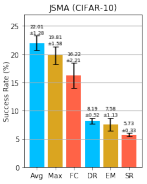
We thank the reviewers for their valuable feedback. The final version will resolve all the concerns raised by the reviews.

[**Reviewer 1**]



Experiments using stronger attacks. During the rebuttal, we experiment the robustness to JSMA attacks with 3 random seeds. As shown in the figure on the right, our approach is the most robust (*i.e.* the lowest attack success rates) to JSMA attacks. We expect that CapsNets would be more robust to gradient-based attack (FGSM, BIM, JSMA) than to optimization-based attack (DeepFool, CW), which will be further discussed in the final draft.

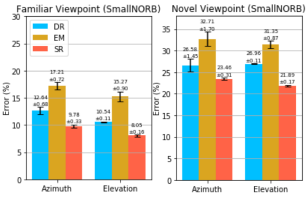
Depth and width of the capsule layers. We fixed the routing iterations for dynamic and EM routing at 3, which yield the best performance in their original papers. In this work, the depth means the number of capsule layers added to the base network, and the width is the number of capsules per layer (the same across all capsule layers). We will clarify this.

[**Reviewer 2**]

Experimental results for ensemble behavior. In our initial experiments, we inspected the effect of agreement-routing by training 3-layer dynamic and EM routing models (Conv→PrimaryCaps→FCCaps) on smallNORB dataset with no viewpoint splits. At test, we then enforced fixed, uniform routing coefficients; as a result, each capsule does not route to its best options anymore but uniformly spreads its influence to all upper-level capsules. Surprisingly, this enforcement barely hurt the performance (DR: 91.48% to 90.76%, EM:86.52% to 85.70%). It partly hints that the effect of agreement-routing could be questionable but the ensemble behavior might be sufficient.

Step-by-step explanation on section 4.2. We here summarize how self-routing works step-by-step: (1) Each capsule multiplies its pose with W_{pose} and W_{route} (learnable parameters) to compute the predicted pose changes and the routing coefficients for the upper-level capsules, respectively. (2) Each capsule multiplies its activation output with the routing coefficients to calculate the voting weights to the upper-level capsules. (3) We obtain the poses of upper-level capsules by weighted-averaging the predicted pose changes of all lower-level capsules with their voting weights. (4) Finally, we compute the activations of upper-level capsules by weighted-averaging the routing coefficients of all lower-level capsules with their activations. In self-routing, if a capsule underperforms on some input data (*e.g.* ambiguous cases), the optimizer learns to assign less weight to the capsule as experts are trained in the MoE setting. This process is much easier in our method because they are directly supervised, whereas the effects of lower-level capsule activations to upper-level ones are indirect in previous agreement-based methods.

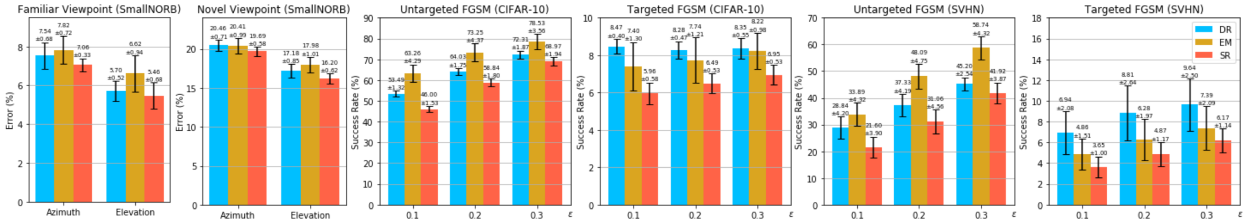
Why CNN baseline. As previous CapsNets employed shallow and weak CNN base networks, we were curious whether the CapsNets can hold any advantage over recent deep networks (with residual connections). In our preliminary experiments, ResNet-20 outperformed the CapsNets even in viewpoint generalization tasks (smallNORB, affNIST) where CapsNets were supposed to be stronger. Thus, we focused on verifying that employing capsule structures could benefit the recent CNNs. It was our anecdotal motivation to employ ResNet as our base network. In the next question, we present the experimental results with no base network as R2 suggested.



Results with no CNN base. During the rebuttal, we perform experiments on smallNORB viewpoint generalization tasks using a 4-layer architecture that consists of a convolution layer followed by 3 consecutive capsule layers (primary, convolutional and fully-connected). The primary and convolutional capsule layers consist of 16 capsules each and each capsule contains 16 neurons. The results are shown in the figures on the right. Each value denotes the mean of the error with 3 random seeds. Our self-routing (SR) outperforms Dynamic and EM routing with significant margins in both tasks. It shows that using shallow feature extractors, the previous routing techniques struggle to learn good representations.

[**Reviewer 3**]

Error bars in the results. We will add error bars to all the results in the final draft. As examples, we below show the mean and standard deviations of the performance with 10 random seeds for the experiments of adversarial attacks and novel viewpoints. Compared to other agreement-based routing CapsNets, our SR approach attains not only much lower errors but also lower or similar variances. All hyperparameters are set to be the same as in the original paper (*e.g.* we use 16 capsules per layer for smallNORB and 32 for CIFAR-10 and SVHN, and fix the depth as 1).



Tables are too dense. We will replace some dense tables with graphs for better readability and visuality.