

Dynamic Routing Between Capsules

Nov 29, 2018

Paper by: Sara Sabour, Nicholas Frosst, Geoffrey Hinton

Presented by: Patrick Li



UNIVERSITY OF WATERLOO
FACULTY OF MATHEMATICS

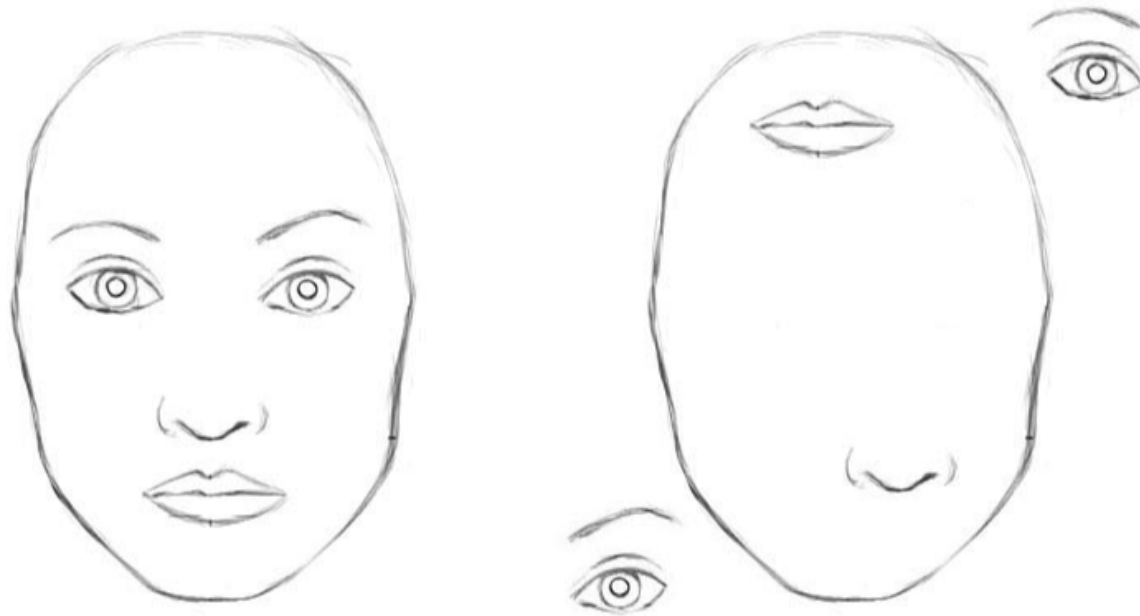
Presentation Architecture

1. Problems with CNNs
2. Motivation and Intuition
3. Network Architecture
 - Capsules
 - Dynamic Routing
4. Experimental Results
5. Critique



Drawbacks of Convolutional Neural Networks

- Translational invariance; no spatial hierarchies between objects
- “The pooling operation used in convolutional neural networks is a big mistake and the fact that it works so well is a disaster.”- Hinton
- Overlapping Segments



Drawbacks of Convolutional Neural Networks

CNN Prediction: Kitten

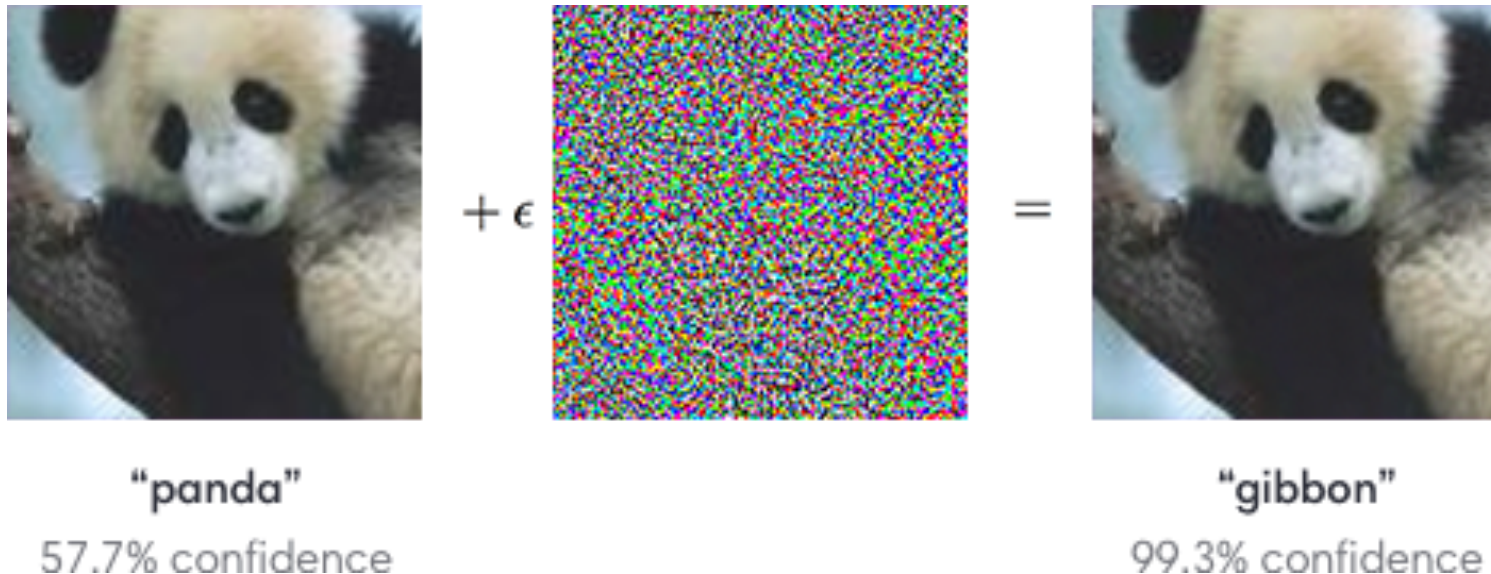


CNN Prediction: Guinea Pig



Adversarial Examples

- Demonstrates the lack of robustness in CNNs
- Security threat, and demonstrates that CNNs may not be learning in the way we desire



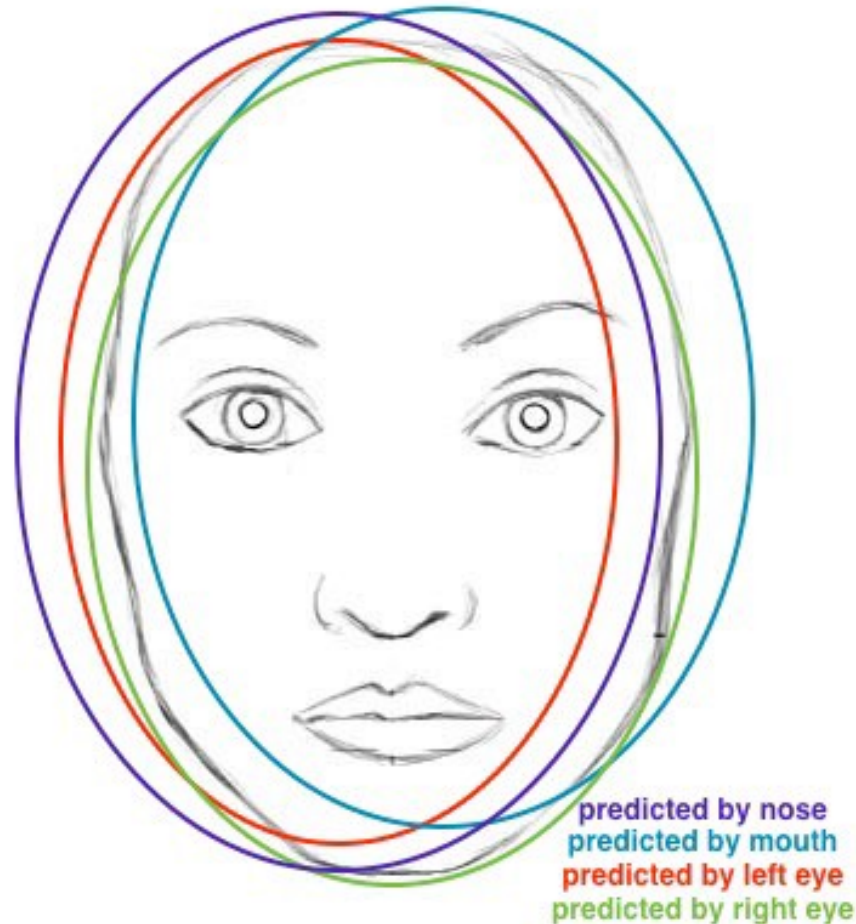
Motivation for CapsuleNets

- Intuition similar to rendering in computer graphics
- Explicit representation of pose relationships to induce rotational invariance



What are Capsules?

- Vector representations of the state of a detected feature
- Low level capsules communicate to infer information of higher level features



Notation

- Squash

$$\mathbf{v}_j = \frac{||\mathbf{s}||^2}{1 + ||\mathbf{s}||^2} \frac{\mathbf{s}_j}{||\mathbf{s}_j||}$$

- Coupling coefficients

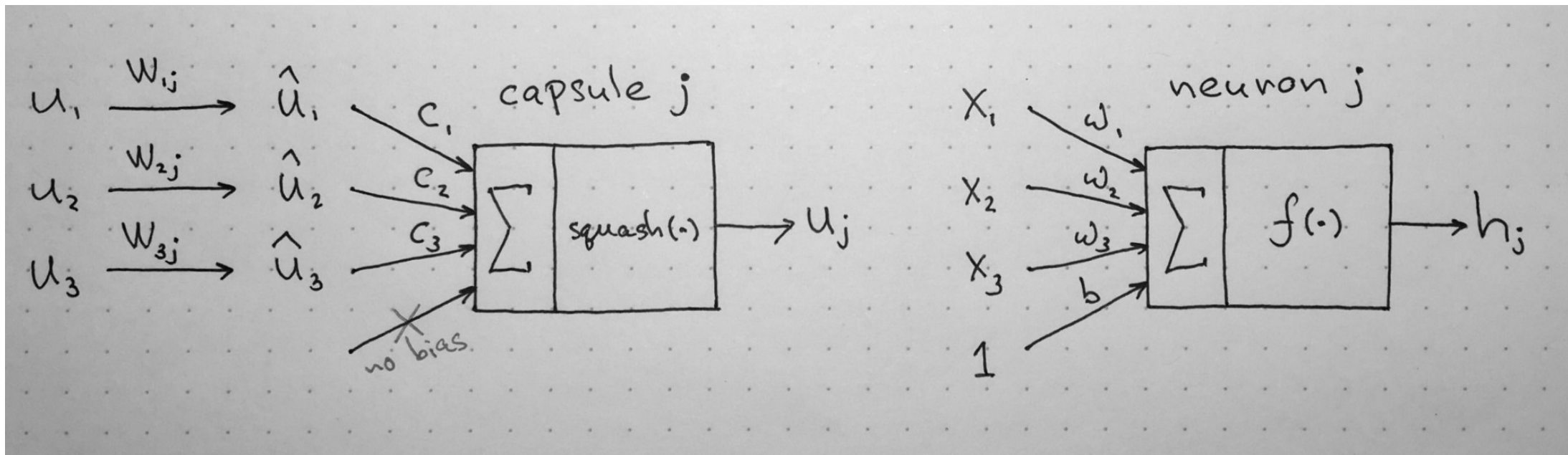
$$c_{ij} = \frac{\exp(b_{ij})}{\sum_k \exp(b_{ik})}$$

- Capsule Operations

$$\mathbf{s}_j = \sum_i c_{ij} \hat{\mathbf{u}}_{j|i}, \hat{\mathbf{u}}_{j|i} = \mathbf{W}_{ij} \mathbf{u}_i$$



Capsules vs. Neurons

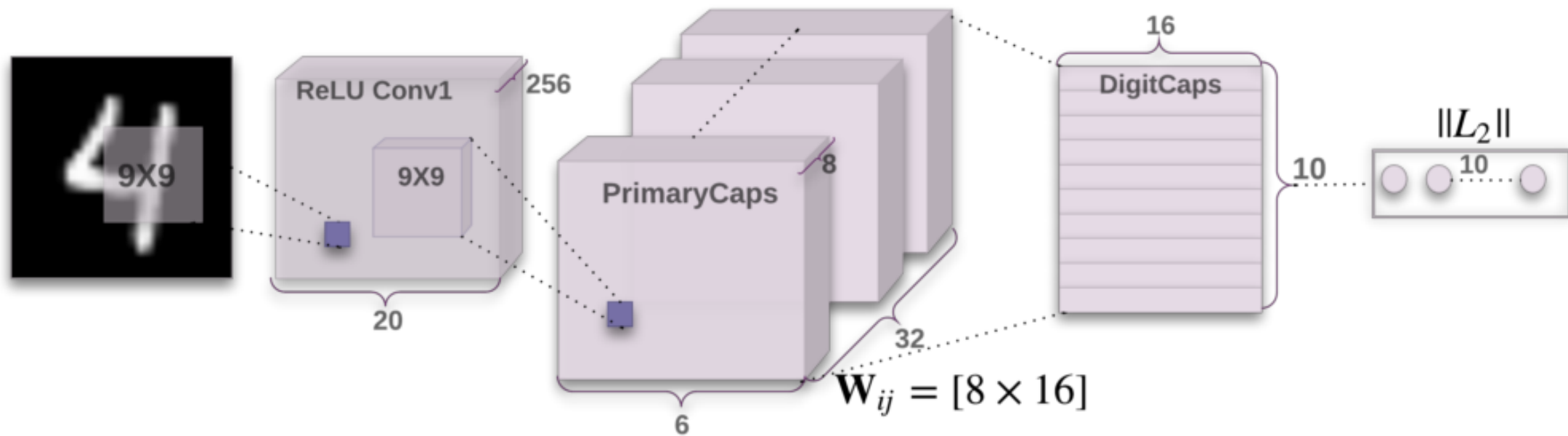


Architecture

1. Encoder
2. Loss Function
3. Decoder
4. Dynamic Routing/Training



Encoder



Loss Function

CapsNet Loss Function

loss term for one DigitCap

calculated for correct DigitCap

calculated for incorrect DigitCaps

$$L_c = T_c \max(0, m^+ - ||\mathbf{v}_c||)^2 + \lambda (1 - T_c) \max(0, ||\mathbf{v}_c|| - m^-)^2$$

1 when correct DigitCap, 0 when incorrect

zero loss when correct prediction with probability greater than 0.9, non-zero otherwise

0.5 constant used for numerical stability

1 when incorrect DigitCap, 0 when correct

zero loss when incorrect prediction with probability less than 0.1, non-zero otherwise

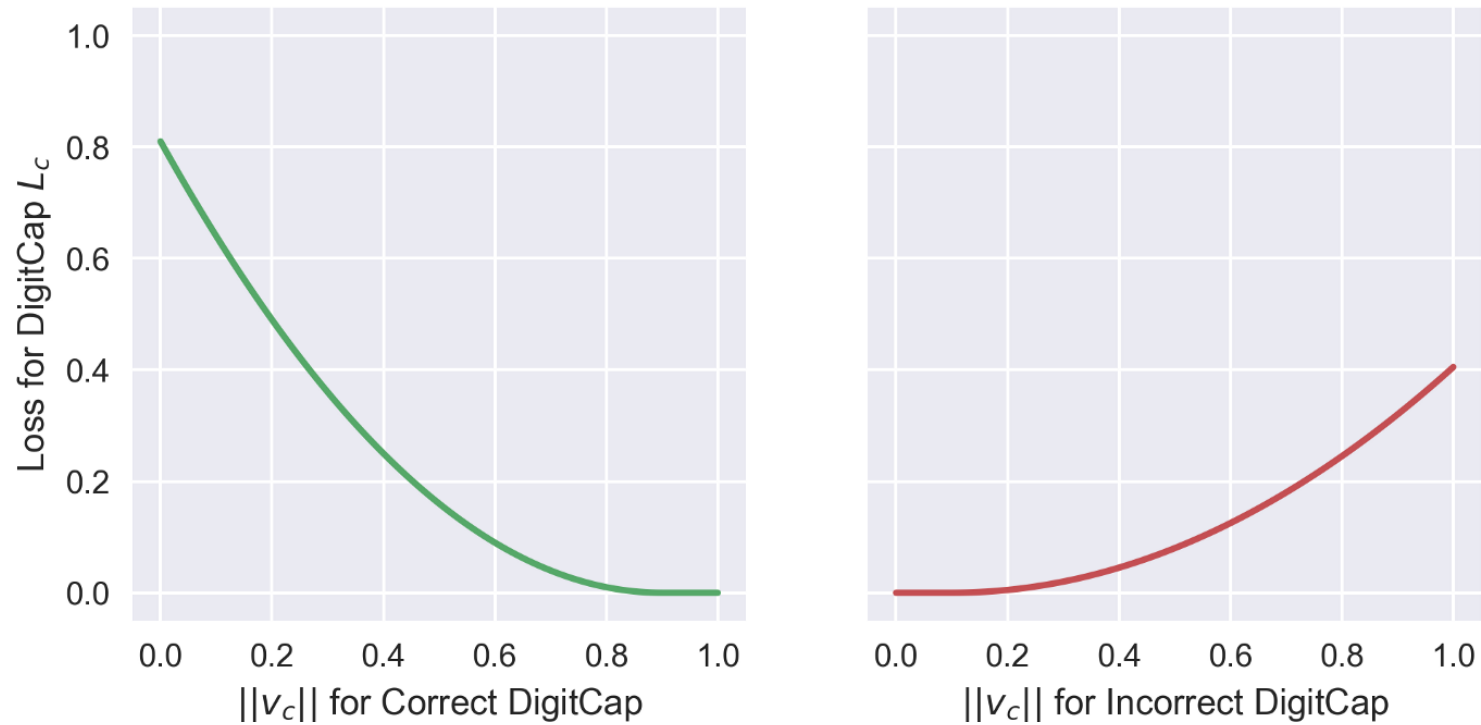
L2 norm

L2 norm

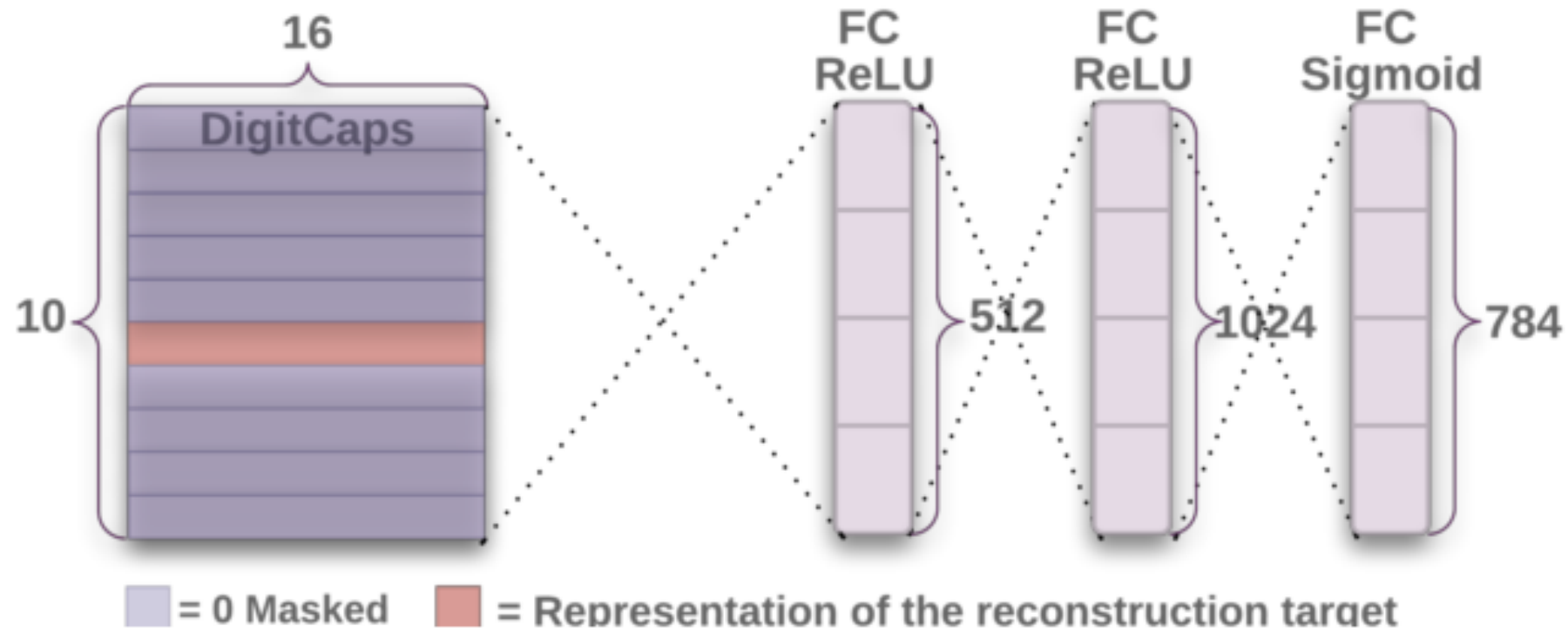
Note: correct DigitCap is one that matches training label, for each training example there will be 1 correct and 9 incorrect DigitCaps

Loss Function

Loss Function Value for Correct and Incorrect DigitCap



Decoder



Decoder

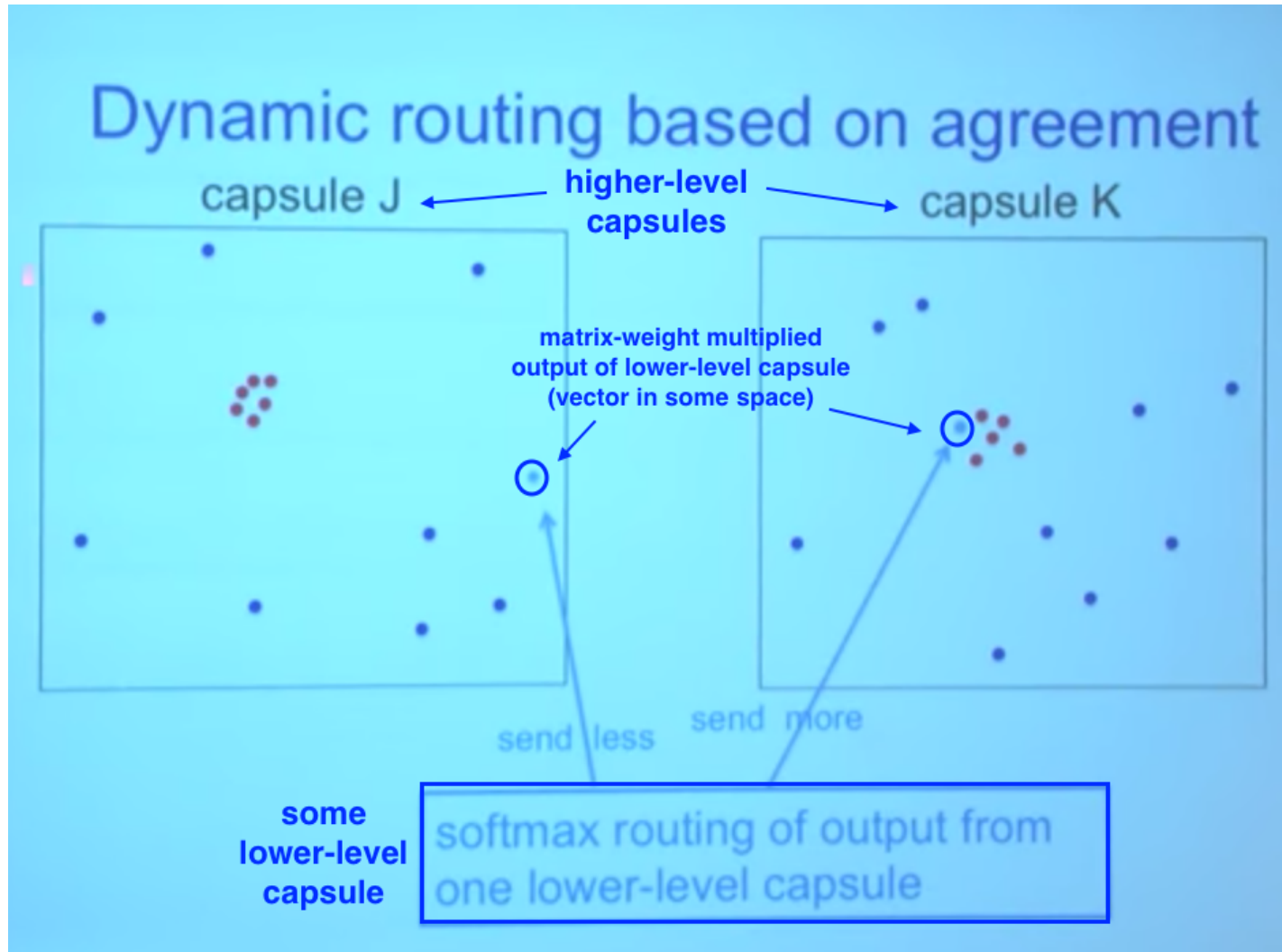
(l, p, r)	$(2, 2, 2)$	$(5, 5, 5)$	$(\tilde{8}, 8, 8)$	$(9, 9, 9)$	$(5, 3, 5)$	$(5, 3, 3)$
Input						
Output						

Dynamic Routing

Procedure 1 Routing algorithm.

```
1: procedure ROUTING( $\hat{\mathbf{u}}_{j|i}, r, l$ )
2:   for all capsule  $i$  in layer  $l$  and capsule  $j$  in layer  $(l + 1)$ :  $b_{ij} \leftarrow 0$ .
3:   for  $r$  iterations do
4:     for all capsule  $i$  in layer  $l$ :  $\mathbf{c}_i \leftarrow \text{softmax}(\mathbf{b}_i)$  ▷ softmax computes Eq. 3
5:     for all capsule  $j$  in layer  $(l + 1)$ :  $\mathbf{s}_j \leftarrow \sum_i c_{ij} \hat{\mathbf{u}}_{j|i}$ 
6:     for all capsule  $j$  in layer  $(l + 1)$ :  $\mathbf{v}_j \leftarrow \text{squash}(\mathbf{s}_j)$  ▷ squash computes Eq. 1
7:     for all capsule  $i$  in layer  $l$  and capsule  $j$  in layer  $(l + 1)$ :  $b_{ij} \leftarrow b_{ij} + \hat{\mathbf{u}}_{j|i} \cdot \mathbf{v}_j$ 
   return  $\mathbf{v}_j$ 
```

Dynamic Routing Intuition



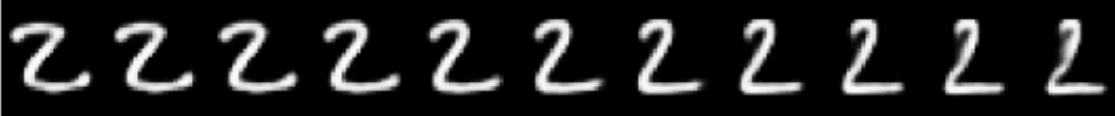
MNIST Experiments

Table 1: CapsNet classification test accuracy. The MNIST average and standard deviation results are reported from 3 trials.

Method	Routing	Reconstruction	MNIST (%)	MultiMNIST (%)
Baseline	-	-	0.39	8.1
CapsNet	1	no	$0.34_{\pm 0.032}$	-
CapsNet	1	yes	$0.29_{\pm 0.011}$	7.5
CapsNet	3	no	$0.35_{\pm 0.036}$	-
CapsNet	3	yes	$0.25_{\pm 0.005}$	5.2

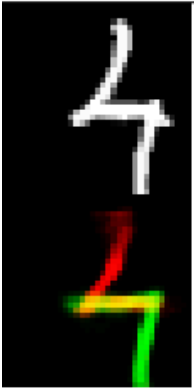
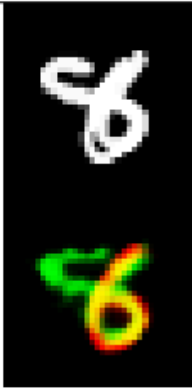
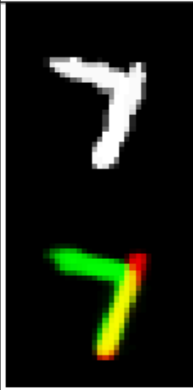
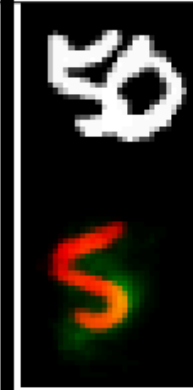
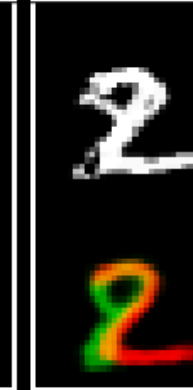
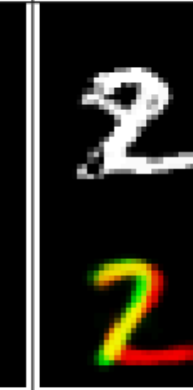
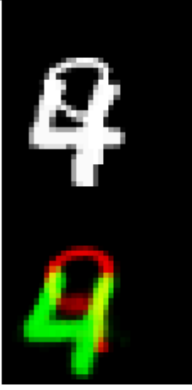
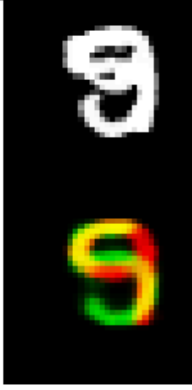
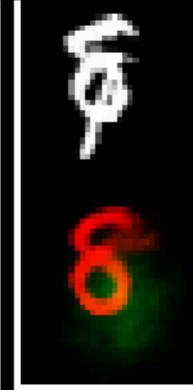
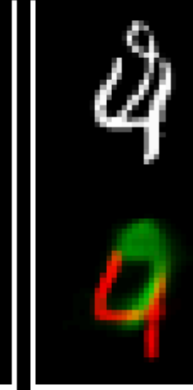
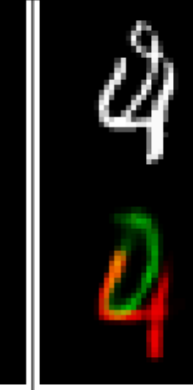


DigiCaps Representations in MNIST

Scale and thickness	
Localized part	
Stroke thickness	
Localized skew	
Width and translation	
Localized part	



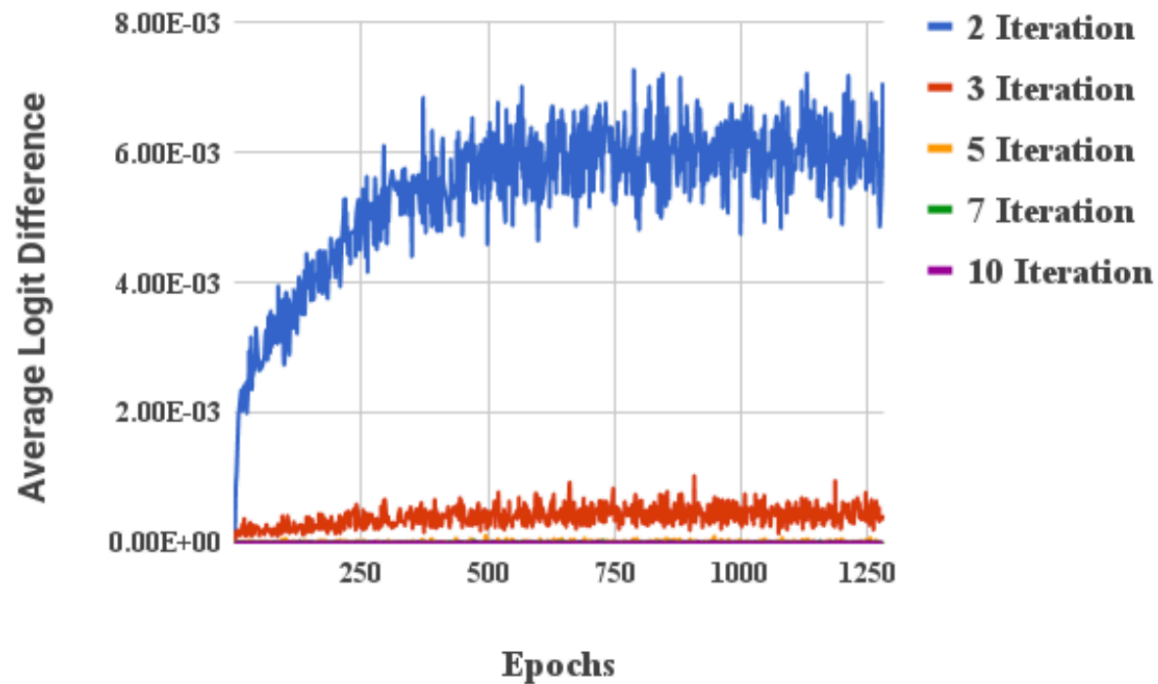
MultIMNIST

R:(2, 7) L:(2, 7)	R:(6, 0) L:(6, 0)	R:(6, 8) L:(6, 8)	R:(7, 1) L:(7, 1)	*R:(5, 7) L:(5, 0)	*R:(2, 3) L:(4, 3)	R:(2, 8) L:(2, 8)	R:P:(2, 7) L:(2, 8)
							
R:(8, 7) L:(8, 7)	R:(9, 4) L:(9, 4)	R:(9, 5) L:(9, 5)	R:(8, 4) L:(8, 4)	*R:(0, 8) L:(1, 8)	*R:(1, 6) L:(7, 6)	R:(4, 9) L:(4, 9)	R:P:(4, 0) L:(4, 9)
							

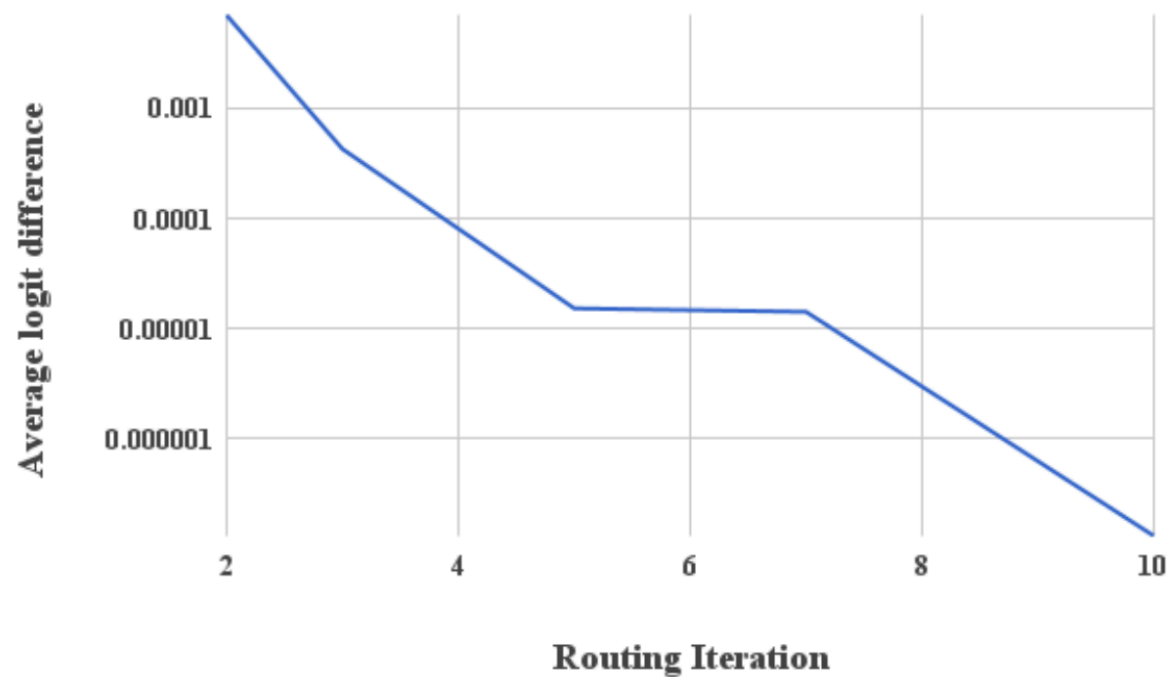


Number of Routing Iterations

(a) During training.

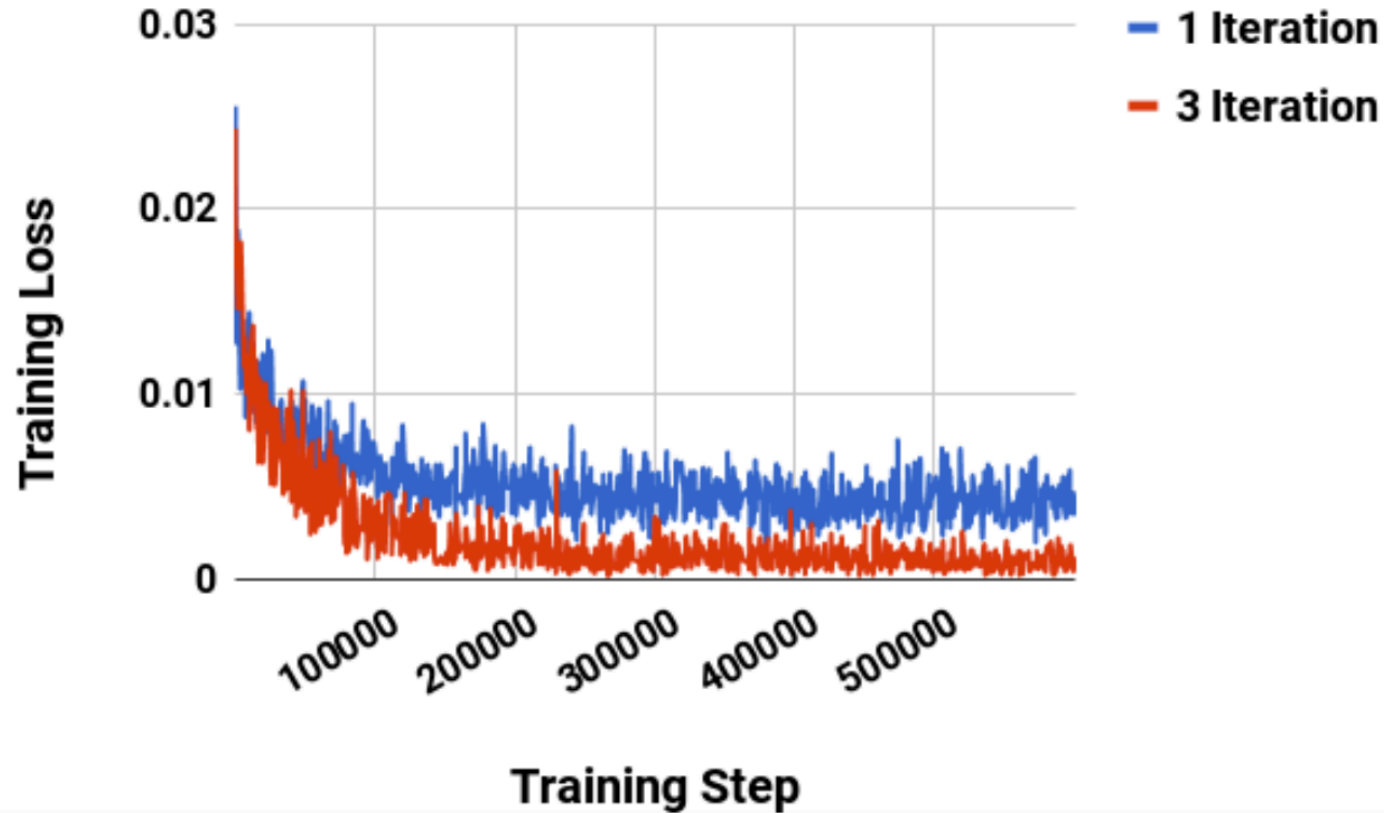


(b) Log scale of final differences.



Number of Routing Iterations

Overfitting Observed in CIFAR 10 Experiment



Critique

- Novel Approach, will lead to a new wave of research
- Resolves shortcomings of CNNs but currently underwhelming
- May require more domain knowledge of specific problems unlike CNNs
- Incredibly slow training for number of parameters (exponential number of iterations for routing training as number of capsule layers increase)
- Conceptually good for “crowded” images and segmentation

