Ladder Capsule Network

Taewon Joeng, Youngmin Lee, Heeyoung Kim
Industrial Statistics Lab, KAIST

2019. 06. 13. ICML

Capsule Network

❖ Components of Capsule Network (CapsNet)

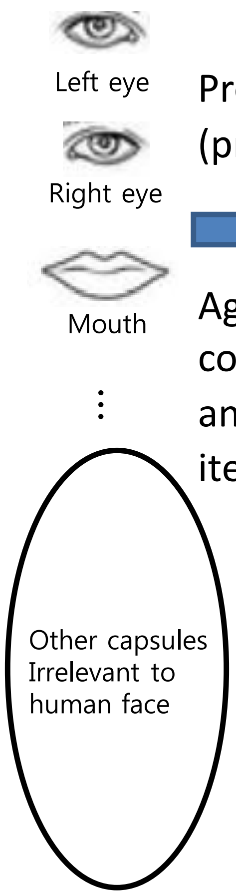
- Basically, the architecture of CapsNet is similar to other feed-forward networks, but uses capsules instead of neurons.

- The output of capsule represents 2 properties of an entity: 
  (1) pose (2) probability of existence (activation), which have the forms of a vector and a logistic unit, respectively. Capsules make it possible to learn an equivariant representation.

- Capsules in higher-level and lower-level layers stand for a whole (e.g. human face) and parts (e.g. left and right eyes, mouth), respectively. To capture the “part-whole” relationship, CapsNet uses the dynamic routing algorithm as an “agreement rule”.

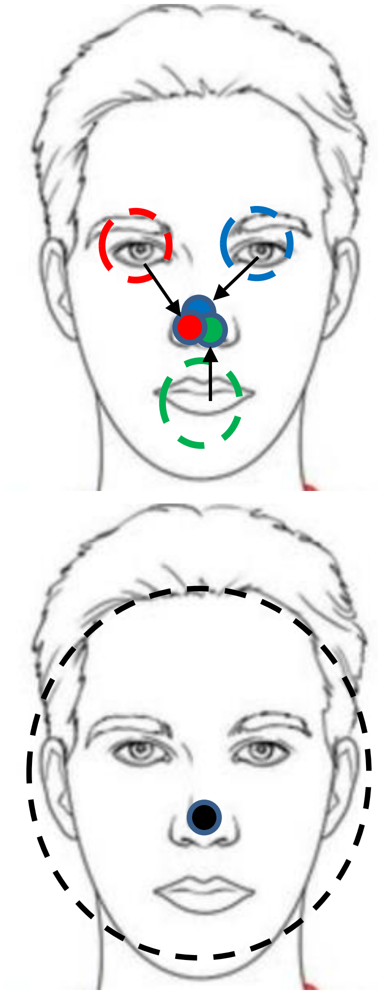
Capsule Network

Dynamic Routing Algorithm



Predictions $\hat{u}_{j|i} = W_{ij} u_i$
(prediction matrix W_{ij})

Agreement: update coupling coefficients c_{ij} and higher-level capsule v_j iteratively



Algorithm 1 Dynamic routing algorithm (Sabour et al., 2017)

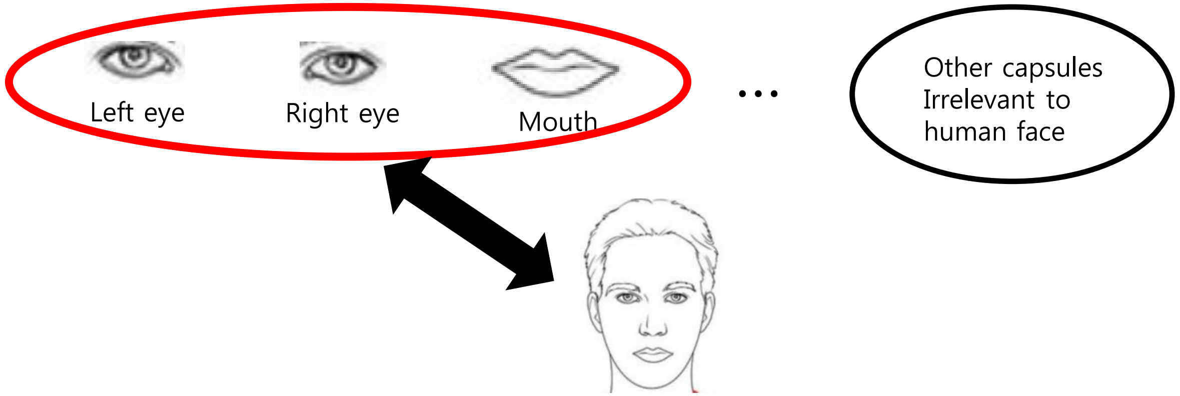
Initialize logit parameters $b_{ij} = 0$ for all capsule i in layer l and capsule j in layer $(l+1)$.

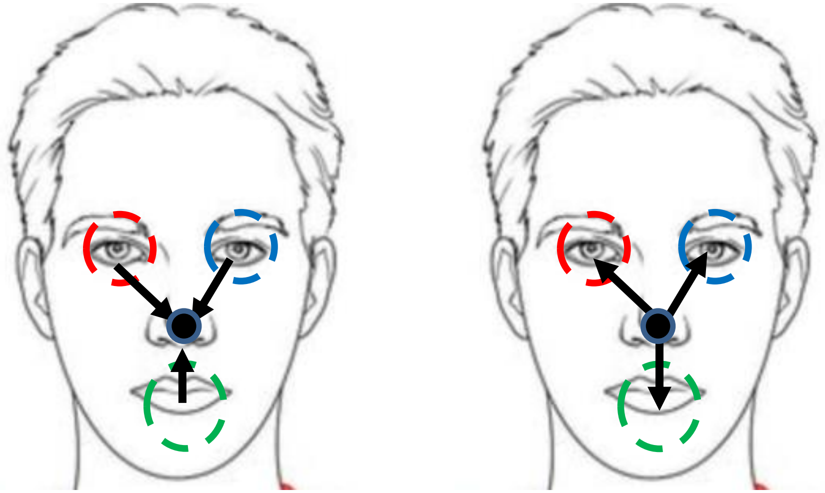
1: **for** 1:$MaxIter$ **do**
2: $\quad c_{ij} = \frac{\exp(b_{ij})}{\sum_k \exp(b_{ik})}$ for all capsule i in layer l.
3: $\quad s_j = \sum_i c_{ij} \hat{u}_{j|i}$ and $v_j = \frac{||s_j||^2}{1+||s_j||^2} \frac{s_j}{||s_j||}$ for all capsule j in layer $(l+1)$.
4: $\quad b_{ij} = b_{ij} + \langle \hat{u}_{j|i}, v_j \rangle$ for all capsule i in layer l and capsule j in layer $(l+1)$.
5: **end for**

Ladder Capsule Network

❖ New Ideas

- Dynamic routing algorithm computes the prediction of higher-level capsule from all of lower level capsules, even though some of them are irrelevant.
- Direction of agreement rule: Instead of the agreement of prediction from lower level to higher level, regression from higher level to lower level could be used for agreement rule.



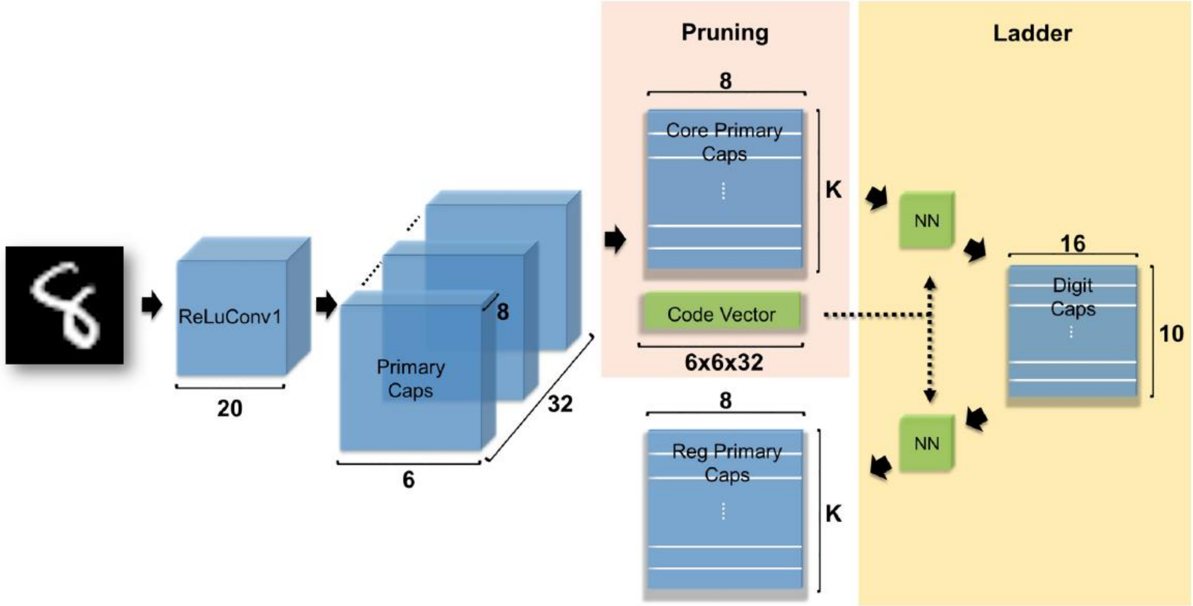


From lower to higher level(DR)

From higher to lower level

Ladder Capsule Network

❖ Pruning & Ladder Layers



- Pruning layer: selects the K relevant capsules among all lower-level capsules, and propagates them to the layer above. Code vector, which is 0-1 encoded to indicate which capsules are selected, is also propagated to the layer above.
- Ladder layer: constructs higher-level capsule, and regresses the K selected lower level capsules from higher-level capsule.

Ladder Capsule Network

❖ Pruning Layer

- Given a pre-fixed number K, choose the K most active capsules among all lower level capsules (i.e. pruning the capsules based on the activation)

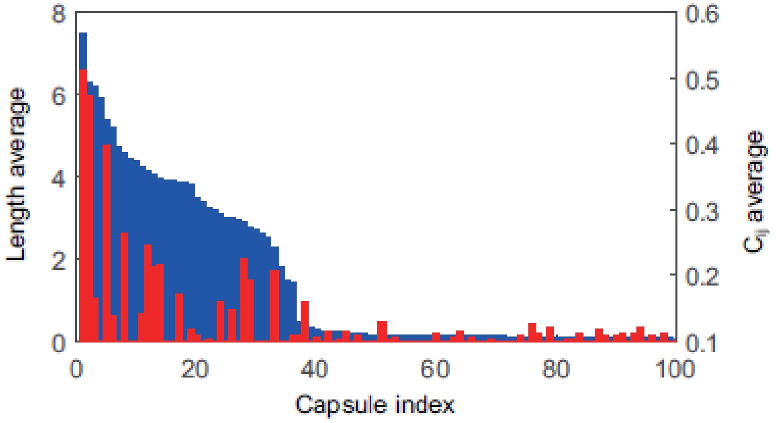
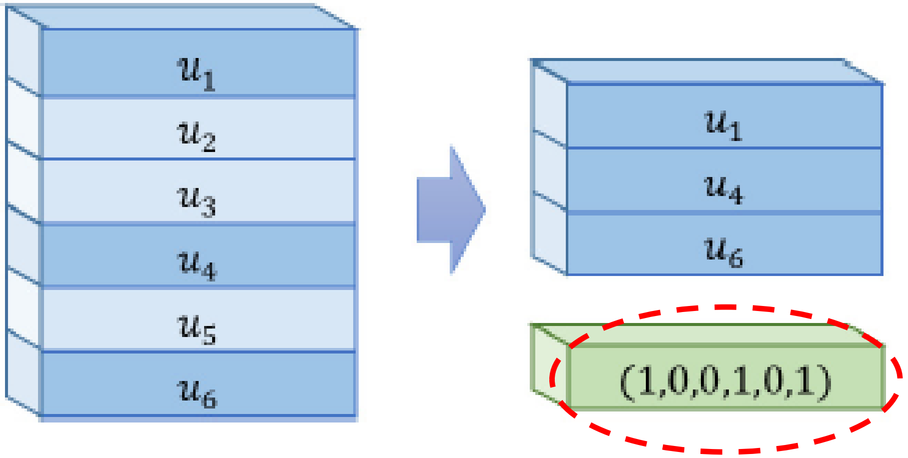


Figure 1: Length of u_i (left y-axis, blue bar) and value of c_{ij} (right y-axis, red bar) of the 100 most active lower-level capsules to predict the "0" digit capsule.

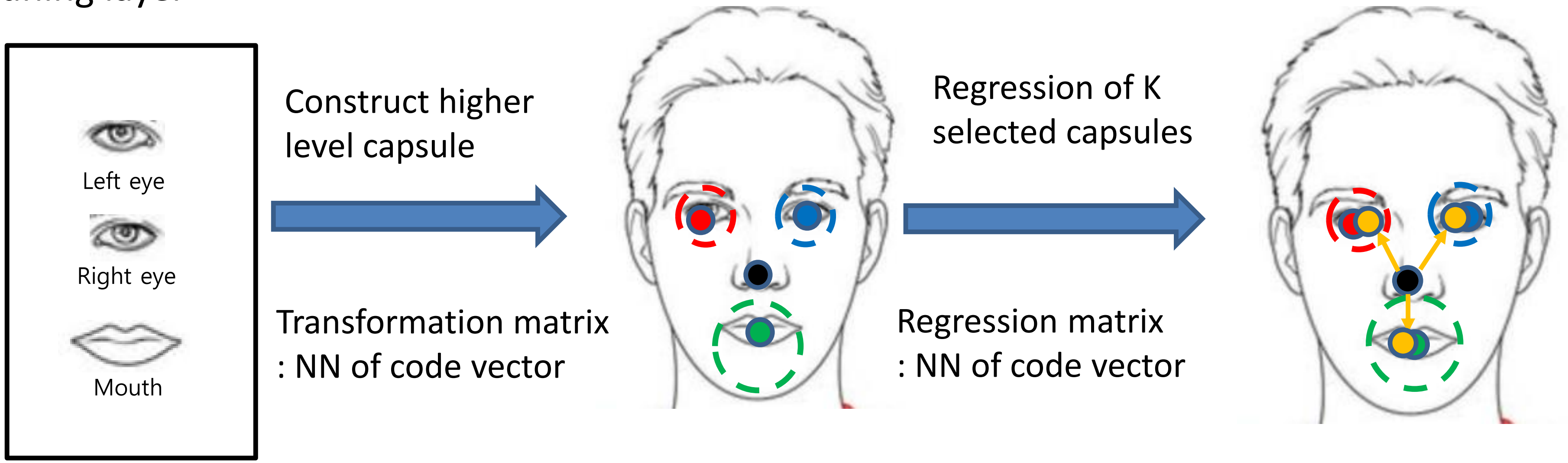


Code vector, which contains the information about which capsules are selected.

Ladder Capsule Network

❖ Ladder Layer

K-selected capsules from
pruning layer



- Unlike the dynamic routing algorithm, the ladder layer does not require several iterations to compute the agreement, thus reduces the computing cost.

Ladder Capsule Network

❖ Experiments

Method	K	MNIST(%)	affNIST(%)
CNN (Sabour et al., 2017)	-	0.39	34.0
CapsNet (Sabour et al., 2017)	-	0.25	21.0
L-CapsNet (9×9 kernel)	50	0.74	13.0
L-CapsNet (9×9 kernel)	70	0.50	12.5
L-CapsNet (9×9 kernel)	100	0.80	13.2
L-CapsNet (15×15 kernel)	50	0.69	12.5
L-CapsNet (15×15 kernel)	70	0.73	12.2
L-CapsNet (15×15 kernel)	100	0.79	13.1

Method	Computation time, in seconds
L-CapsNet ($K = 50$)	0.2034 (0.010)
L-CapsNet ($K = 70$)	0.2159 (0.008)
L-CapsNet ($K = 100$)	0.2953 (0.001)
CapsNet ($r = 3$)	1.732 (0.026)
CapsNet ($r = 4$)	2.123 (0.041)
CapsNet ($r = 5$)	2.656 (0.085)