-Ettinger, Saxe, Tenenbaum, *Nature Human Behavior*, 2017;
Jara-Ettinger et al., *Trends in Cognitive Sciences*, 2016)



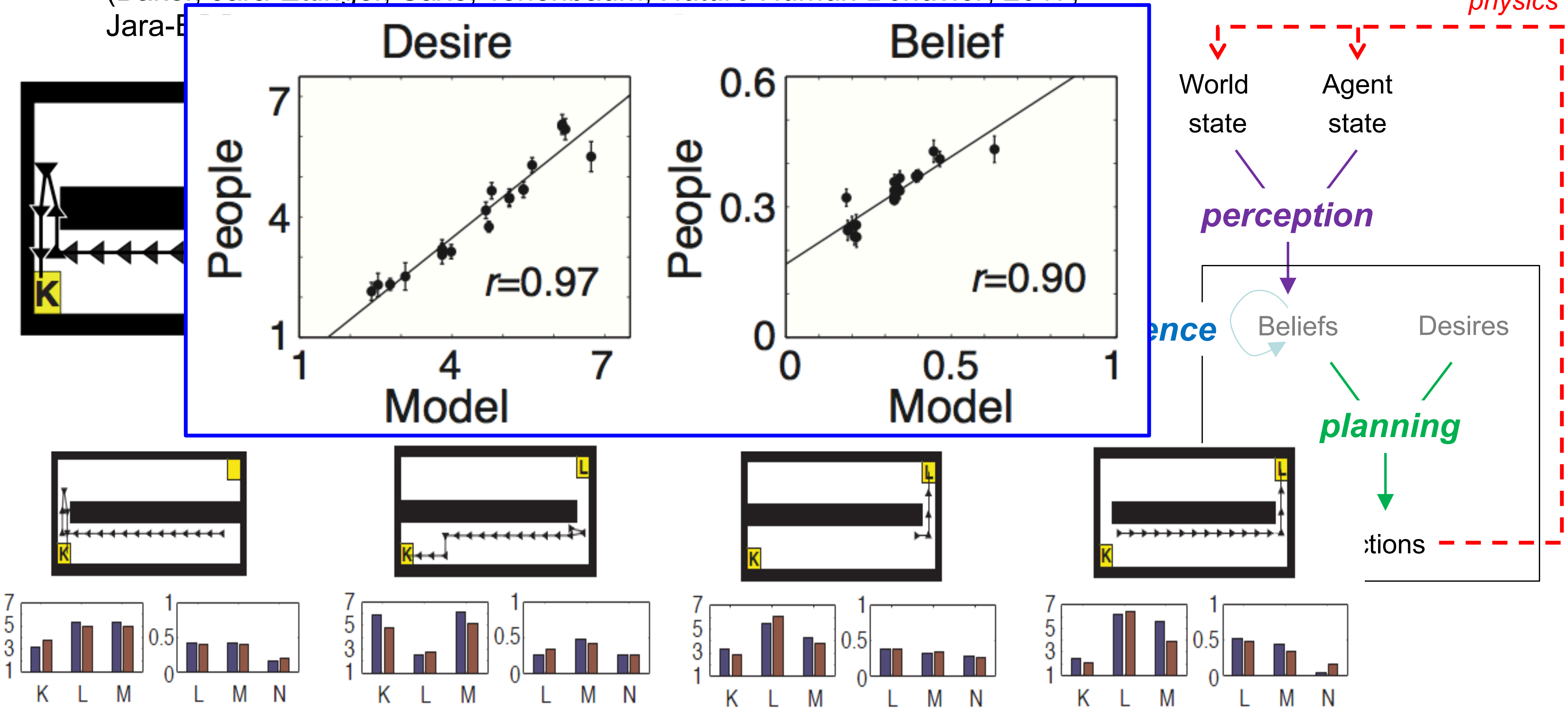




Three trucks come to campus on different days:
Korean (K), Lebanese (L) and Mexican (M)

What is Holly's favorite truck? And what did she believe was on the far side of the building when she first left her office?

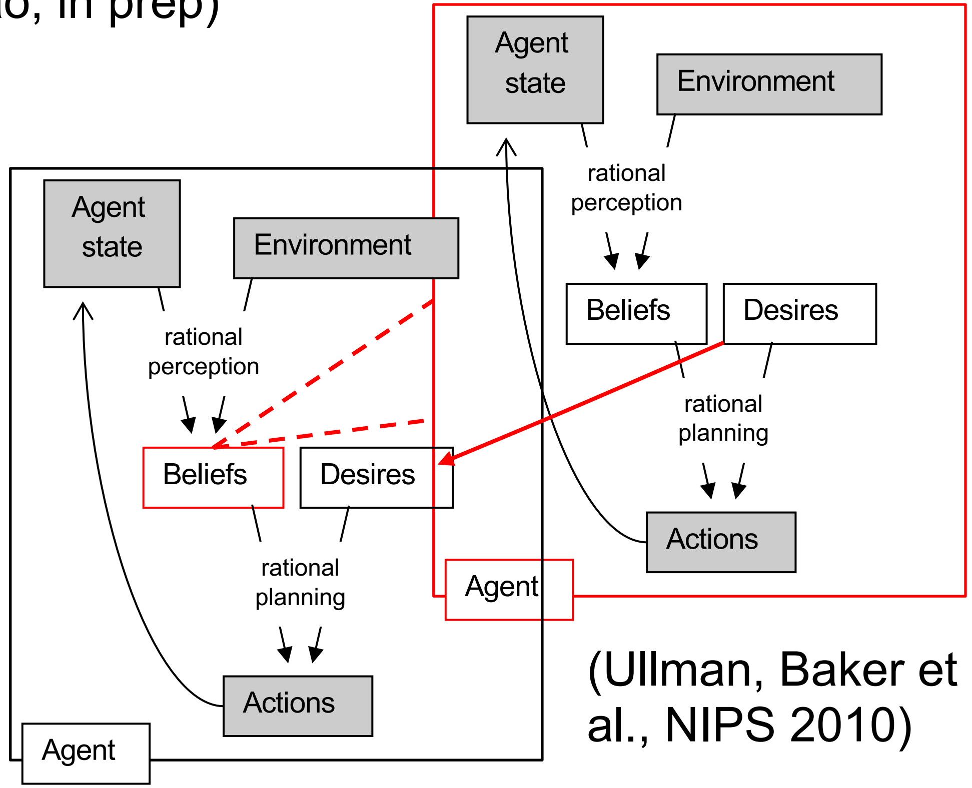
# Joint inference of beliefs and desires

(Baker, Jara-Ettinger, Saxe, Tenenbaum, *Nature Human Behavior*, 2017;
Jara-E



# Understanding social interactions

(Tao Gao, Chris Baker, Yibiao Zhao, in prep)




(Ullman, Baker et al., NIPS 2010)

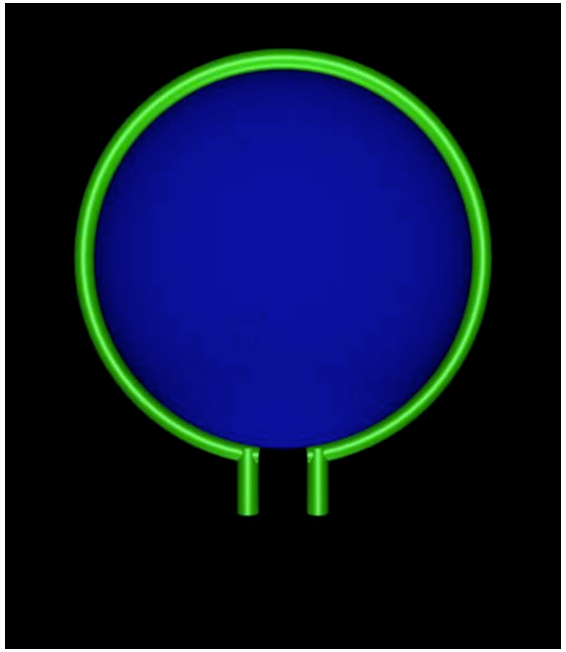
Recursive agent models, with one agent's utilities dependent on another's.

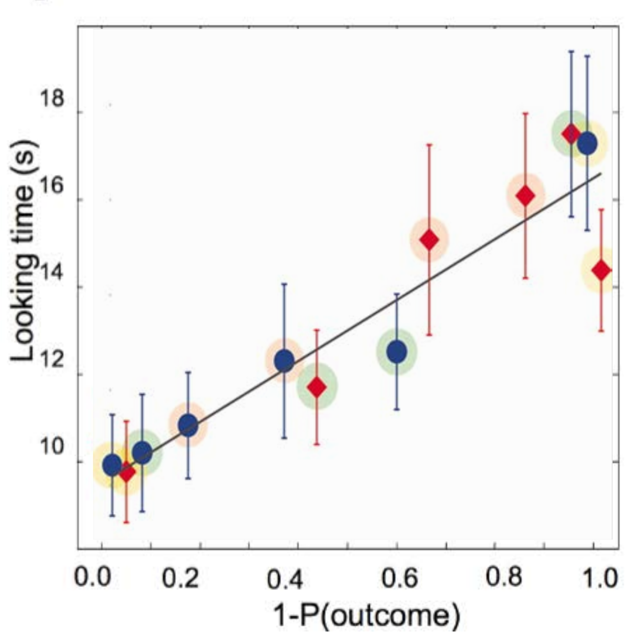
**Helping = *positive* utility dependence**
**Hindering = *negative* utility dependence**

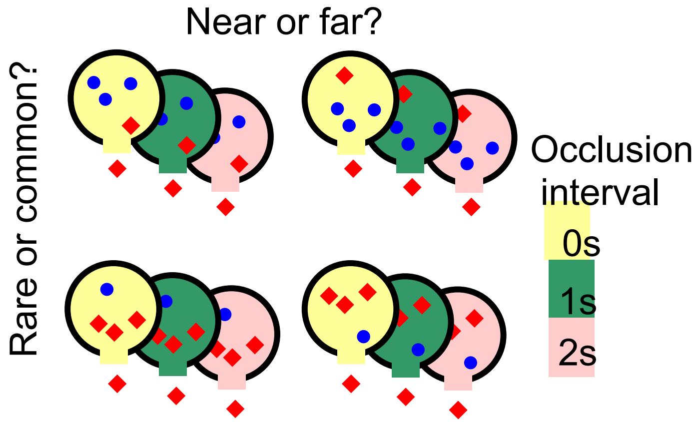
# The origins of common sense in babies

## Intuitive physics in 12 month olds

(Teglas, Vul, Girotto, Gonzalez, Tenenbaum, Bonatti, *Science* 2011)



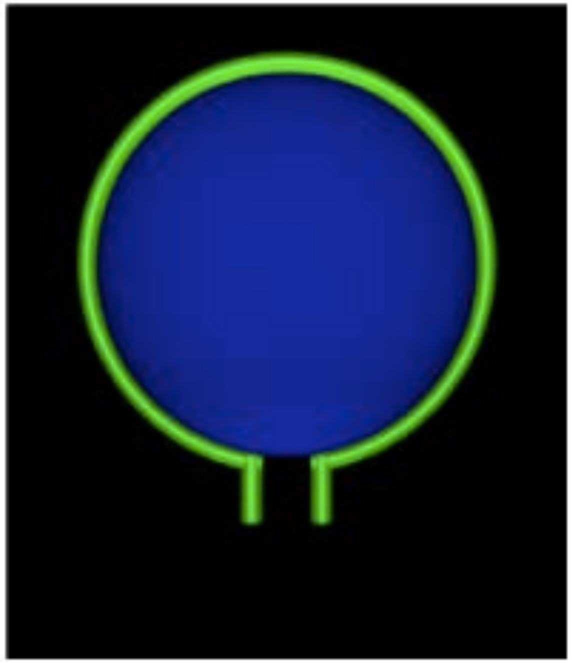


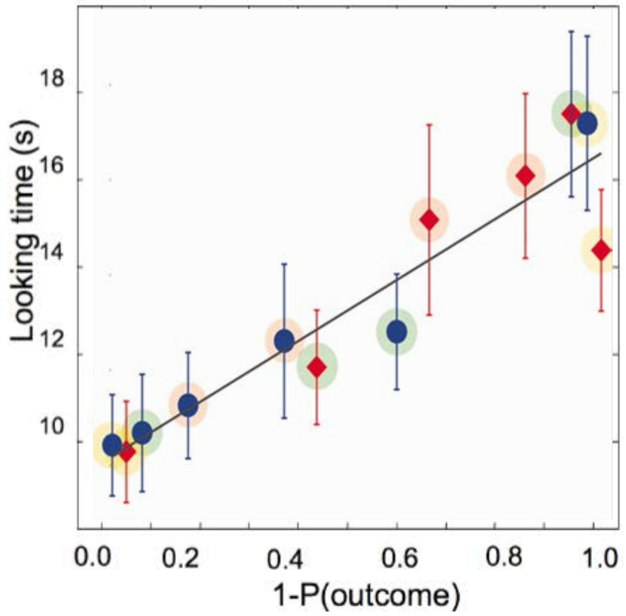



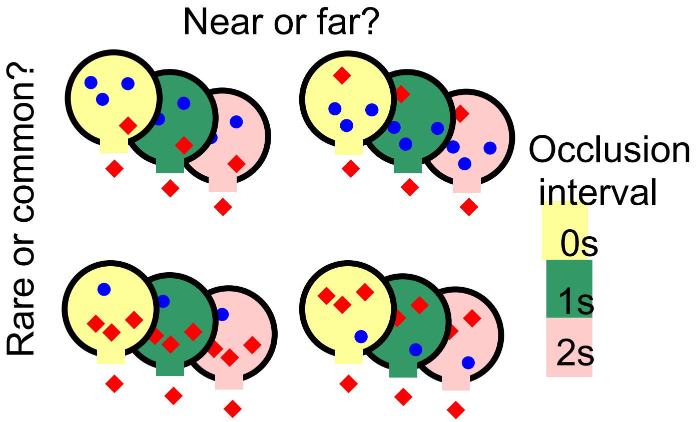
# The origins of common sense in babies

## Intuitive physics in 12 month olds

(Teglas, Vul, Girotto, Gonzalez, Tenenbaum, Bonatti, *Science* 2011)

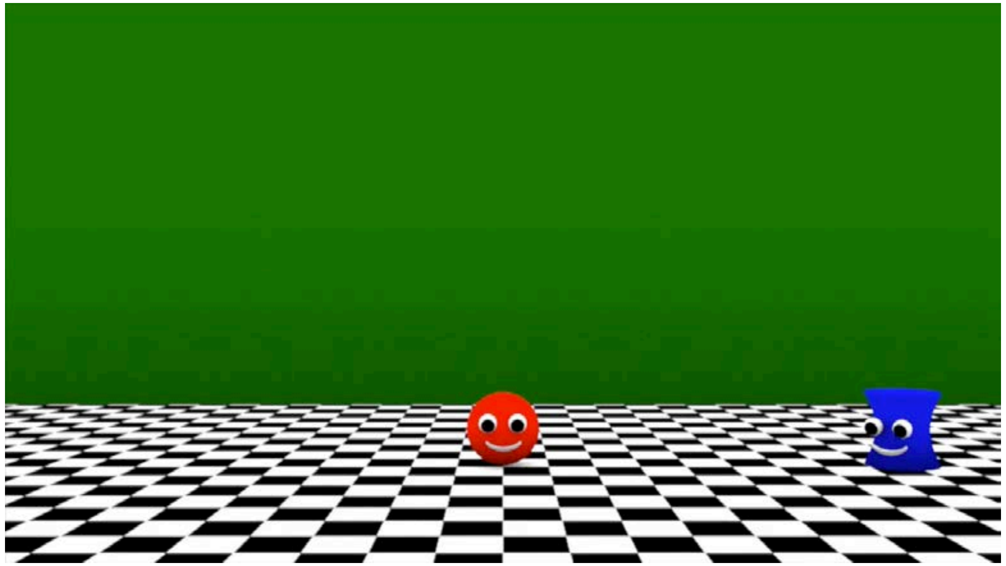


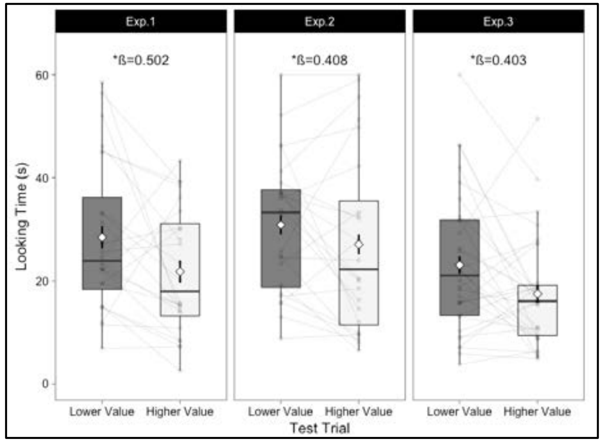


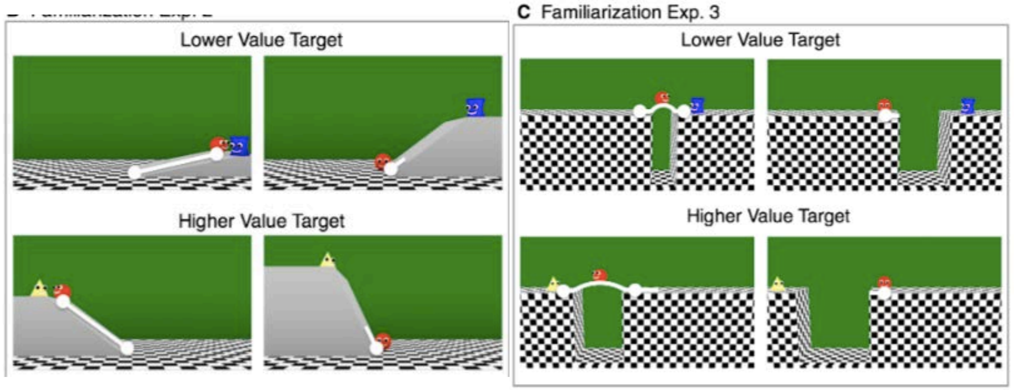




## Intuitive psychology in 10 month olds

(Liu, Ullman, Tenenbaum, Spelke, *Science* 2017)





# Where does learning come into the picture?

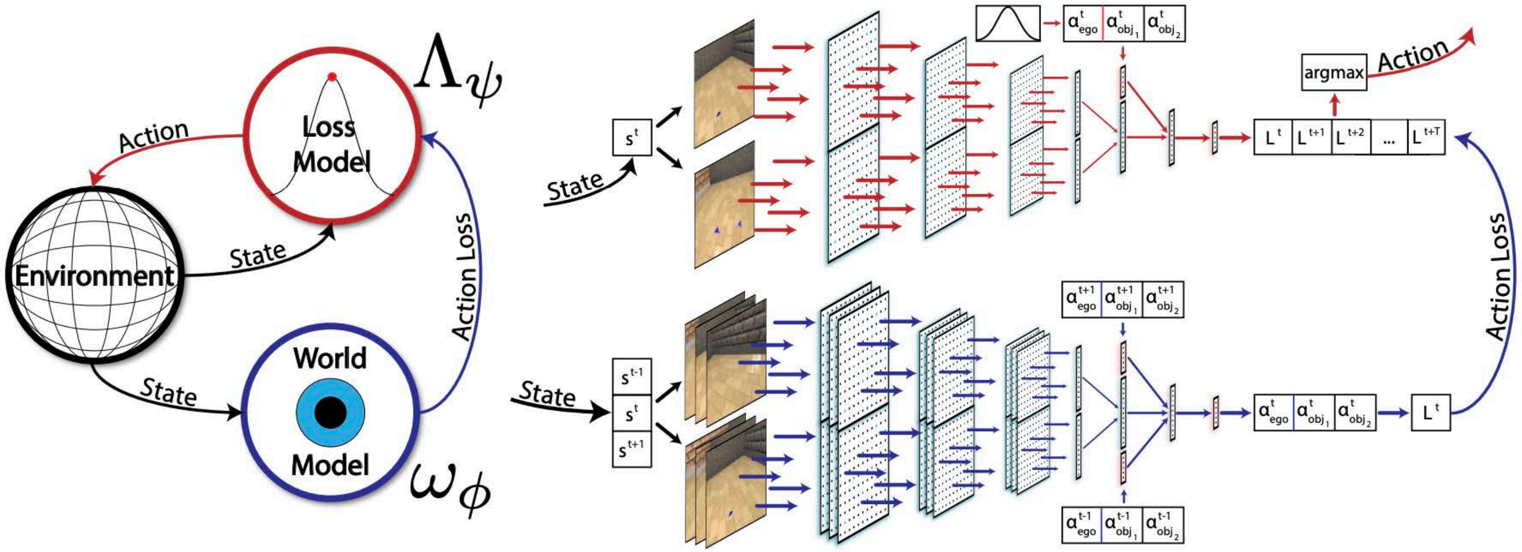
# Where does learning come into the picture?

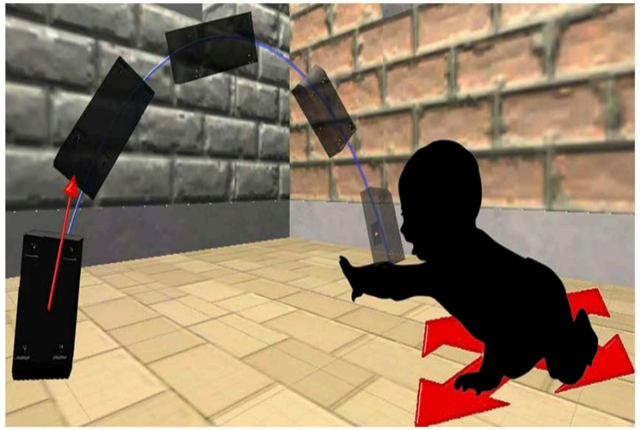
- One possibility: These systems emerge mostly from scratch, in each child's mind, by learning end-to-end from raw pixels what is needed to support prediction and interaction with the world.

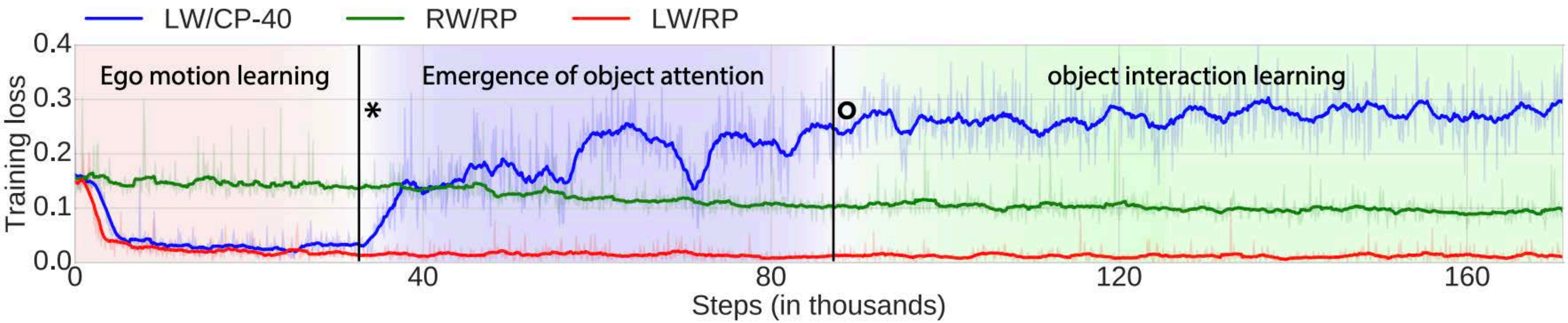
# Where does learning come into the picture?

**Emergence of Structured Behaviors from Curiosity-Based Intrinsic Motivation**

Nick Haber, Damian Mrowca, Li Fei-Fei, Daniel L. K. Yamins
( [nhaber, mrowca, feifeili, yamins]@stanford.edu )



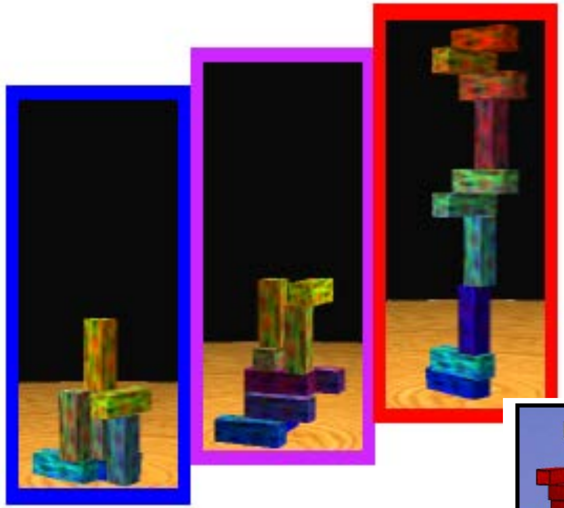



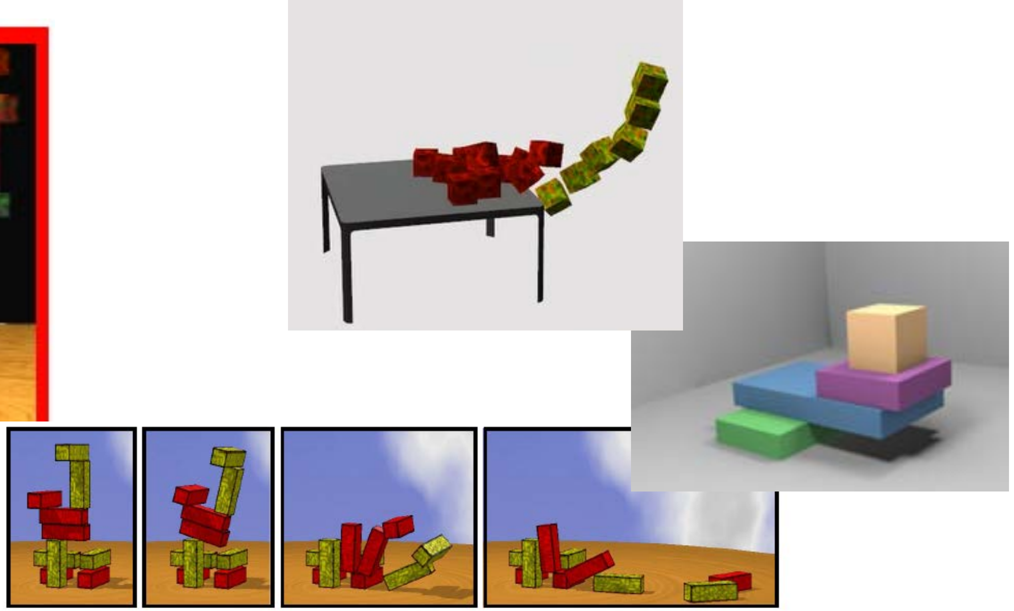


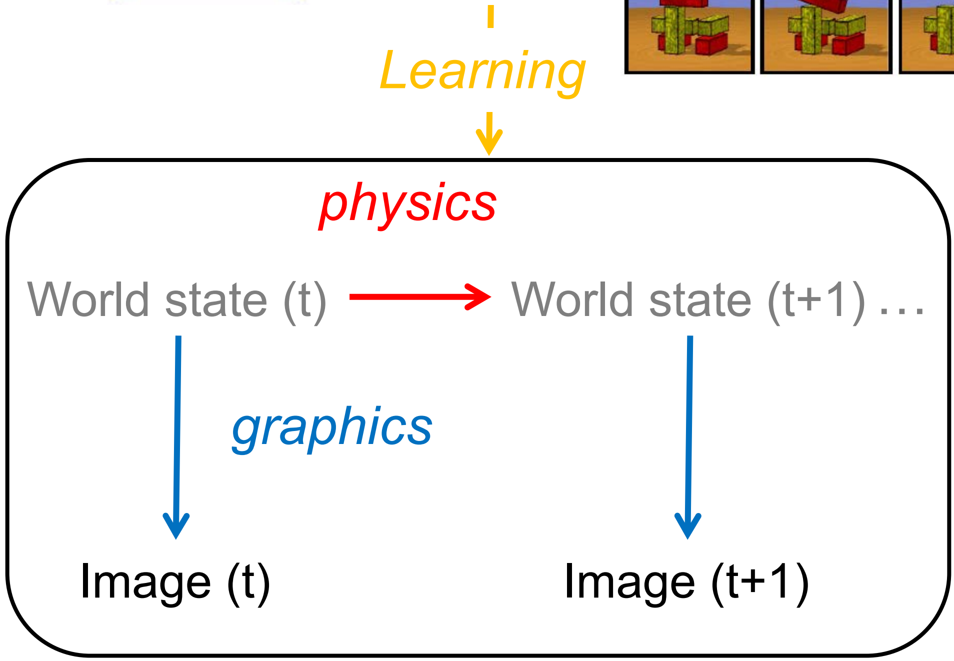
# Where does learning come into the picture?

- Learning in the game engine
- Learning the game engine itself
- Using these foundations to learn everything else

# What kind of learning algorithm can build a physics engine? "Program-learning programs"

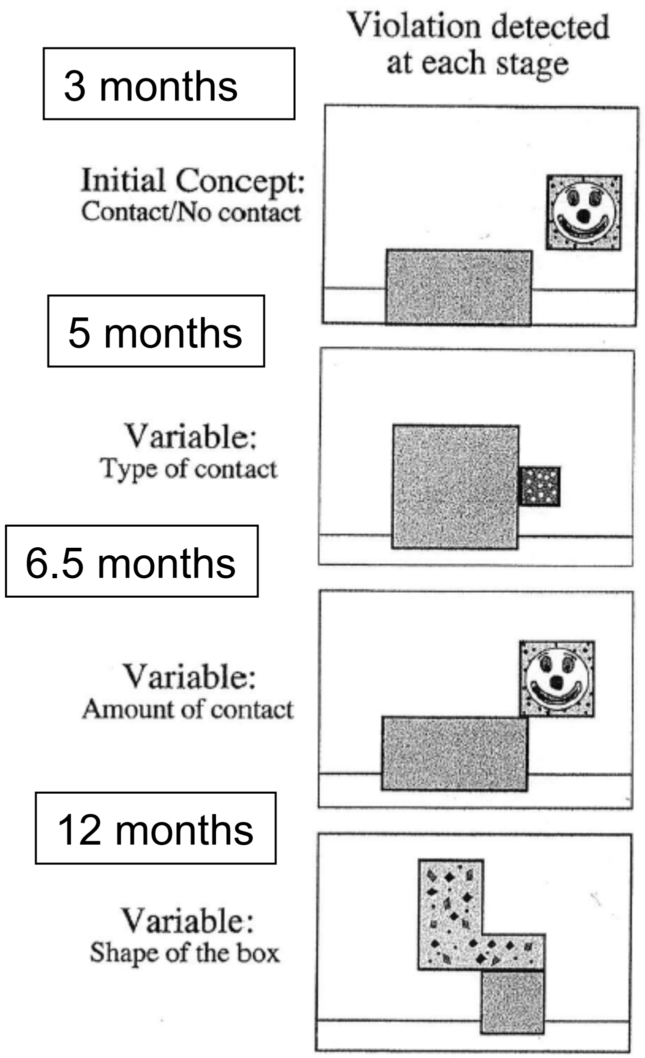







# The development of intuitive physics in humans

- 0-4 months:
  - Object permanence, spatiotemporal continuity, solidity, rigidity
- 6-7 months:
  - Stability, support, causality.
- 8-10 months:
  - Gravity, inertia, transfer of momentum, physics integrated with object shape perception.
- 10-12 months:
  - Center of mass, weight, shape constancy, object tracking integrated with intuitive psychology for joint attention and intention.



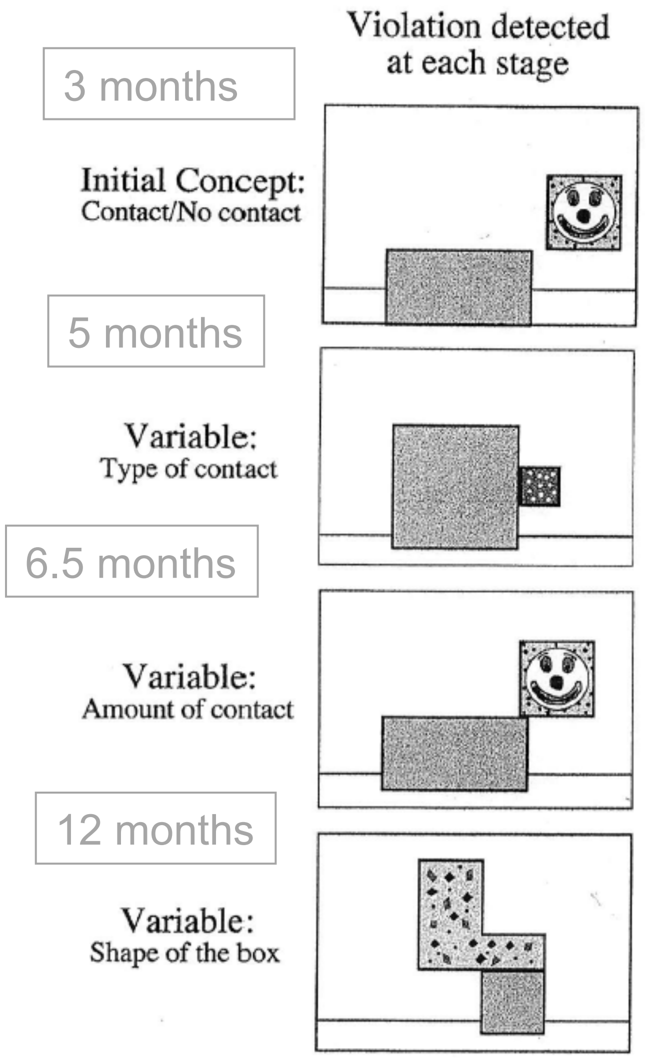

(Baillargeon)

# The development of intuitive physics in humans

- 0-4 months:
  - Object permanence, spatiotemporal continuity, solidity, rig
- 6-7 m
  - St
- 8-10
  - Gr int
- 10-12
  - Center of mass, weight, shape constancy, object tracking integrated with intuitive psychology for joint attention and intention.

Can we...

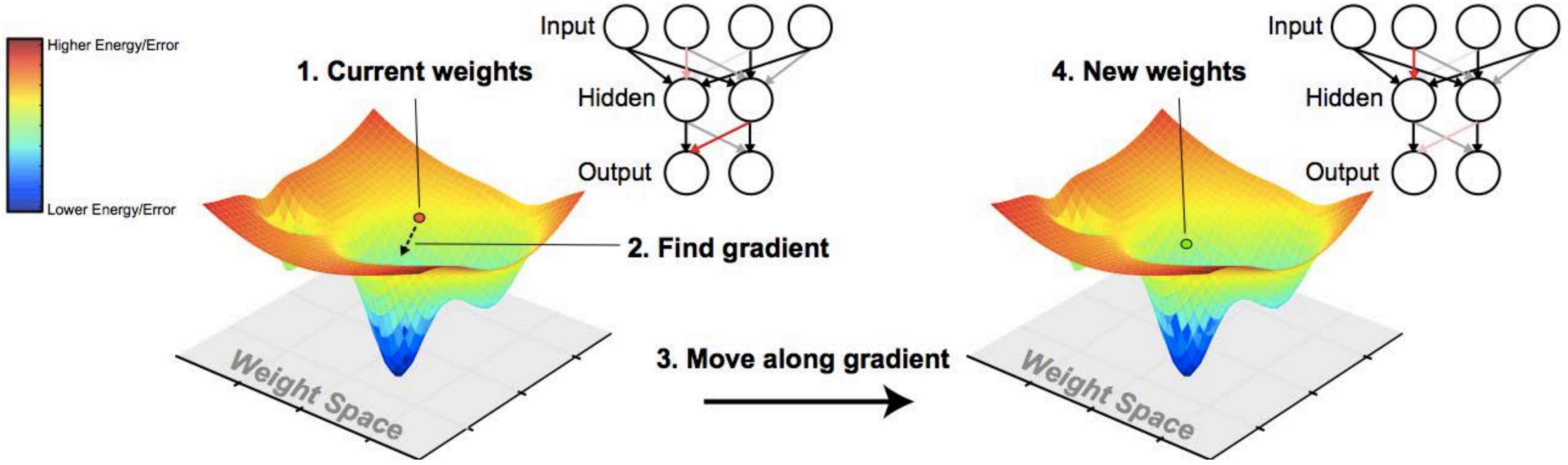
- Measure precisely the stages and learning trajectories that children follow?
- Capture different knowledge stages with a sequence of game-engine programs?
- Explain the trajectory of stages as some kind of rational search in the space of programs?



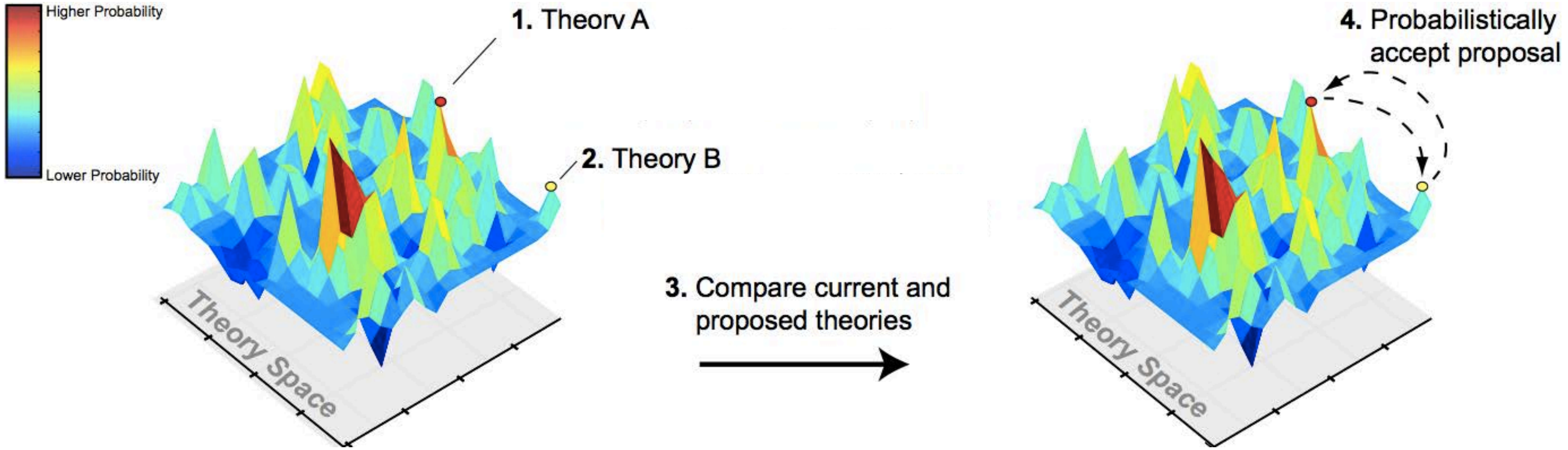

(Baillargeon)

# The hard problem of learning

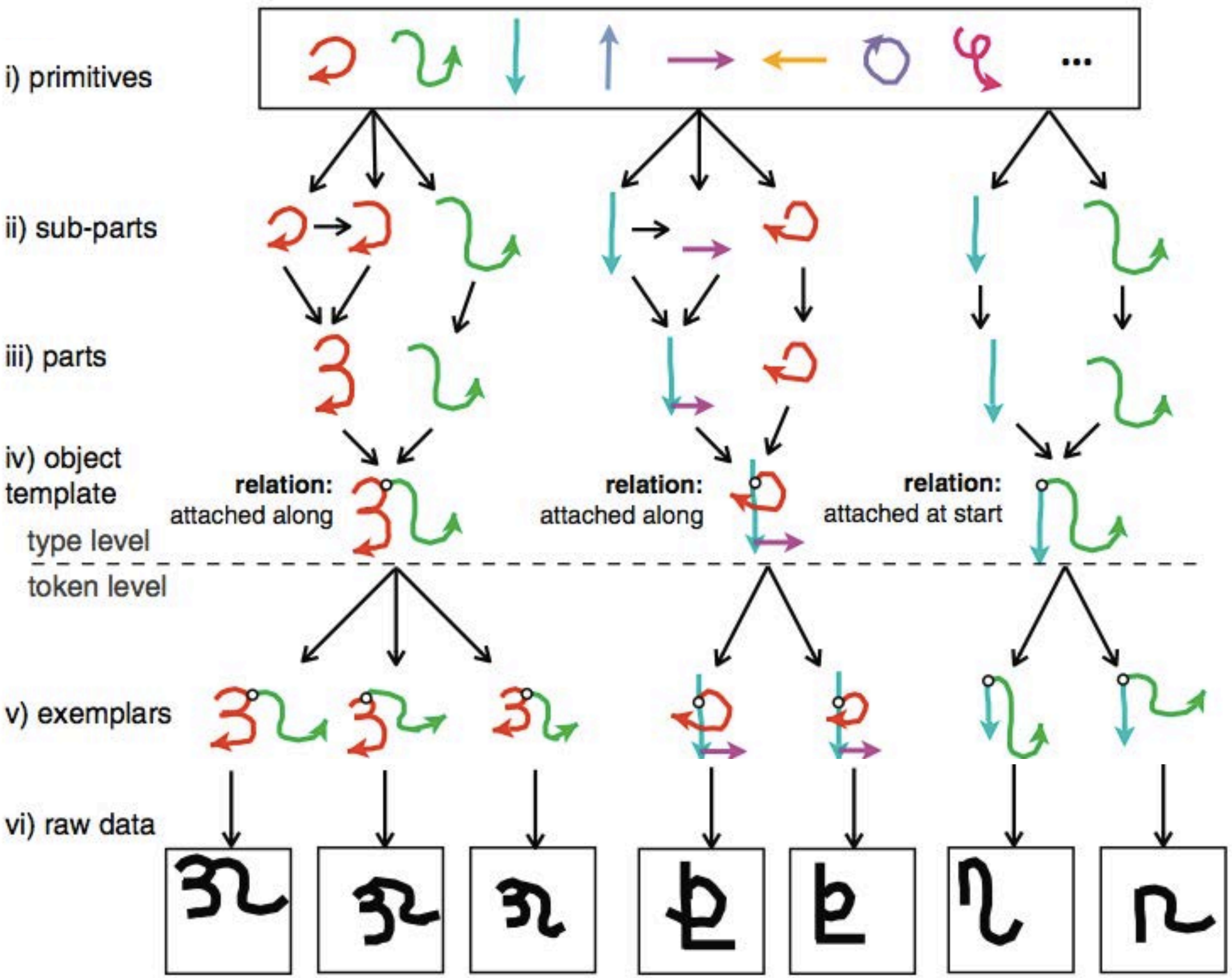
Search in neural network weight space:

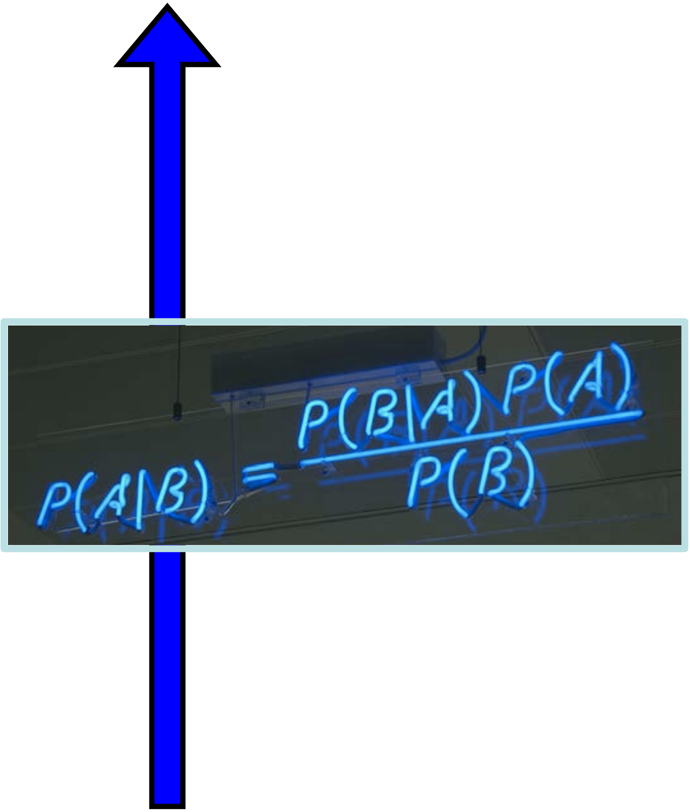



Search in the space of programs:

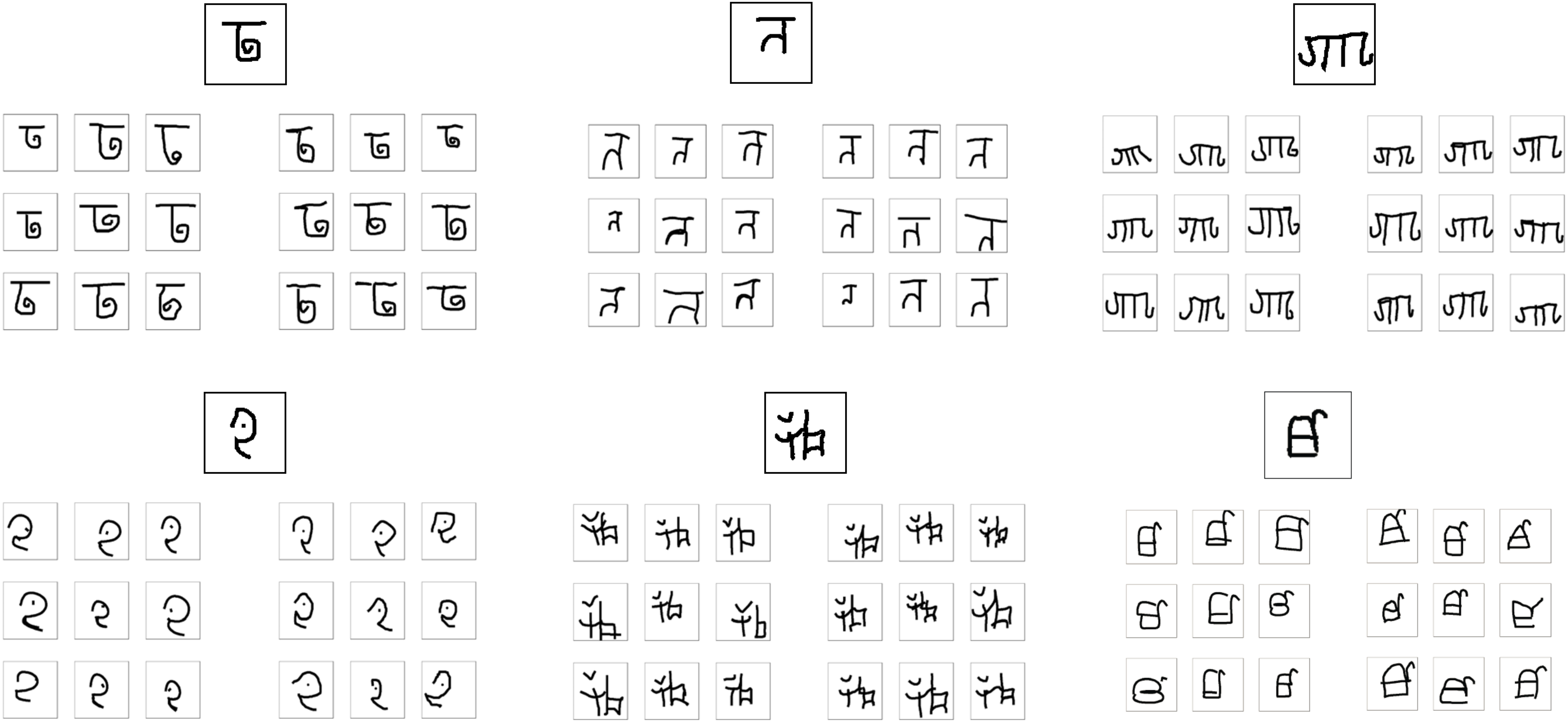


# Bayesian Program Learning

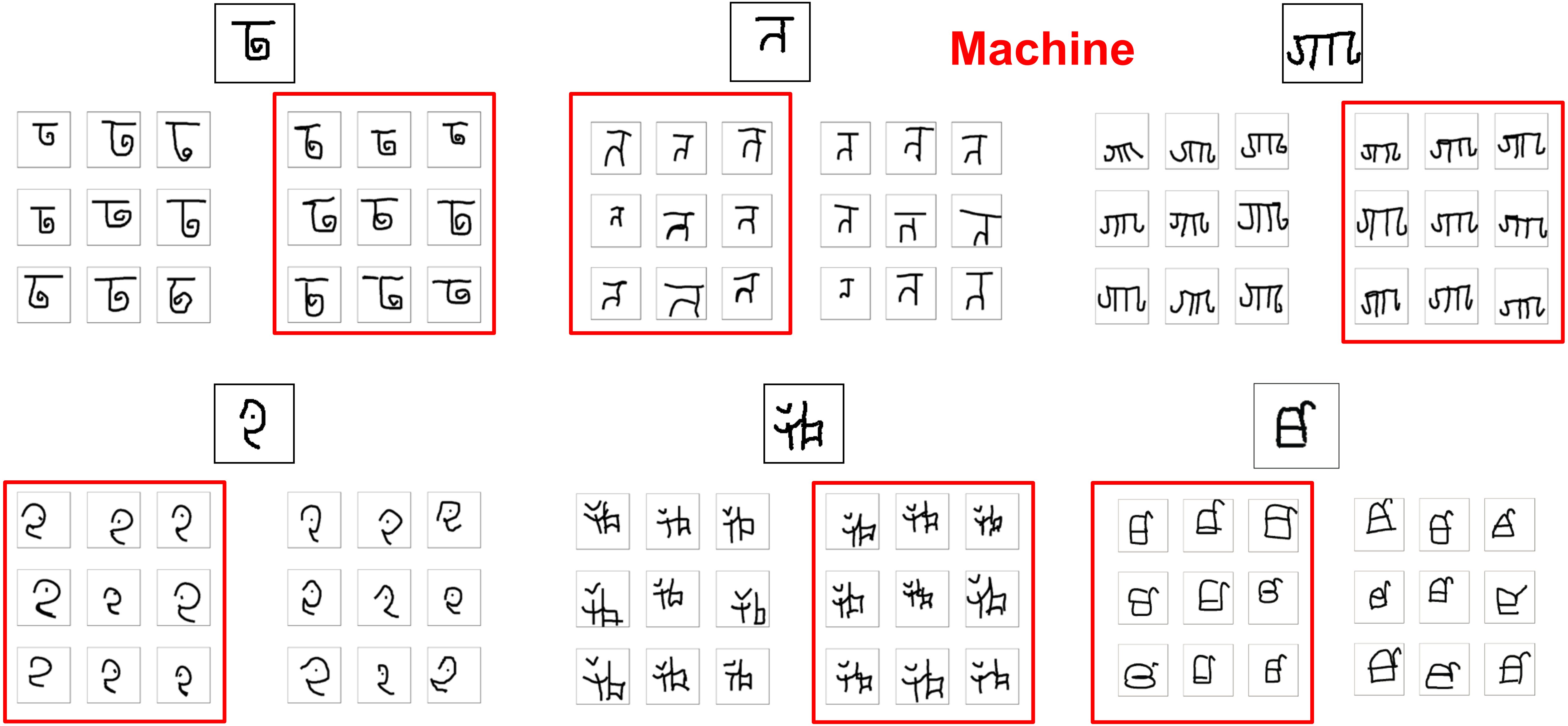


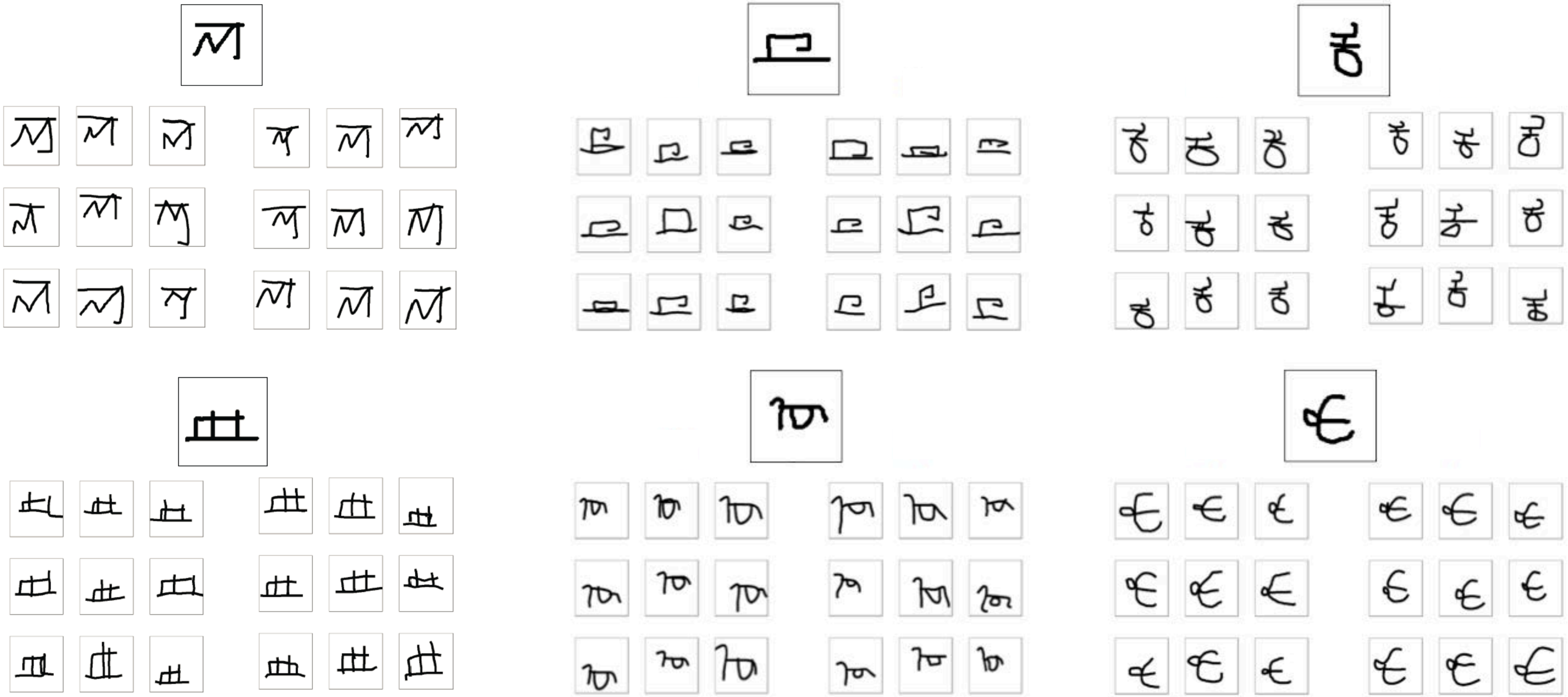
# A “Turing test” for program learning



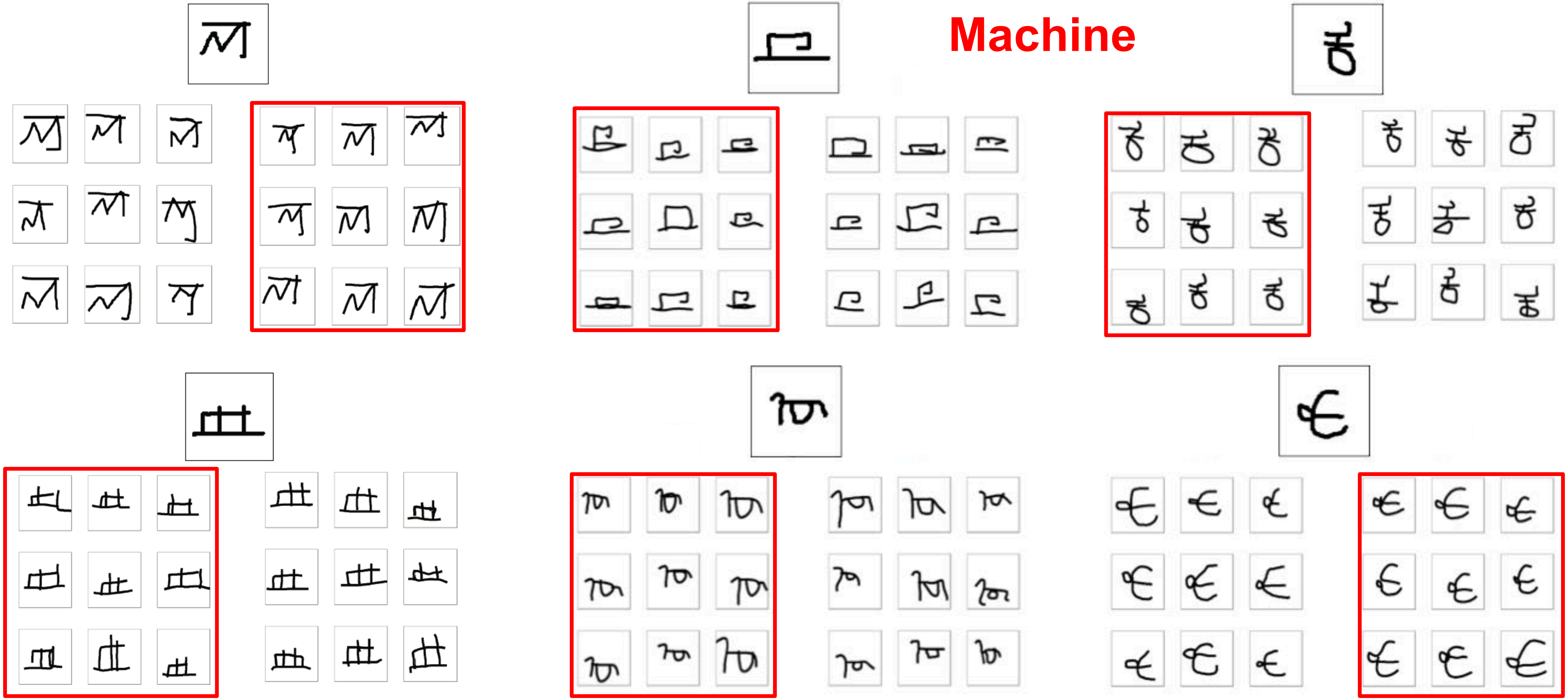
# A "Turing test" for program learning



# A “Turing test” for program learning



# A "Turing test" for program learning



# Bayesian program learning for richer concepts

## Cultural symbols

spoken words

| word | phonemes |
| --- | --- |
| "mouse" | imgresmaUs |
| "house" | haUs |
| "spouse" | spaUs |

dance

gestures

## Objects (natural, human-made)

## 3D Shape programs

Input

Reconstruction after adaption

```
draw('Top','Cir',(P=(0,0,0),G=(2,6)))

draw('Support','Cyl',P=(-11,0,0), G=(13,1))

for(i<5,'Rot',θrot=72,ax=(-10,0,0))
 draw('Base','Line',P=(-10,0,0),
      G=(-11,-6,-3),θrot×i, ax)

draw('TiltBack','Cub',P=(3,2,-5),G=(8,2,9,7))

for(i<2,'Trans',u1=(0,0,11))
 for(j<2,'Trans',u2=(0,4,0))
  draw('ChairBeam','Cub',P=(2,-4,-6)
   +(j×u2)+(i×u1),G=(3,1,2))

for(i<2,'Trans',u=(0,0,10))
 draw('HoriBar','Cub',P=(4,-4,-6)
   +(i×u),G=(1,5,2))
```

# Learning as programming
## ("The child as coder" / "The child as hacker")

The goal of learning: *Making your code more awesome*.

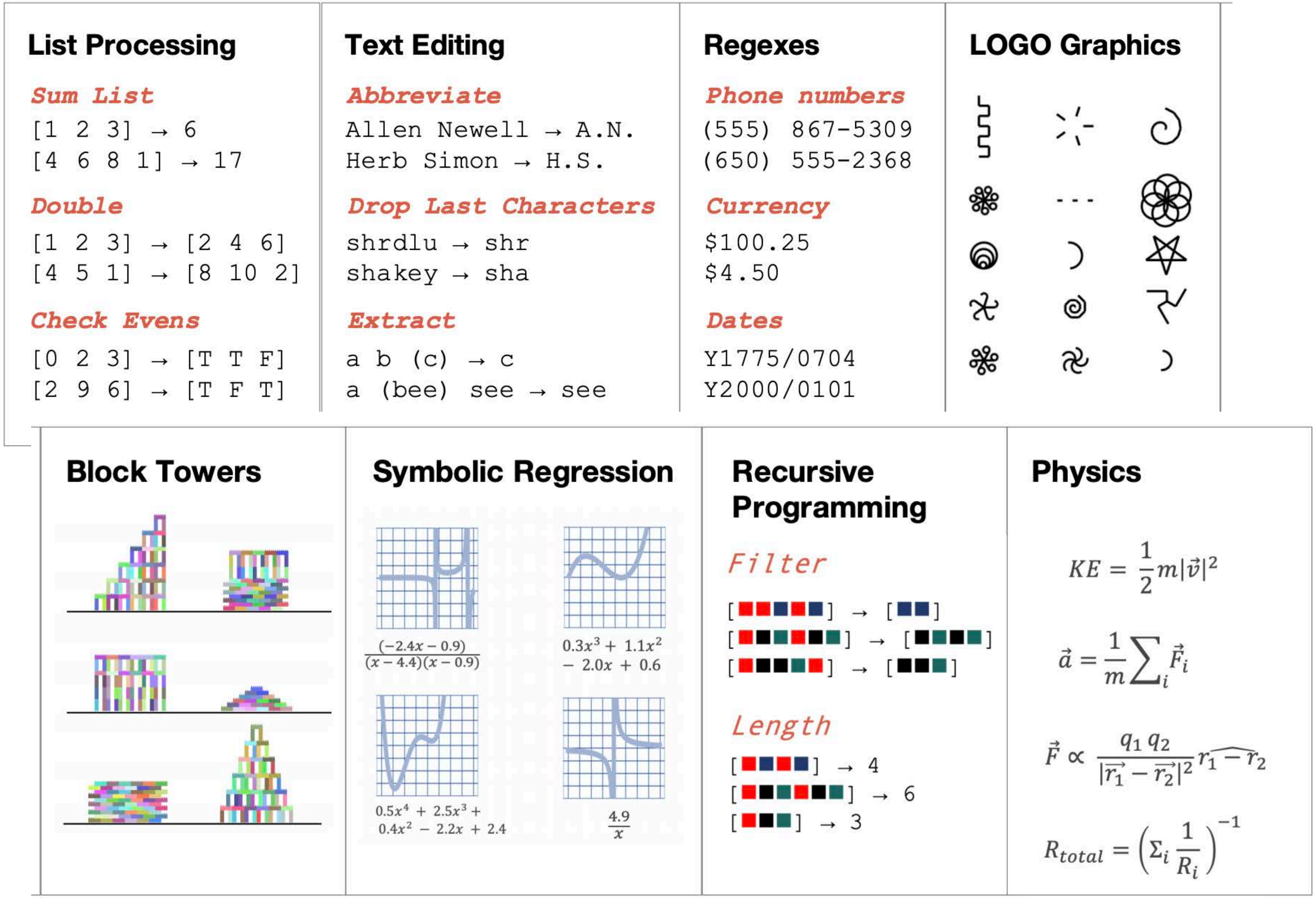
Think about all the ways you modify code to achieve this goal:

- Tuning parameters of existing functions
- Extending or fixing existing functions
- Debugging (finding and removing faulty (inaccurate, not robust) code)
- Rewriting (e.g., cleaning up, refactoring) a library of existing functions
- Adapting existing code written for other purposes
- Getting code from other people or published sources
- Translating existing code to a different language
- Compiling code (from interpretable high-level language -> efficient low-level language)
- Writing a new programming language or compiler

… All of these activities have analogs in human learning, and we need to understand every one of them algorithmically.

DREAMCODER: Growing human-like abstract knowledge with wake-sleep Bayesian program learning

(Ellis, Wong, Nye, Morales, Carey, Hewitt, Sable-Meyer, Solar-Lezama, Tenenbaum)

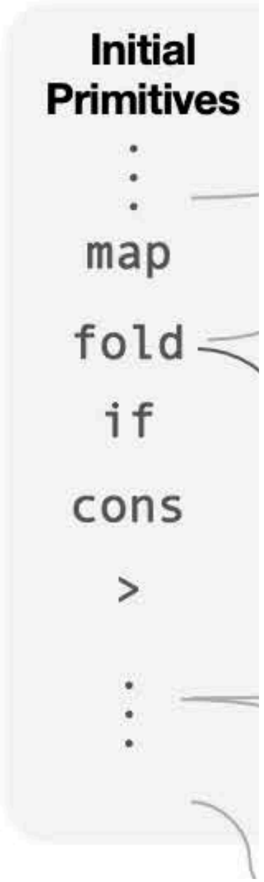


## Discovered Problem Solutions

**Problem: `Sort List`**

[9 2 7 1] -> [1 2 7 9]

[3 8 9 4 2] -> [2 3 4 8 9]



## Discovered Problem Solutions

**Problem: Sort List**

[9 2 7 1] -> [1 2 7 9]

[3 8 9 4 2] -> [2 3 4 8 9]

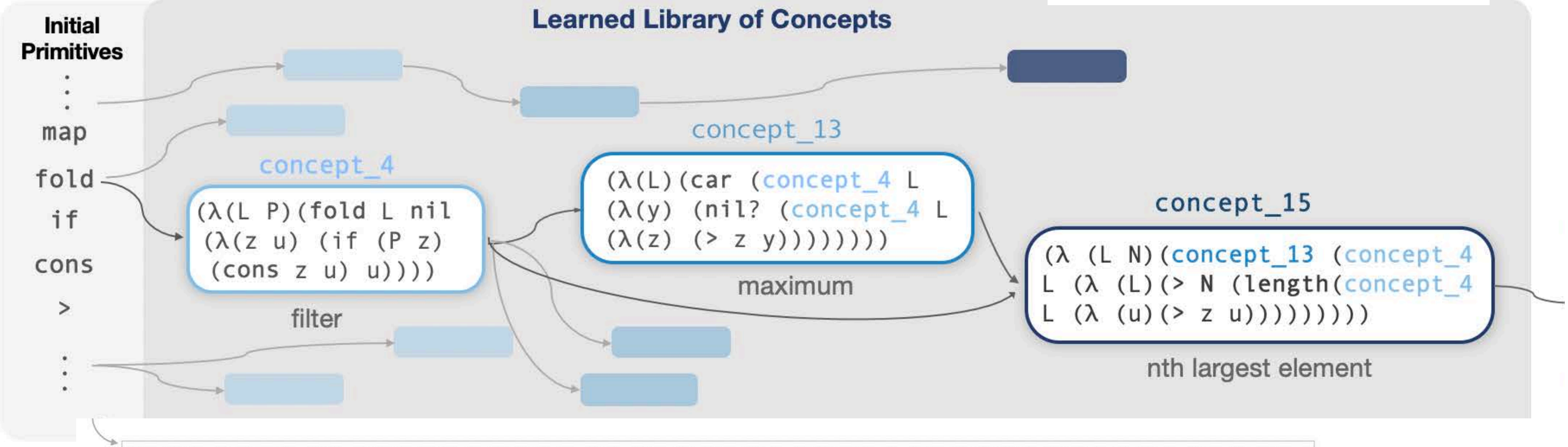
...

**Solution in learned language:**

```
(map (l (n)
  (concept_15 L (+ 1 n)))
  (range (length L)))
```

get nth largest element

where n = 1, 2, 3 ….length of list

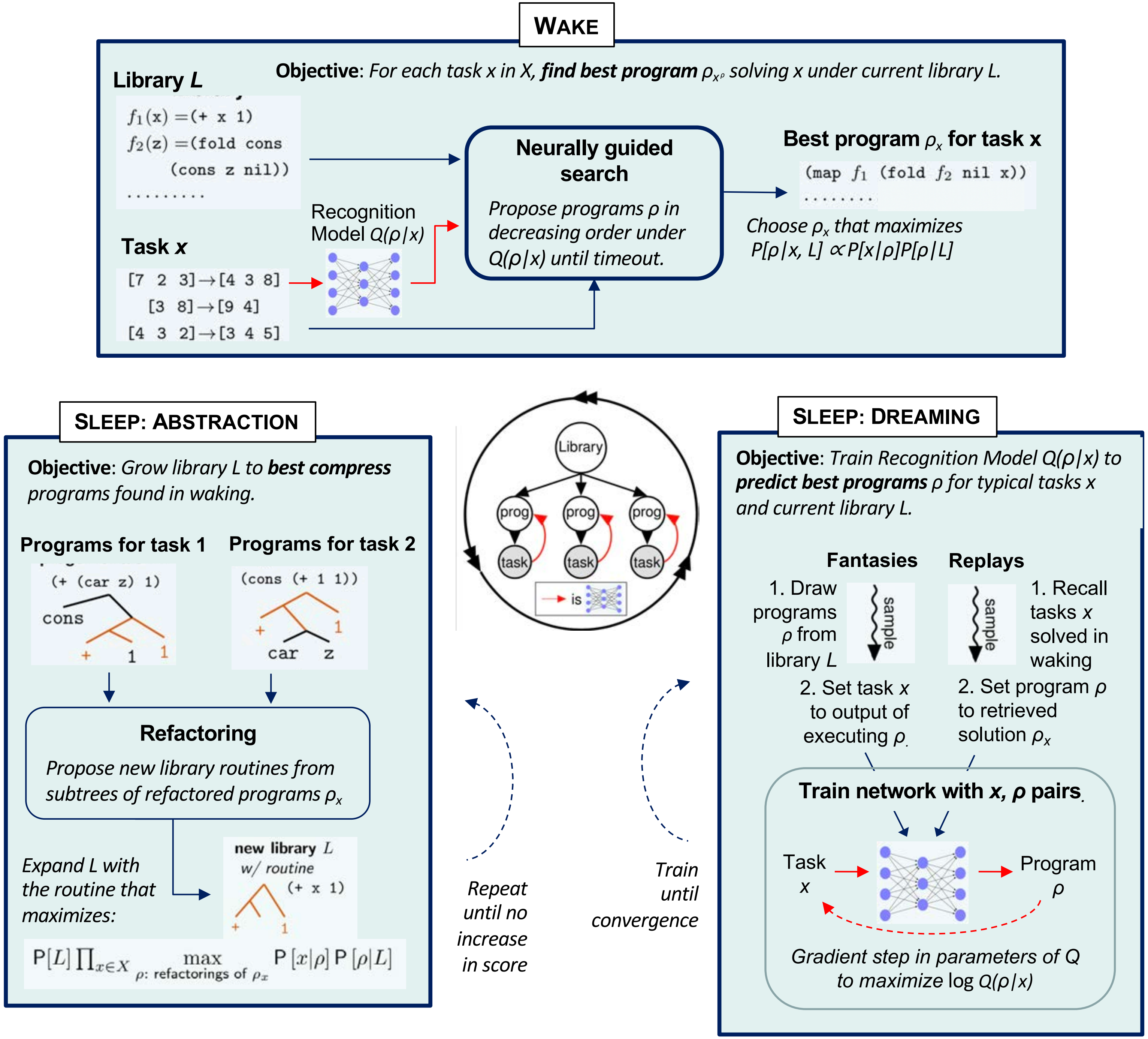



```
(λ (x) (map (λ (y) (car (fold (fold x nil (λ (z u) (if (gt? (+ y 1) (length (fold x nil (λ (v w)
(if (gt? z v) (cons v w) w))))) (cons z u) u))) nil (λ (a b) (if (nil? (fold (fold x nil (λ (c d)
(if (gt? (+ y 1) (length (fold x nil (λ (e f) (if (gt? c e) (cons e f) f))))) (cons c d) d))) nil
(λ (g h) (if (gt? g a) (cons g h) h)))) (cons a b) b))))) (range (length x))))
```

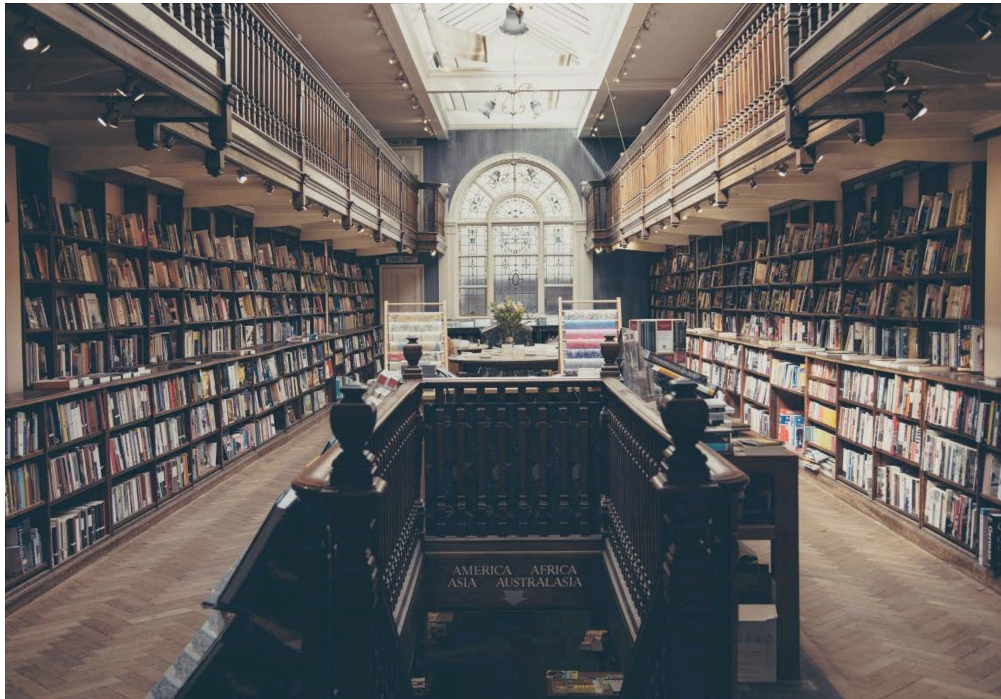
Solution expressed in initial primitives

# DreamCoder: Growing human-like abstract knowledge with wake-sleep Bayesian program learning

(Ellis, Wong, Nye, Morales, Carey, Hewitt, Sable-Meyer, Solar-Lezama, Tenenbaum)




Dr. Seuss
The Butter Battle Book
Dr. Seuss


AMERICA AFRICA
ASIA AUSTRALASIA

# How to grow a mind: A roadmap

Looking forward, can we fulfill AI's oldest dream, to build a machine that grows intelligence the way a human being does? And thereby come to understand better how our own minds are built?

**What is the the starting state (inductive bias)?**
More content than we might have thought, some of it very structured:
*"Core cognition"*
*"The game engine in your head"*

Game engine-style intuitive physics:
Objects + interactions (forces)

... and intuitive psychology:
Agents + intentions (utilities)

**What are the learning procedures?**
More mechanisms than we might have thought, some of them very smart:
*"The child as scientist"*
*"The child as coder"*

Probabilistic programs
Program induction
Program synthesis