Neural Architects Workshop

28th October, ICCV 2019

Capsule Architectures

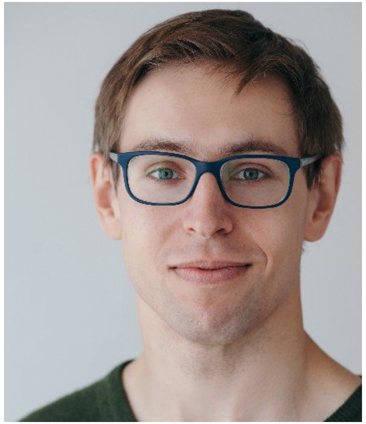
Sara Sabour

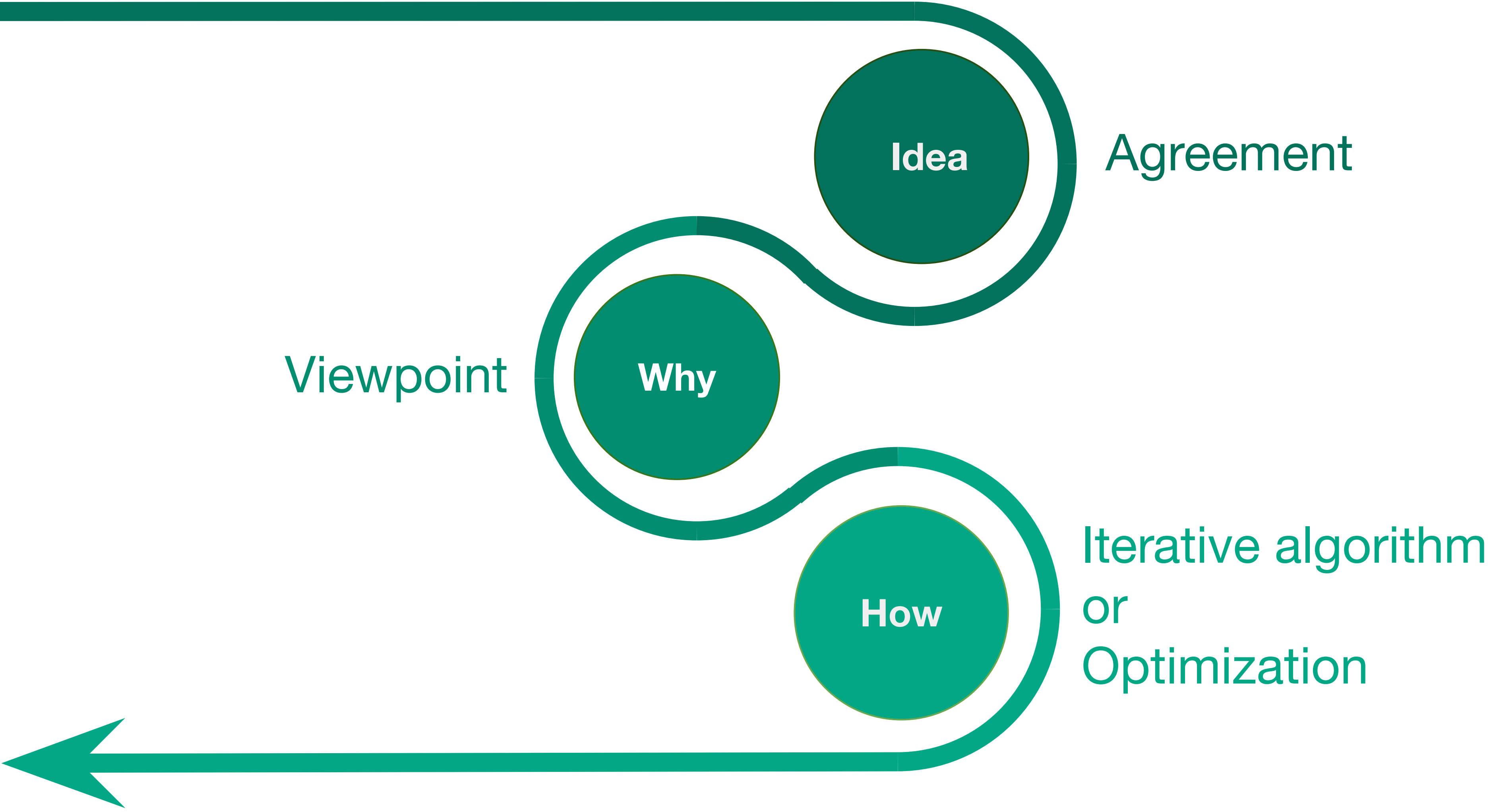
Google Brain, University of Toronto

Joint work with

- Geoff Hinton @Google brain
- Nicholas Frosst @Google brain
- Adam Kosiorek @Oxford University
- Yee Whye Teh @Oxford & Deepmind



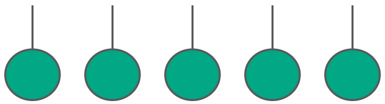




Idea
Agreement
Viewpoint
Why
How
Iterative algorithm
or
Optimization

Idea 101: Agreement and Capsules

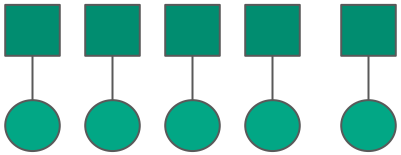
Close look at a typical non-linearity

1. Each neuron is multiplied by a trainable parameter.



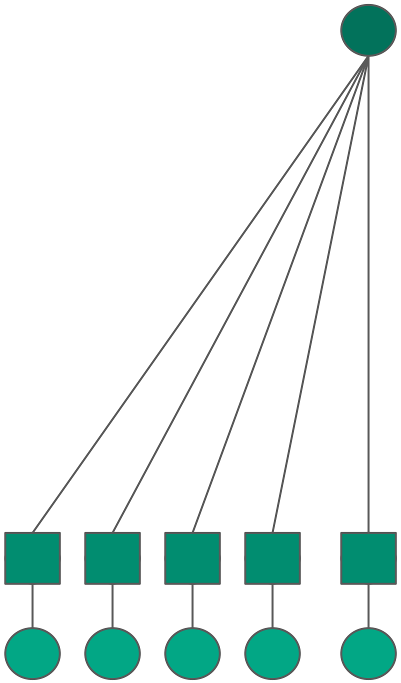
Close look at a typical non-linearity

1. Each neuron is multiplied by a trainable parameter.



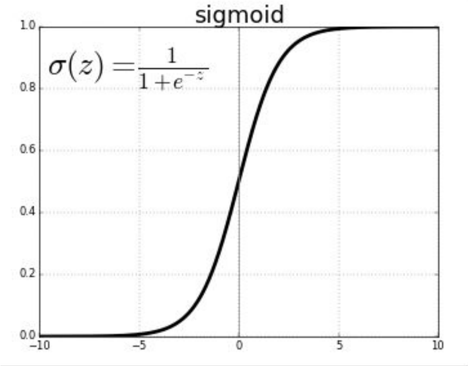
Close look at a typical non-linearity

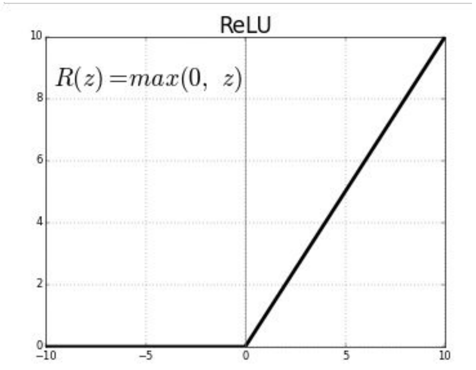
1. Each neuron is multiplied by a trainable parameter.
2. The incoming votes are summed.

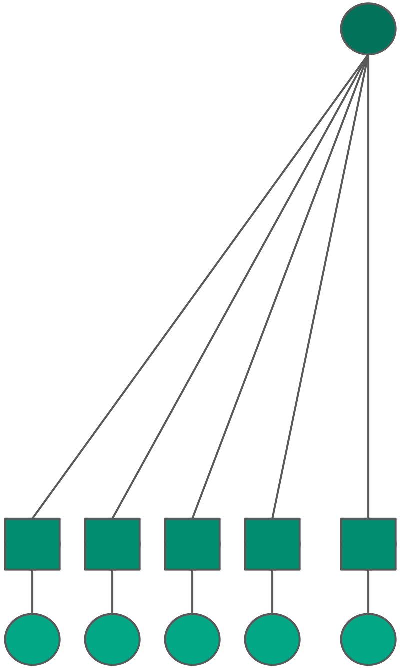


Close look at a typical non-linearity

1. Each neuron is multiplied by a trainable parameter.
2. The incoming votes are summed.
3. A nonlinearity (ReLU) is applied where a higher sum means more activated.



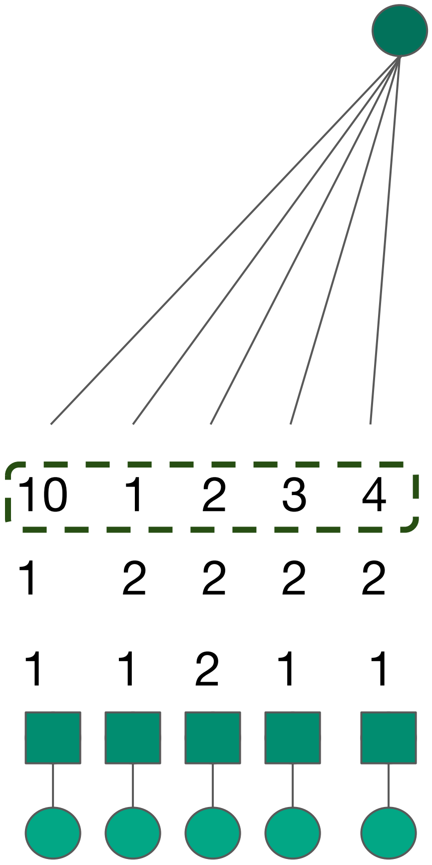




Close look at a typical non-linearity

1. Each neuron is multiplied by a trainable parameter.
2. The incoming votes are summed.
3. A nonlinearity (ReLU) is applied where a higher sum means more activated.

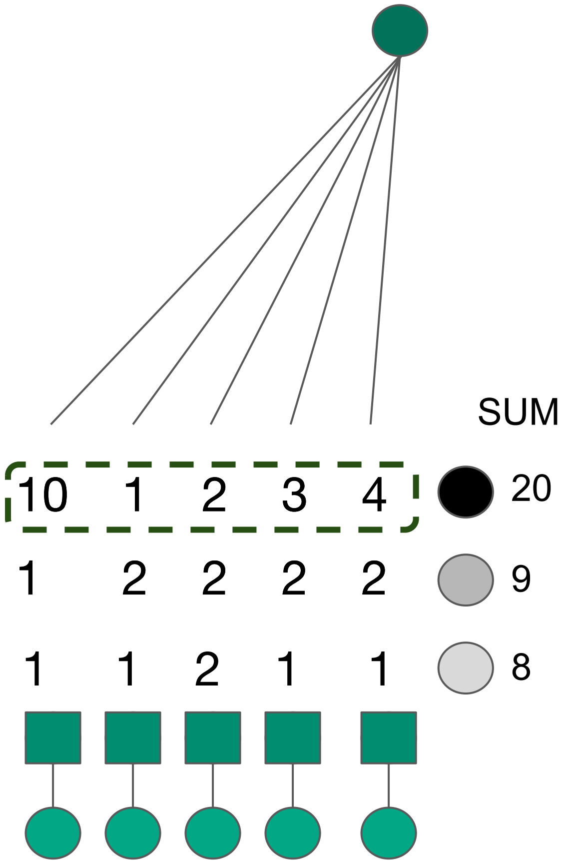
Consider these three cases:



Close look at a typical non-linearity

1. Each neuron is multiplied by a trainable parameter.
2. The incoming votes are summed.
3. A nonlinearity (ReLU) is applied where a higher sum means more activated.

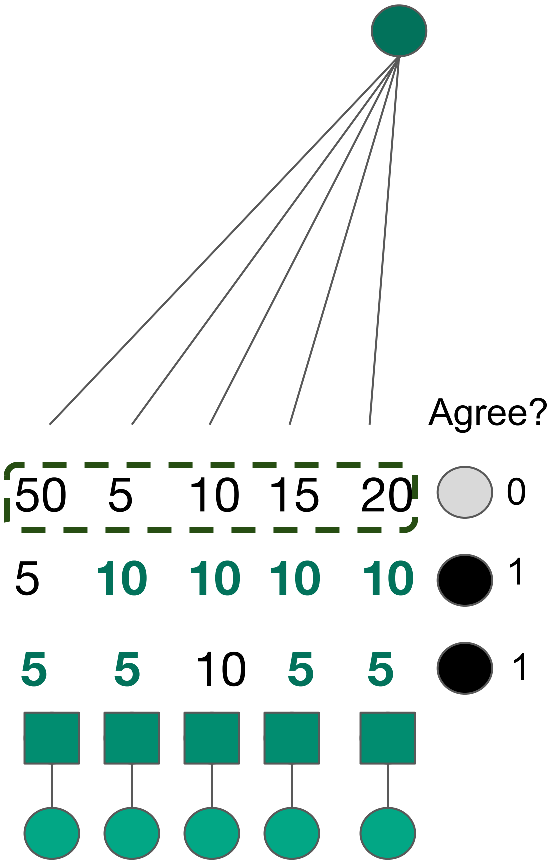
Consider these three cases:



Dictatorship
Support comes from a confident shouter!

Agreement Invariance

1. Each neuron is multiplied by a trainable parameter.
2. Do they agree with each other.

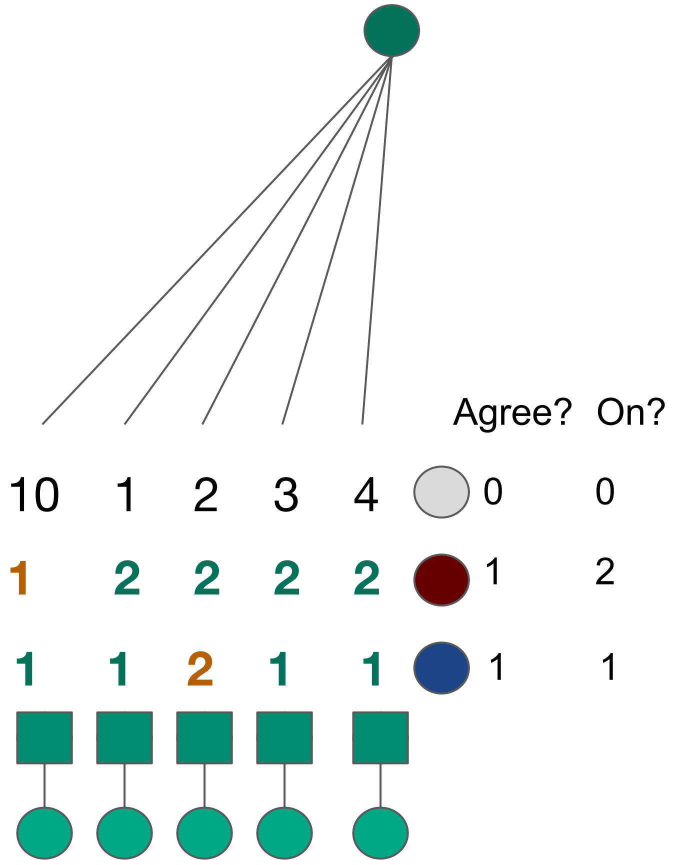


Democracy
Support comes from coordinated mass!

SUM + ReLU -------------> Count

Agreement, enhanced
Invariance
Equivarience

1. Each neuron is multiplied by a trainable parameter.
2. Do they agree with each other.
3. What are they agreeing upon.

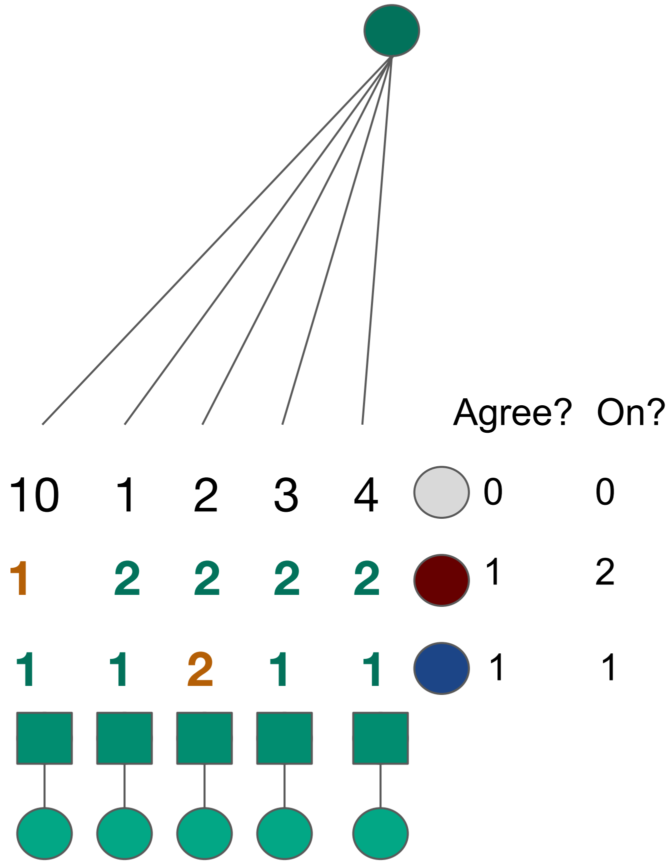


No loss of information!

If 5 is multiplied to everything, what they are agreeing upon will be multiplied by 5.

Agreement, what we get?
Invariance
Equivarience

1. Each neuron is multiplied by a trainable parameter.
2. Do they agree with each other.
3. What are they agreeing upon.



Training with this non-linearity
- Counting: Non-differentiable
- Similarity function: differentiable

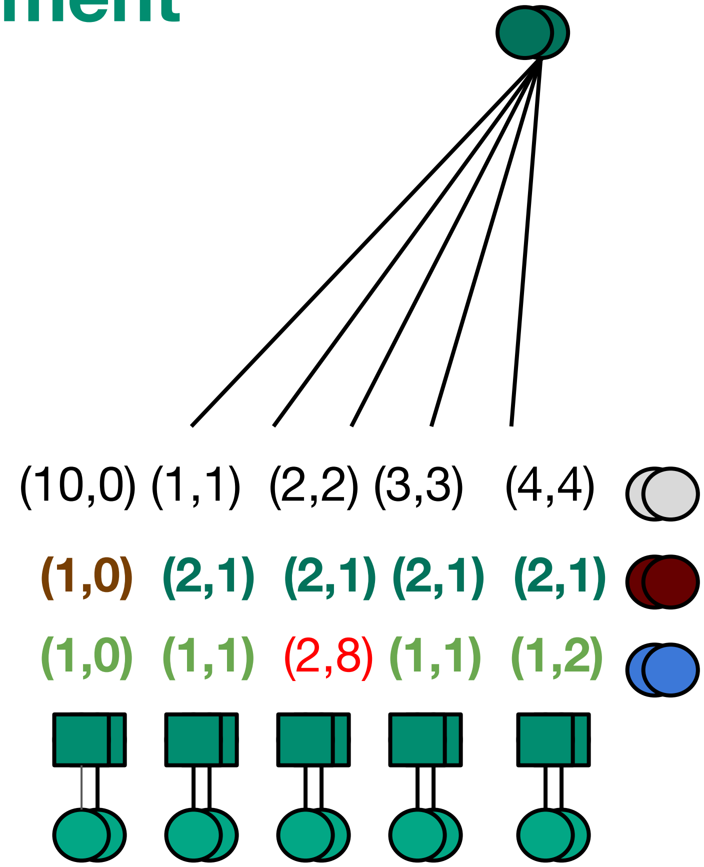
Multi Dimension Enhanced Agreement

Stronger Invariance

Stronger Equivarience

1. Each neuron is multiplied by a trainable parameter.
2. Do they agree with each other.
3. What are they agreeing upon.

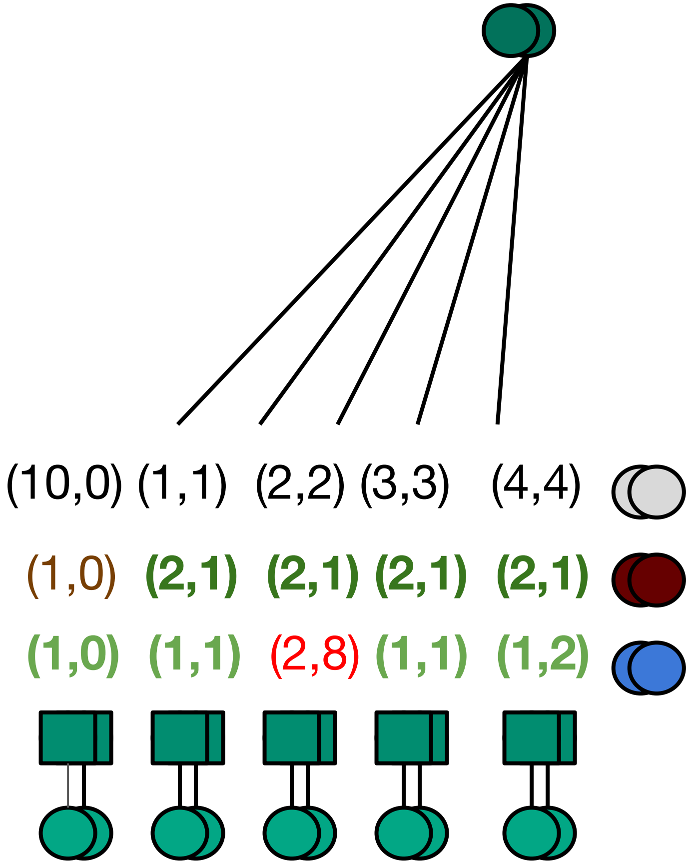
Stronger and more robust agreement finding.



Recap

- Base idea
  Agreement non-linearity
  How many are the same
  rather than who is larger
- Enhancements
  - Presence + Value
  - Multi-Dimensional Value

New neurons: Capsules



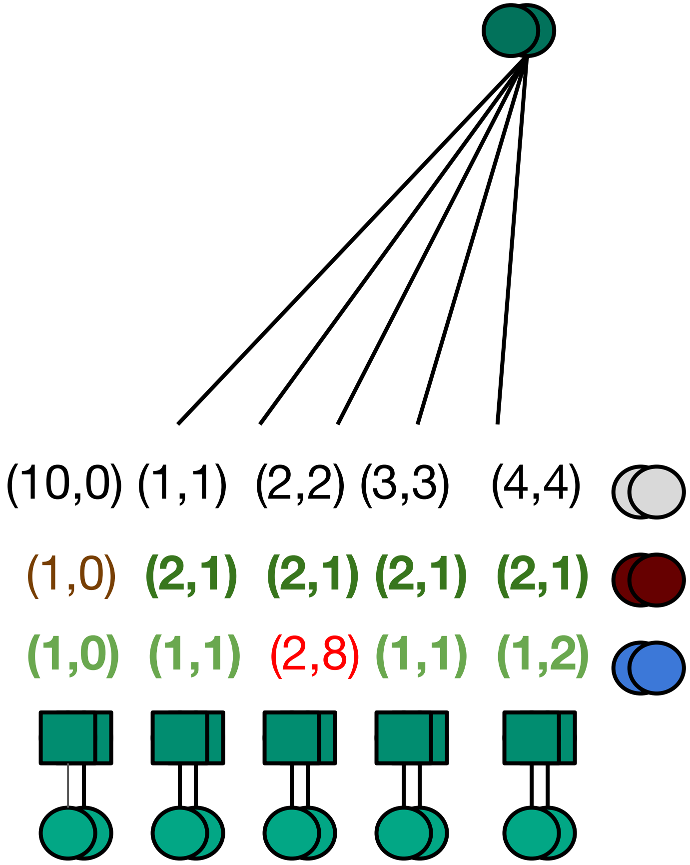
Recap: Capsules

- Base idea

  Agreement non-linearity
  How many are the same
  rather than who is larger
- Enhancements
  - Presence + Value
  - Multi-Dimensional Value

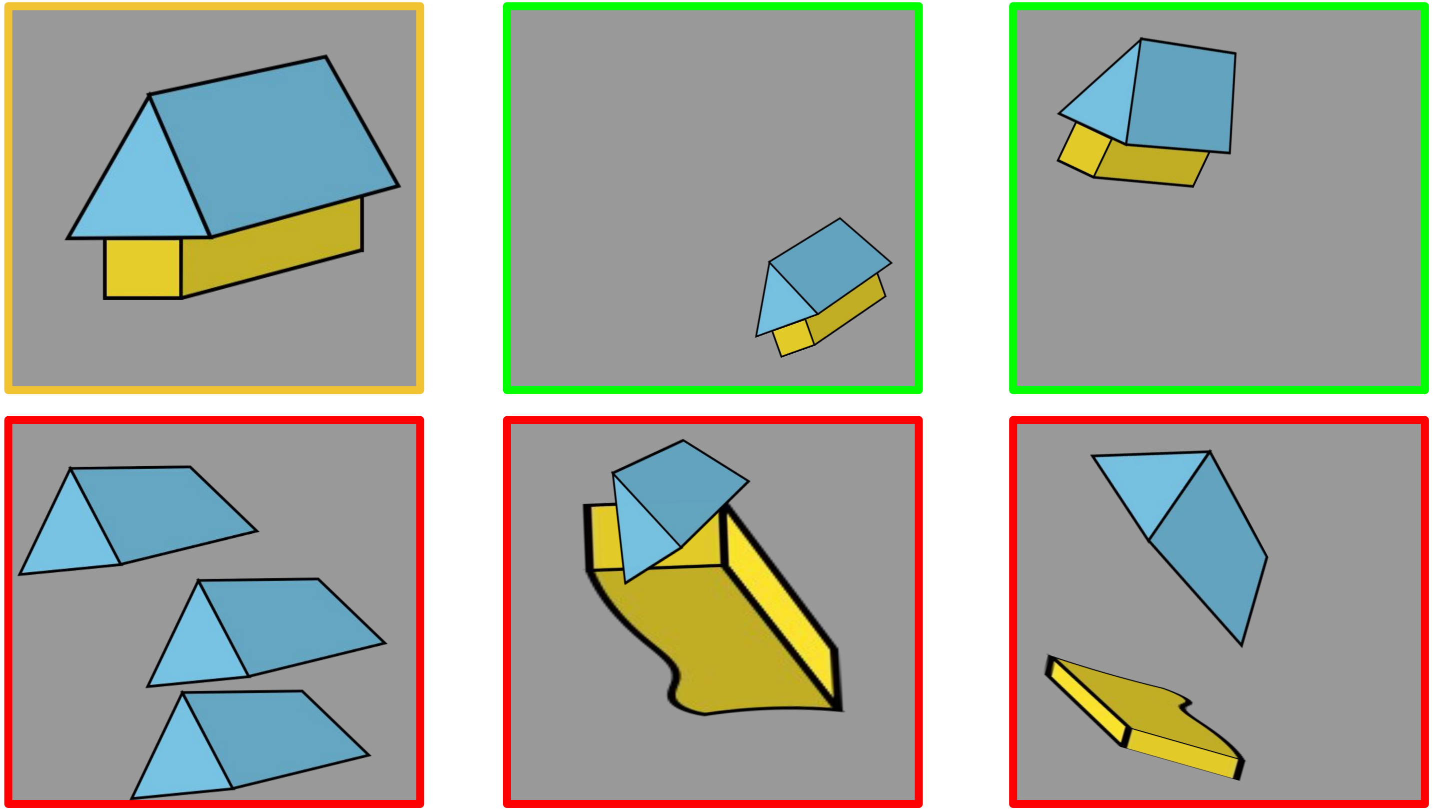
A network of Capsules

- Each capsule has whether it is present and how it is present.
- Each capsule gets activated if incoming votes agree.



Use Case: Computer Vision

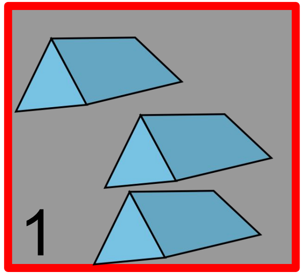
Which one is a house?

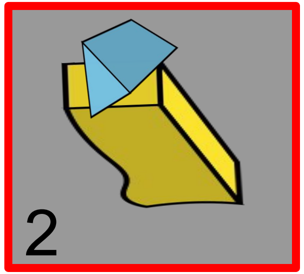


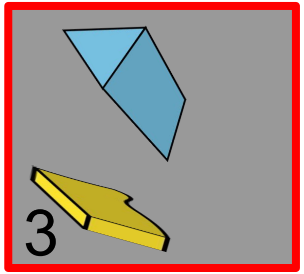
Which one is a house?

1. Both the parts should exist.
   - Image 1 is not a house.

2. How the roof and the walls exist should match a common house.
   - Image 2 & 3 are not houses.

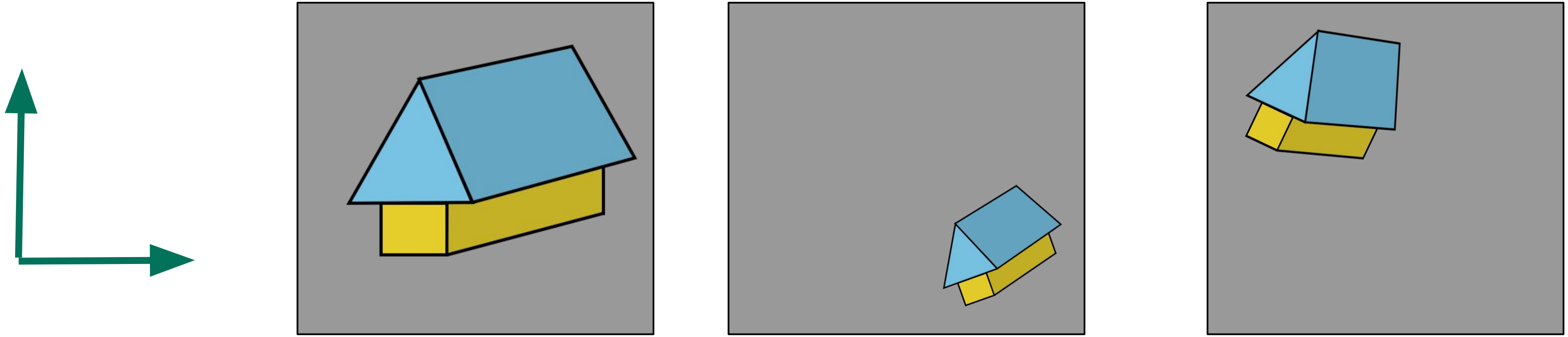






What stays constant?

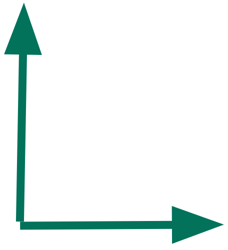
The relation between a part and the whole stays constant.



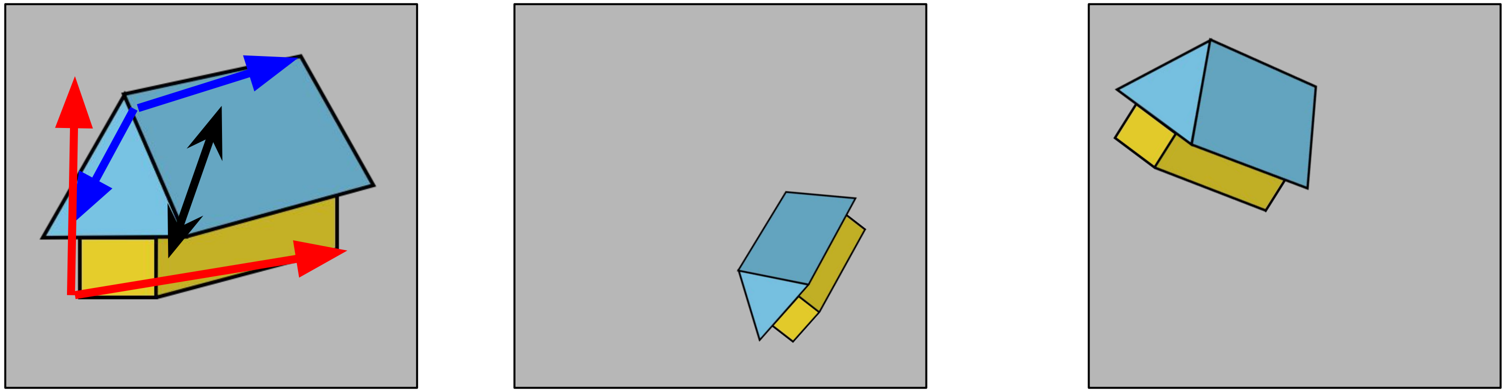
Camera
Coordinate
Frame

What stays constant?

The relation between a part and the whole stays constant:
Between the Roof arrows and the House arrows.

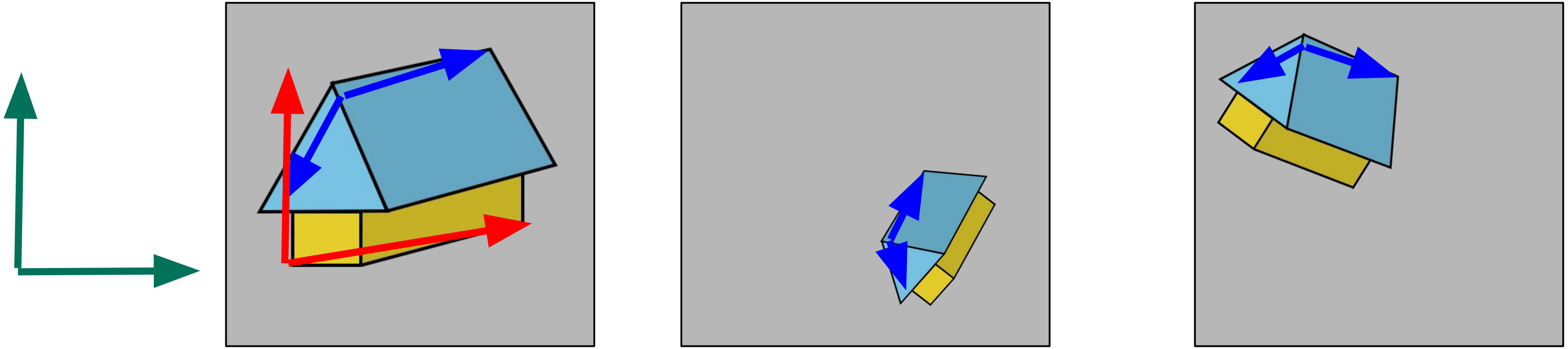


Camera
coordinate
Frame

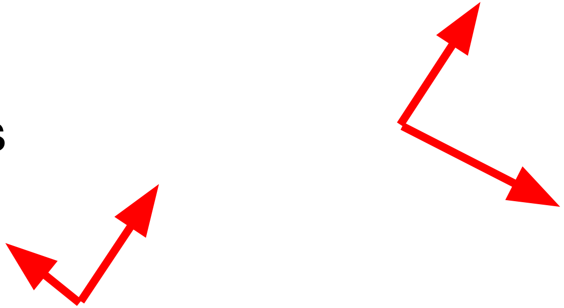


What stays constant?

The relation between a part and the whole stays constant:
Between the Roof arrows and the House arrows.

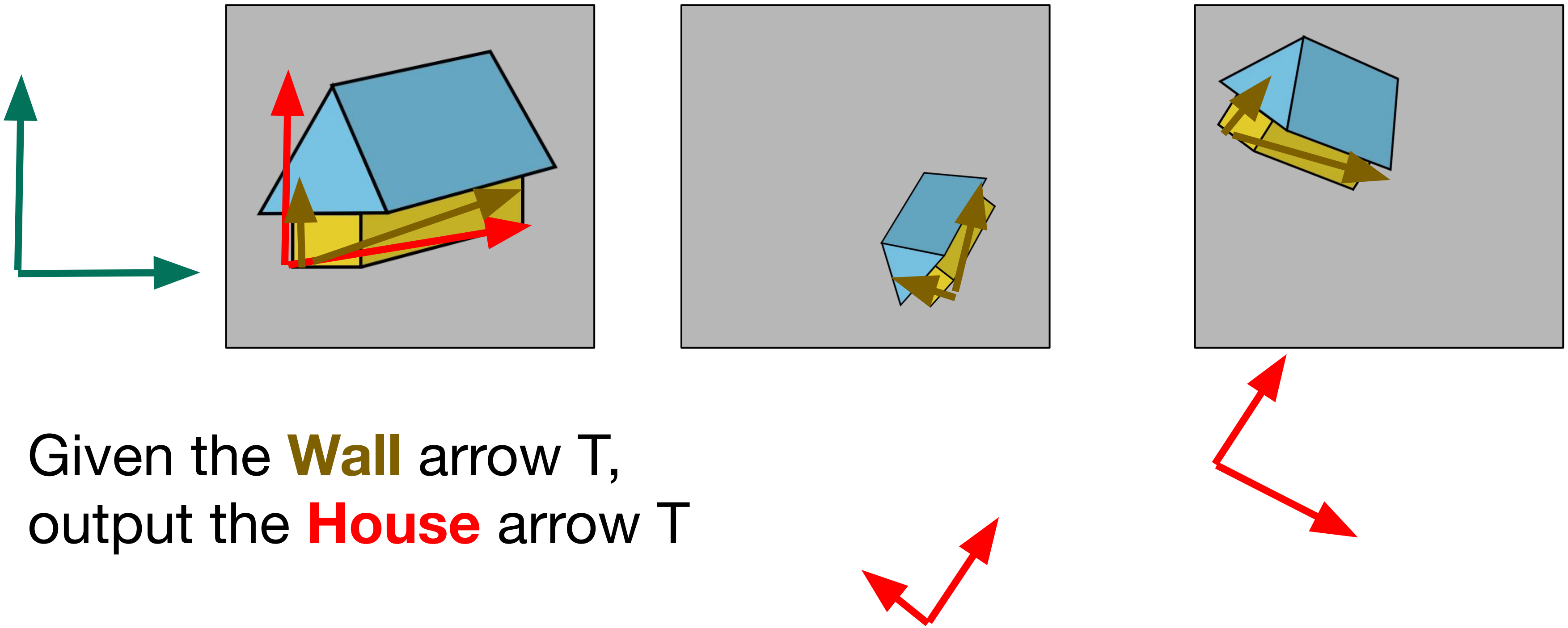


Given the Roof arrow transformation,
output the House arrow transformations



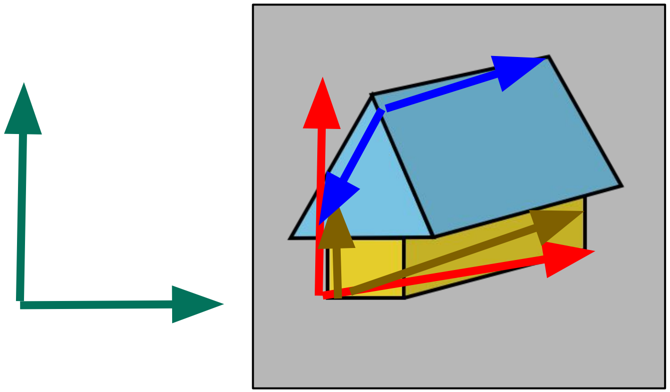
What stays constant?

The relation between a part and the whole stays constant:
Between the **Wall** arrows and the **House** arrows.



Given the **Wall** arrow T,
output the **House** arrow T

Recap



Input to the layer:

How to **transform** the **Camera** arrows
Into **Roof** and **Wall** arrows.

$$T_r \quad T_w$$

Output of the layer:

How to **transform** the **Camera** arrows
Into **House** arrows.

$$T_h$$

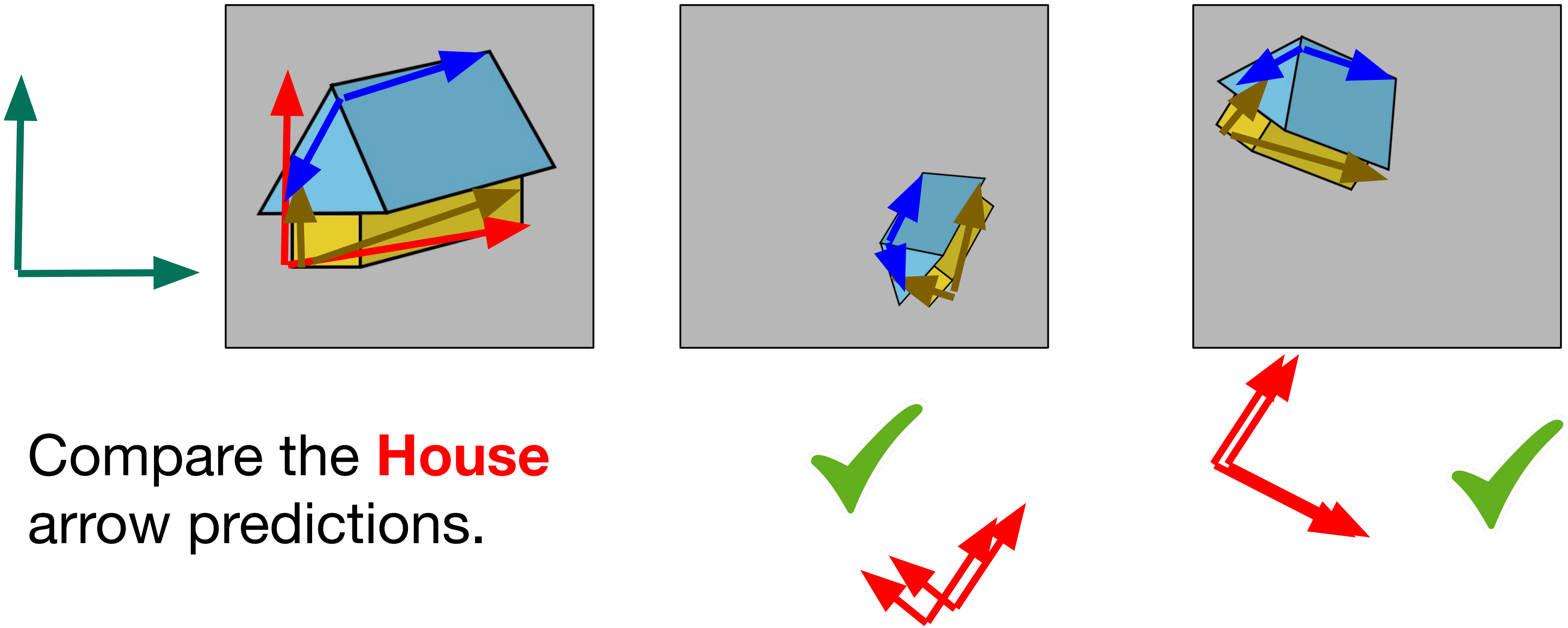
What we learn:

How to transform the transformations.

$$T_h = T_r W_{rh} \qquad T_h = T_w W_{wh}$$

What stays constant?

The relation between a part and the whole stays constant:
Between the **part** arrows and the **House** arrows.

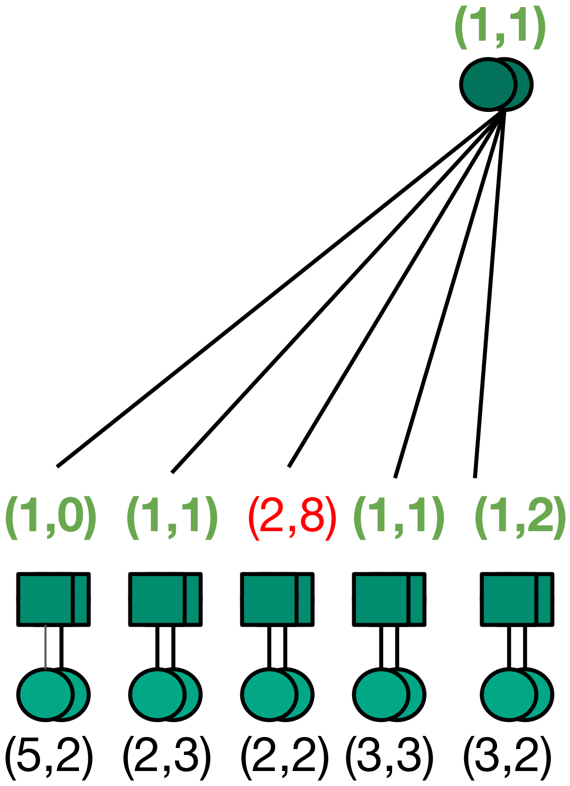


Compare the **House** arrow predictions.

Network of Capsules for Computer Vision

Each Capsule represents a part or an object.

- The presence of a capsule represents whether that entity exists in the image.
- The value of a capsule carries the spatial position of how that entity exists. I.e. the **transformation** between the coordinate frame of camera and the entity.
- The trainable parameter between two capsules is the **transformation** between their coordinate frame **transformations** as a part and a whole.



Capsule Network

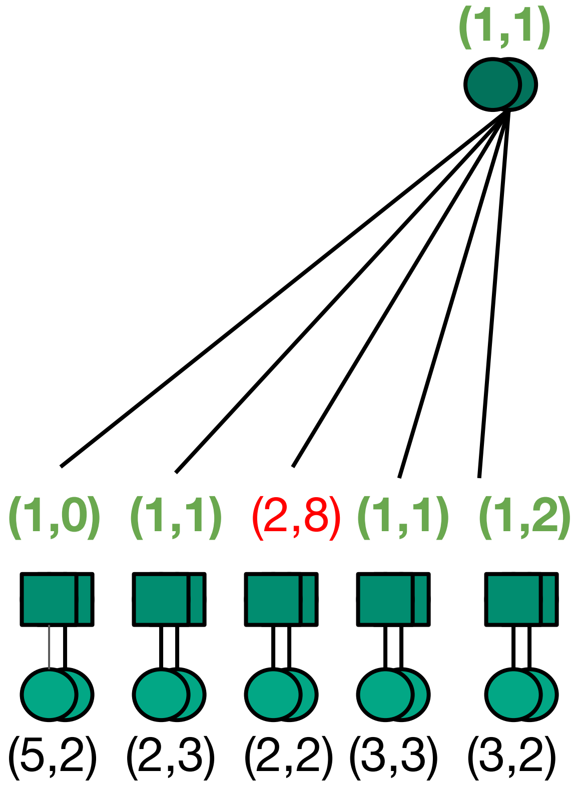
Same trained transformation works for all viewpoints of input.

- Input is transformed and so the value of the output capsule is transformed accordingly.
  Value is viewpoint equivariant.

$$T_{r'} = RT_r$$

$$T_{h'} = RT_h = RT_r W_{rh} = T_{r'} W_{rh}$$

- The agreement of parts would not change. Presence is viewpoint invariant.

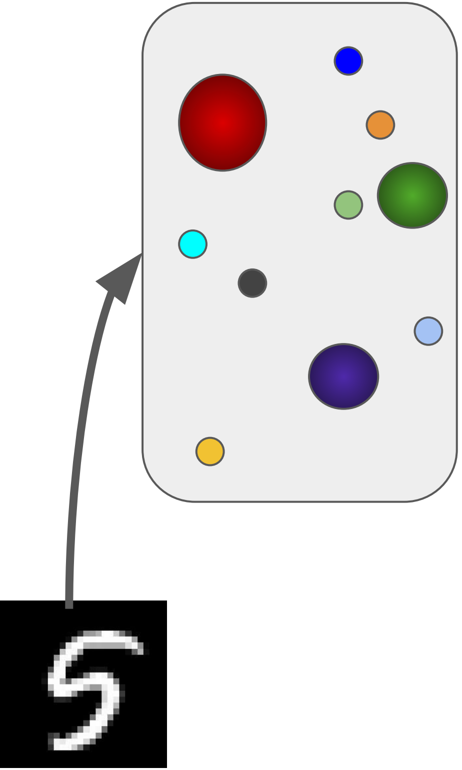


How: Iterative routing

Matrix Capsules with EM routing
Geoff Hinton   Nick Frosst
ICLR 2018

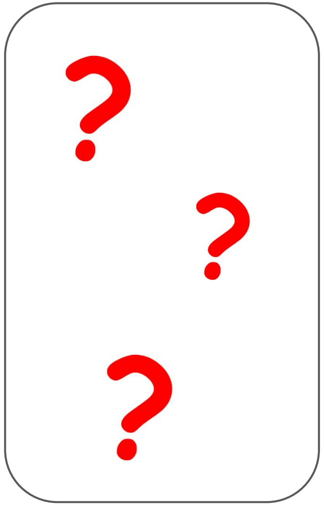
EM routing for Gaussian Capsules

Layer L

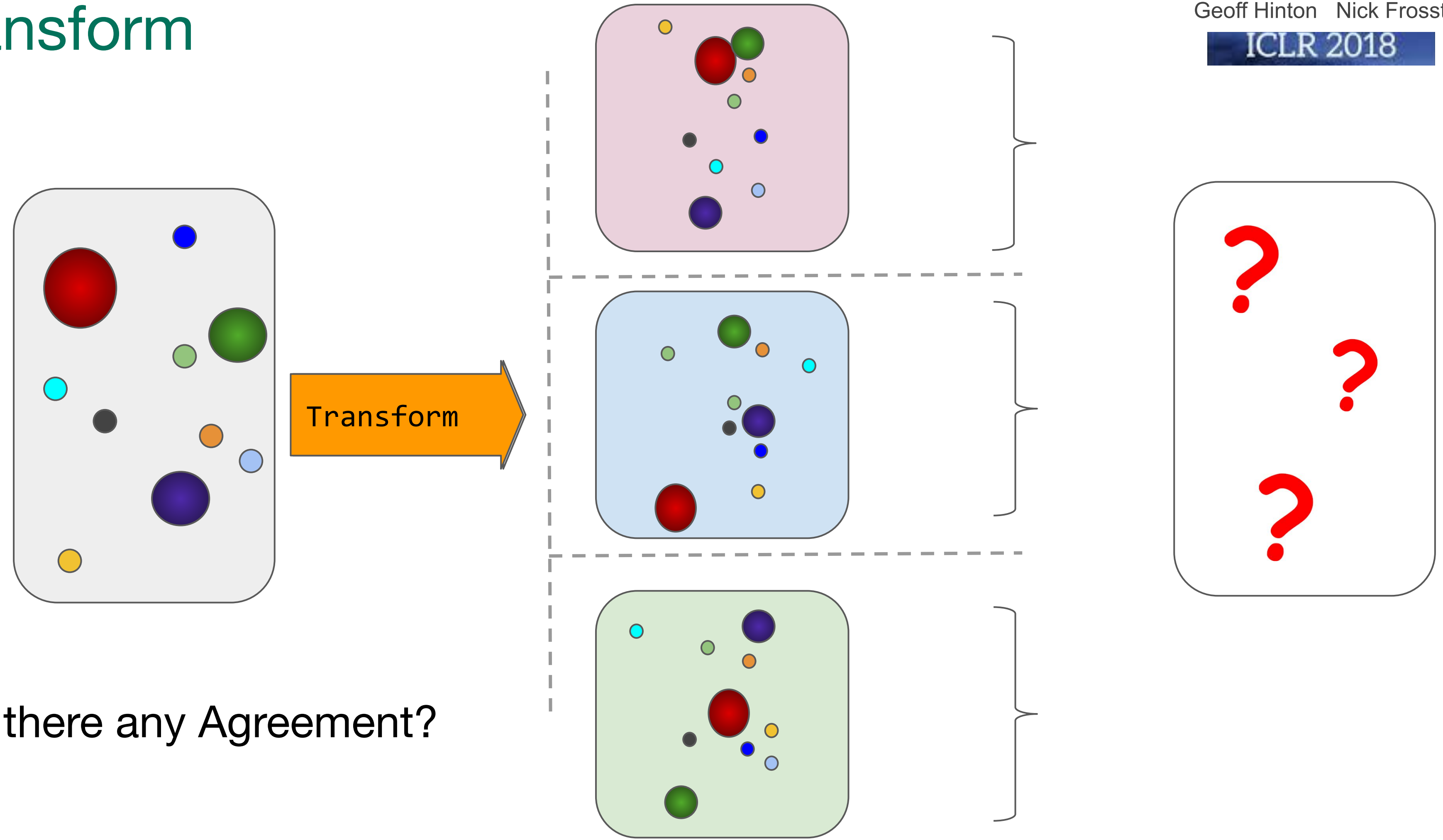


- 2D capsules
- Position shows their 2D value
- Radius shows their presence
- What is the value and presence of next layer capsules?

Layer L+1



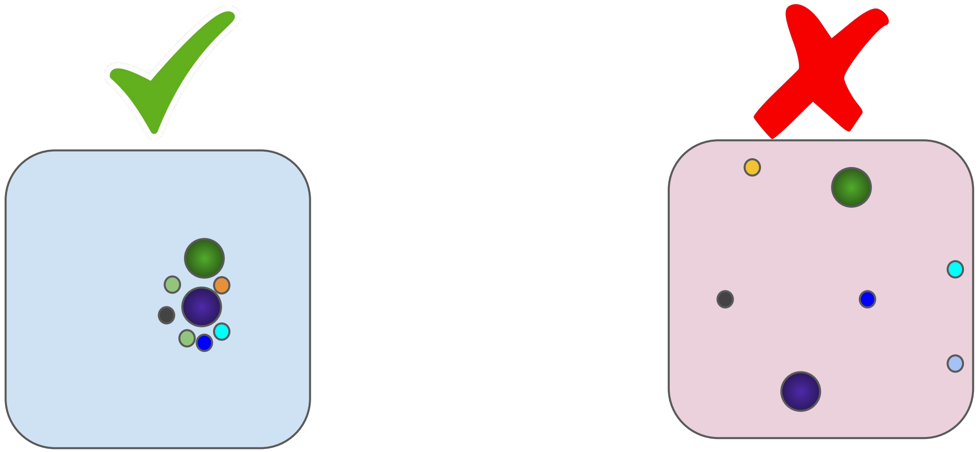
Transform



Is there any Agreement?

Agreement (M step)

Euclidean Distance

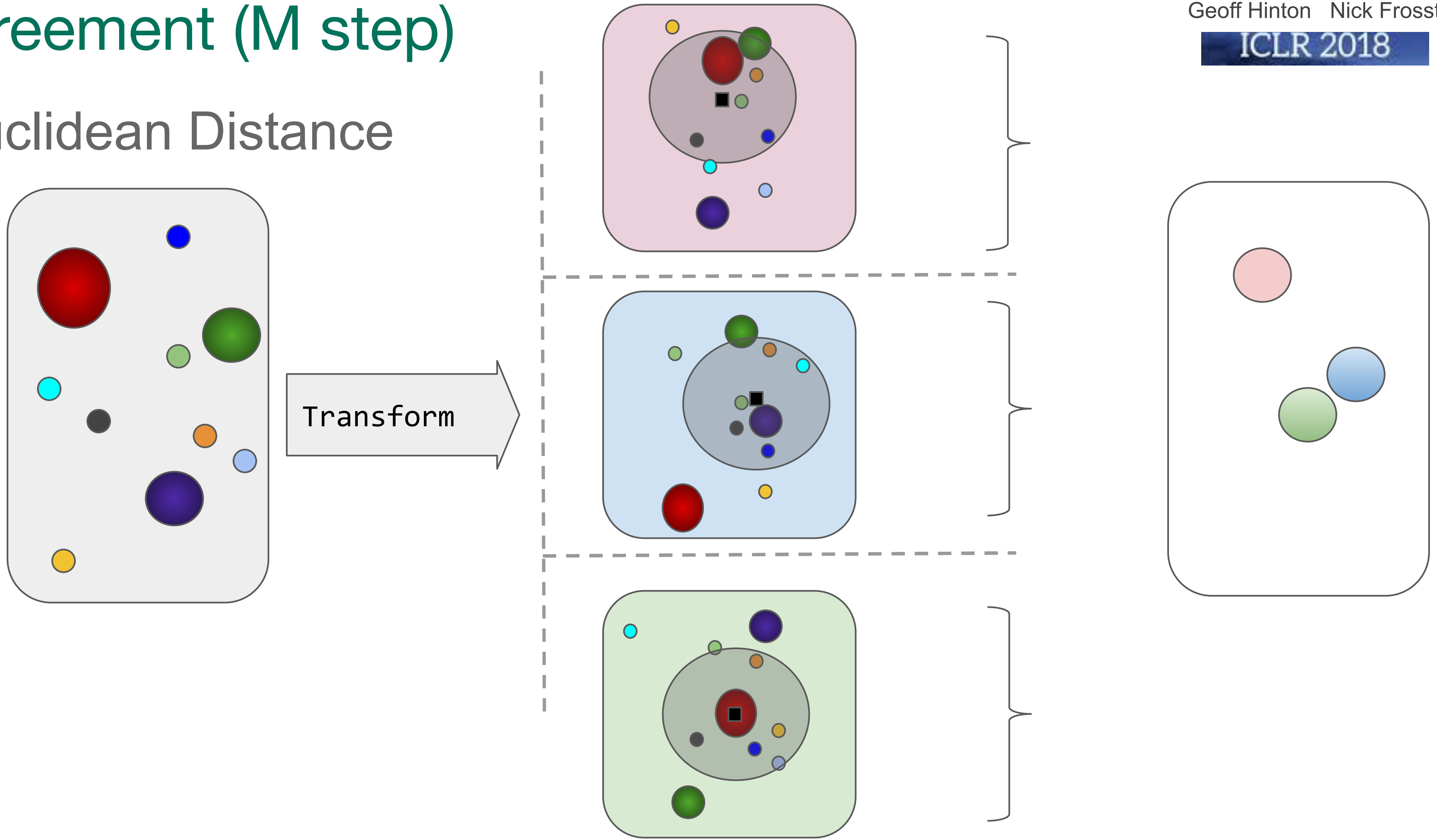


Find the clusters

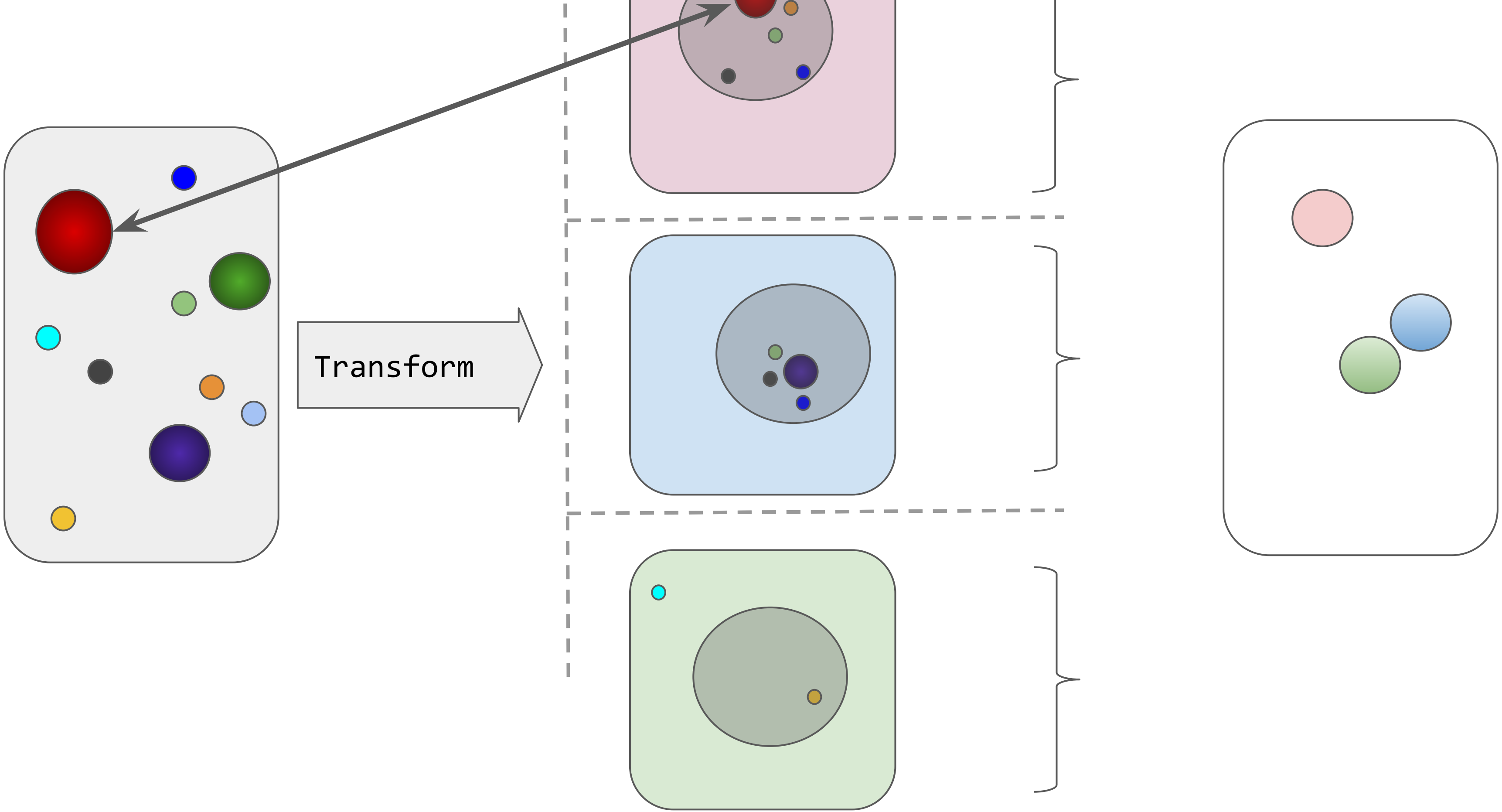
Expectation Maximization for fitting Mixture of Gaussians.

Agreement (M step)

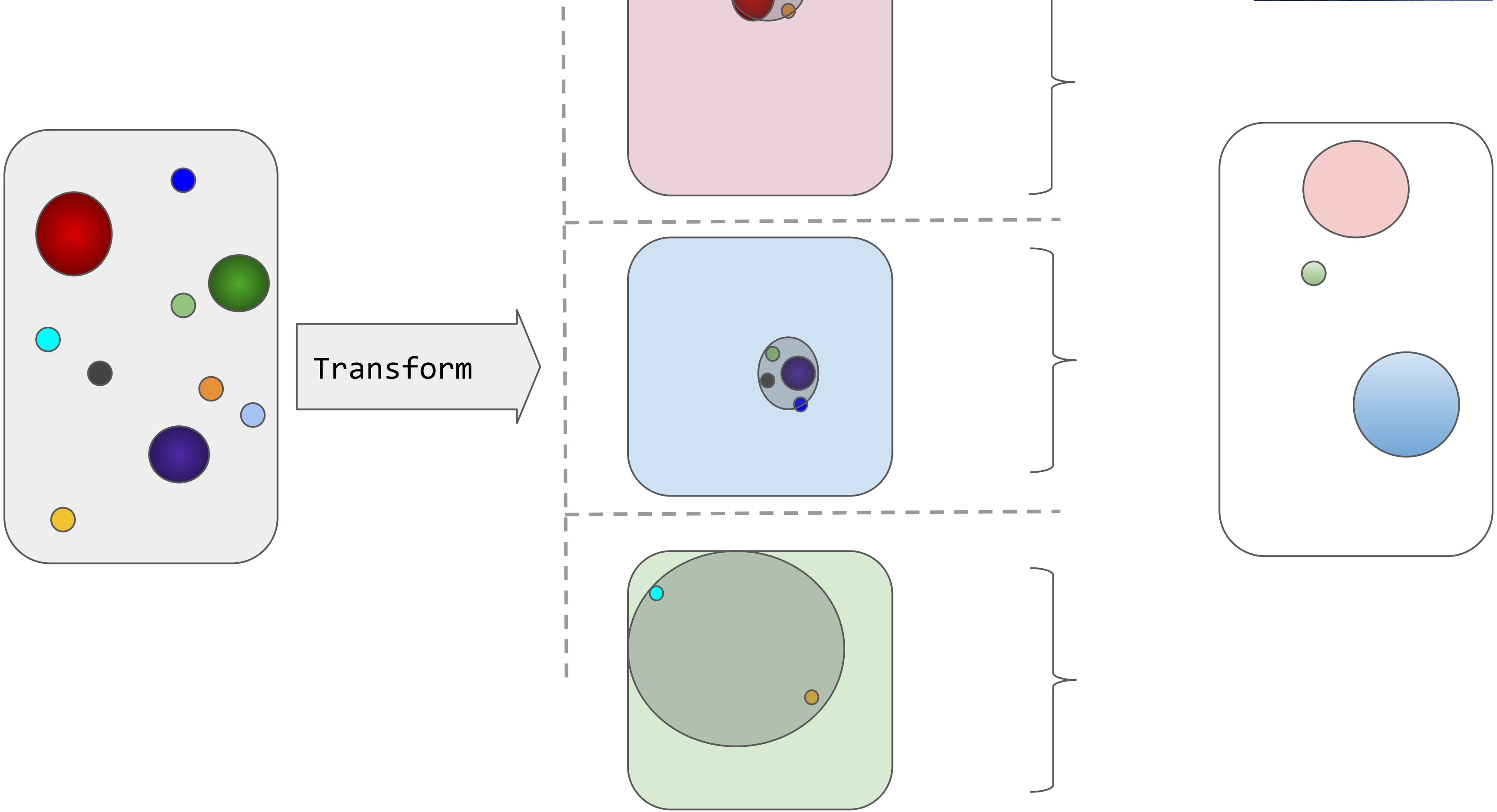
Euclidean Distance



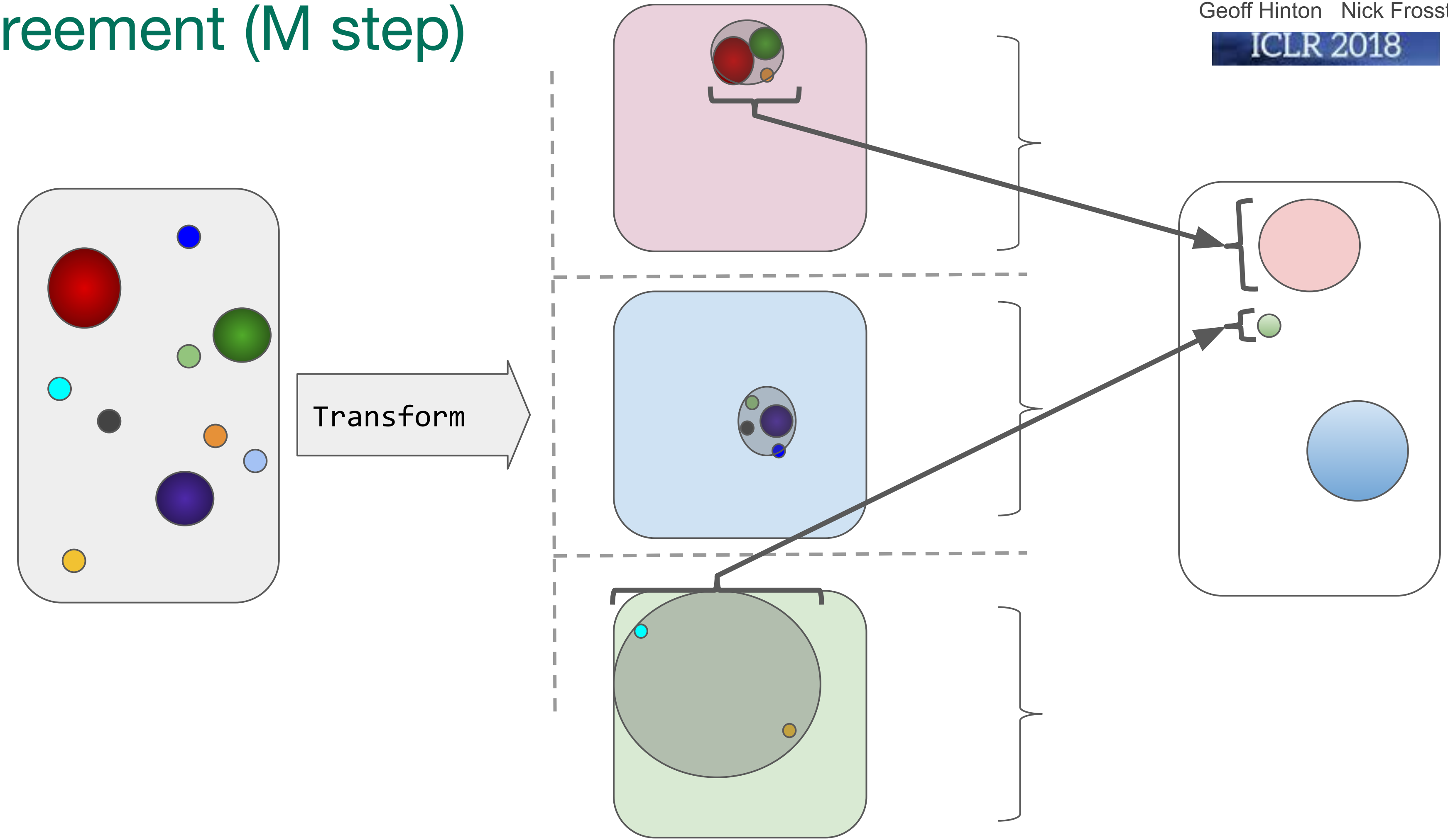
Assignment (E step)



Agreement (M step)



Agreement (M step)

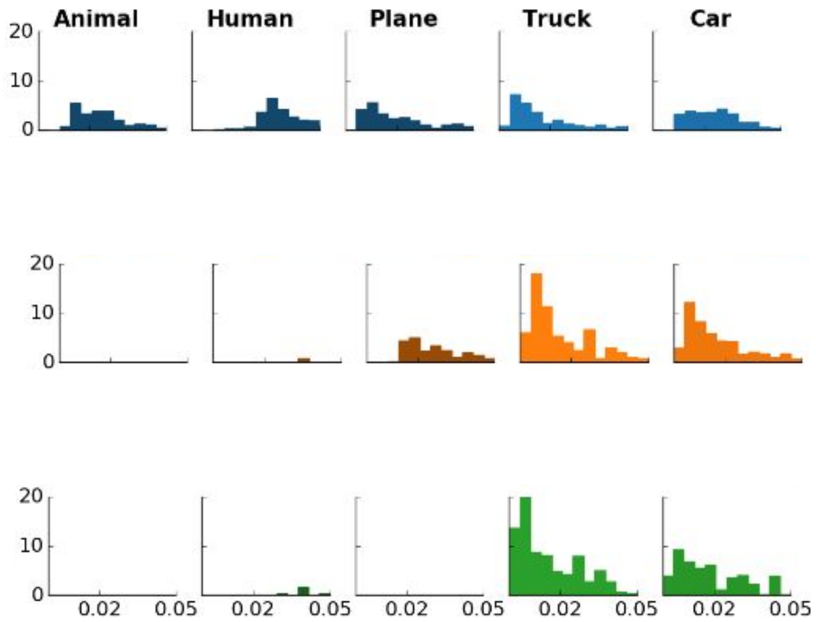


Routing in action

Iteration 1

Iteration 2

Iteration 3



Viewpoint generalization

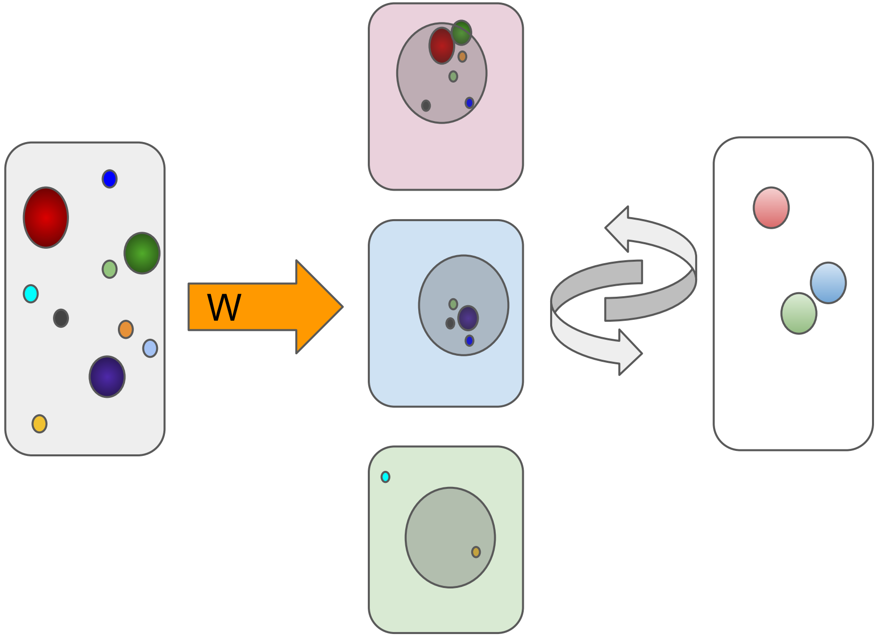
Train

Test

Test error %	CNN	vs	Capsule
Azimuth	20%		**13.5%**
Elevation	17.8%		**12.3%**

Code available at:
https://github.com/google-research/google-research/tree/master/capsule_em

Agreement Finding



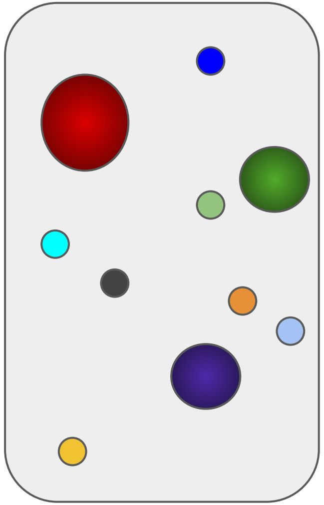
Iterative Routing

- Opt-Caps & SVD-Caps [1, 2]
- G-Caps & SOVNET [3, 4]
  - Explicit group equivarience
- EncapNet [5]
  - Sinkhorn iteration

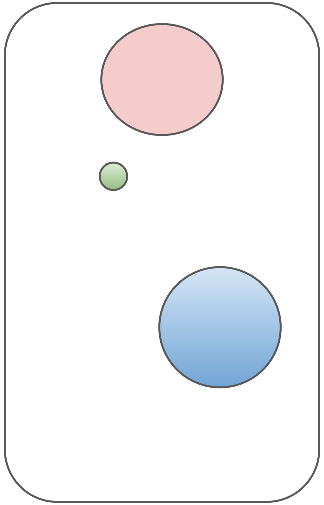
[1]: Dilin Wang and Qiang Liu. An optimization view on dynamic routing between capsules. 2018.
[2]: Mohammad Taha Bahadori. Spectral capsule networks. 2018
[3]: Jan Eric Lenssen, Matthias Fey, and Pascal Libuschewski. Group equivariant capsule networks, NIPS 2018
[4]: Anonymous ICLR 2020 submission.
[5]: Hongyang Li, Xiaoyang Guo, Bo Dai, Wanli Ouyang, and Xiaogang Wang. Neural network encapsulation. ECCV, 2018.

Can we learn a neural network to do the clustering rather than running explicit clustering algorithm?

Learn a cluster finder



Neural Network

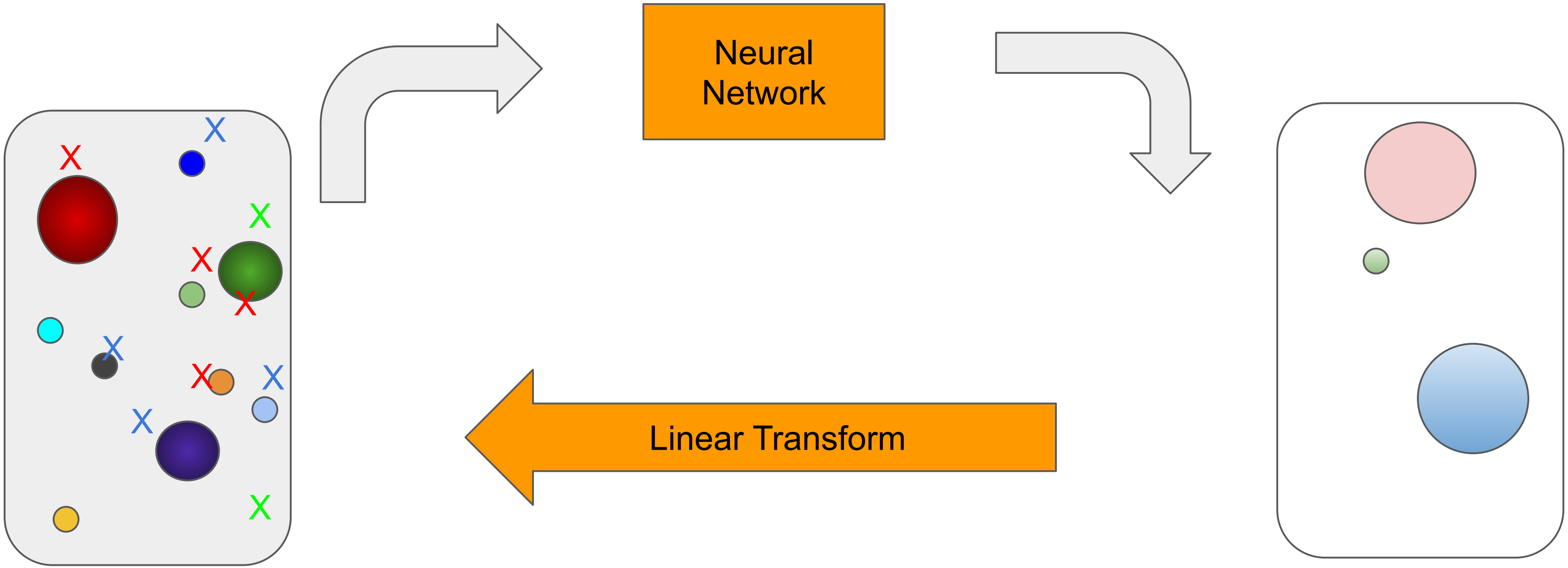


Previously: $$T_h = T_r W_{rh}$$

Now: $$T_h = f(T_r)$$

It should still be true: $$T_r = T_h W_{rh}^{-1}$$

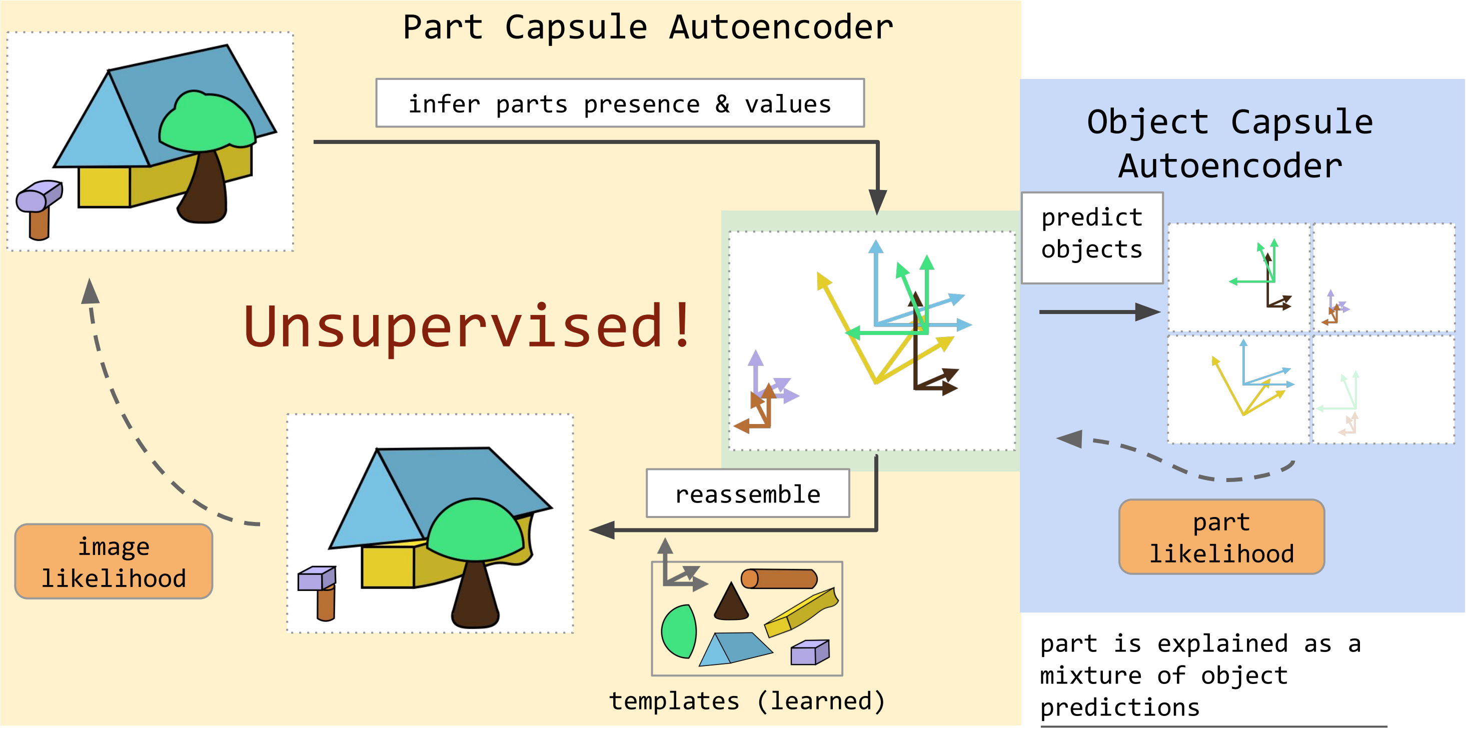
Learn a cluster finder



Optimize mixture model
log-likelihood.

- Each Layer is an autoencoder with a single linear decoder.
- A whole capsule gives predictions for its part capsules.

Stacked Capsule Autoencoder



Adam Kosiorek et al, Neurips 2019.

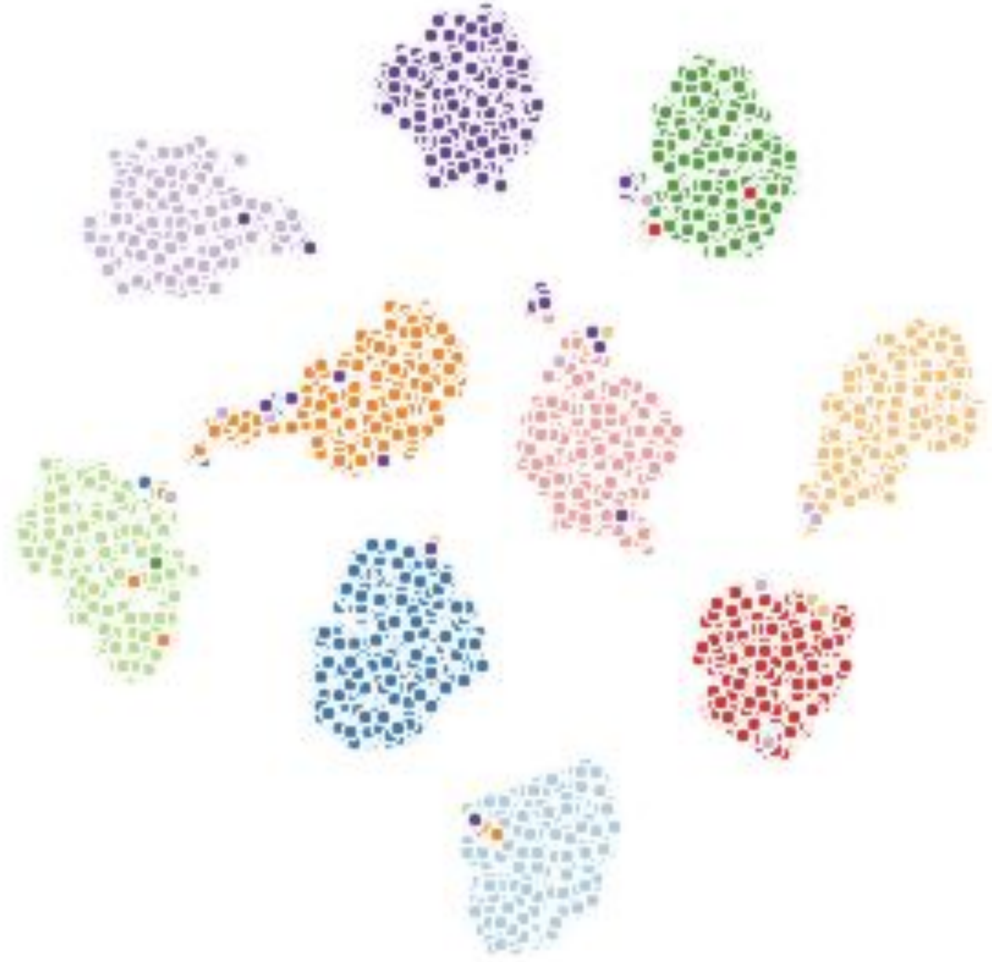
SCAE on MNIST
Unsupervised

Train with 24 object capsules.

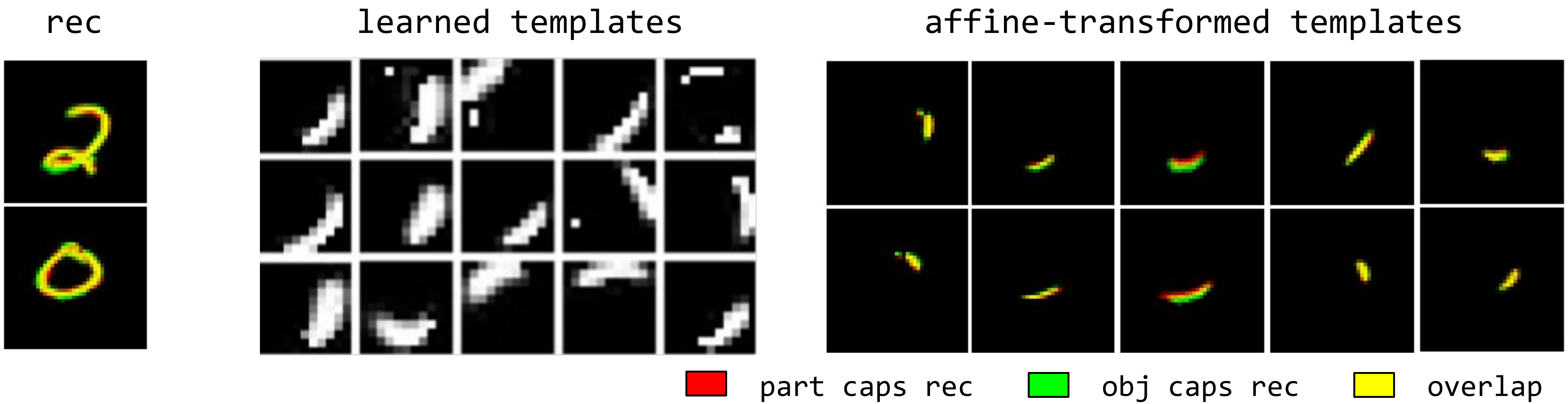
Cluster -> 98.7% Accuracy.

No Image Augmentation.

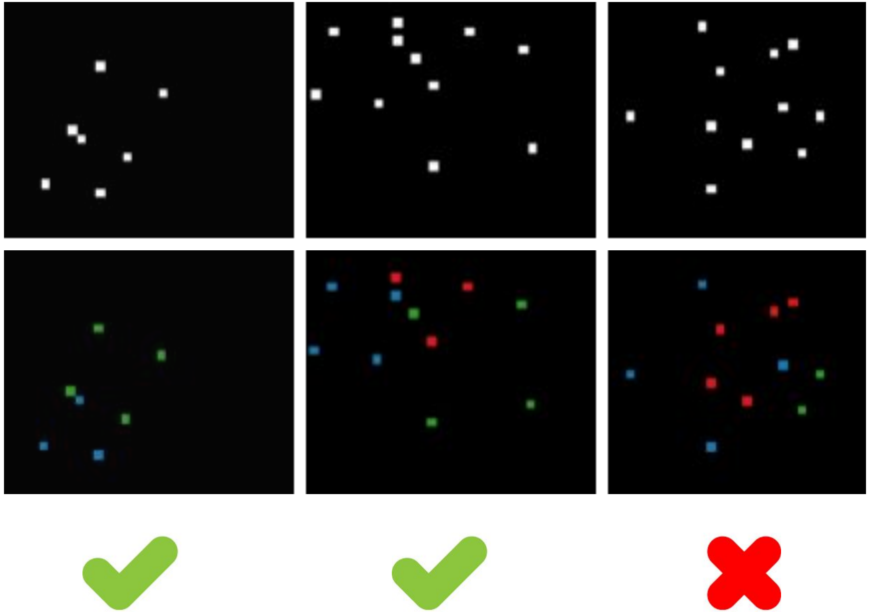
TSNE of Capsule Presences:



MNIST: Part Capsules



Finding Constellations



- Two squares and a triangle
- Patterns might be absent
- Visualizing the mixture model assignments.

Error:

- Best: 2.8%
- Average: 4.0%
- Baseline: 26.0%

Discussion & Future Work

- Introduced Capsule Networks with agreement.
- Capsule Networks can model viewpoint more efficiently.
  - Better viewpoint generalization.
  - Better unsupervised training.
- Future directions
  - The background.
  - The texture.

Questions