We appreciate the reviewers' support for the novelty, remarkable performance, and the impact of our work. We thank the reviewers for their valuable feedback and thoughtful reviews. Below we address the concerns raised by the reviewers.

Reviewer 1: It would be great to compare inference time and memory consumption against other defense methods.

Reply: Our defense method has negligible overhead in inference time and memory compared with inference by the standard ResNet without defense, and this is also the case for the PGD adversarial training (arXiv:1706.06083) and TRADES (arXiv:1901.08573). For CIFAR10, the inference time for ResNet20 and En_1ResNet20, averaged over 100 runs, with batch size 1K on a Titan Xp are 1.6941s and 1.6943s, resp. The corresponding peak memory is 4807MB for both ResNet20 and En_1ResNet20. We will report the inference time and memory in the revised manuscript.

Reviewer 1: In Fig. 4, it is unclear why neural networks made wrong predictions. It could be due to the difficulty of the input example or adversarial attack. To make it more clear, authors need to add prediction results for the clean input.

Reply: We will add the classification results for the clean input in the revised manuscript.

Reviewer 1: I do not see experiments that can support claim that this method is a complement to existing defenses.

Reply: Directly replacing the cross-entropy loss with the TRADES loss can further improve the robust accuracy, under the IFGSM^{20} attack, of the WideResNet (arXiv:1901.08573) by $\sim 0.9\%$. We will report more results in the revision.

Reviewer 1: The method is inherently limited to ResNet based network architectures.

Reply: The EnResNet is motivated from the Euler-Maruyama discretization of the Itô process below Eq. (10). Other numerical discretization may motivate ensemble of new network architectures like neural ODE as our future work.

Reviewer 3: The authors are probably not aware of some similar papers like [1,2,3].

Reply: We will discuss these papers in the related work Sec. in the revision.

Reviewer 3: The proposed PDE formalism and the resulted method does not depend on the training objective function. Therefore, Theorem 1 should provide some degree of adversarial robustness even in the absence of adversarial training. Still, empirical results regarding the performance of the proposed method with natural training are missing.

Reply: For natural training, En_1ResNet20 and En_2ResNet20 (with injected Gaussian noise of standard deviation 0.1) has accuracy 27.93% and 28.75%, resp., under the FGSM attack with $\epsilon = 8/255$, in contrast to ResNet20 with robust accuracy 10.45% under the same attack. Moreover, as shown in Sec. 3.3, the naturally trained En_nResNet20 has slightly better robust accuracy than ResNet20 under the IFGSM^{20} attack. Adversarial training enables En_nResNets to have remarkably better robust accuracy than ResNets under the IFGSM^{20} attack. We will make this clear in the revision.

Reviewer 3: I wonder if the ResNets are in a single ensemble sharing weights. If not, the implementation is not consistent with the proposed PDE formalism. If yes, I would like to make sure that the share-weights ensemble (SWE) of ResNets truly outperforms standard ResNet on natural examples.

Reply: In our paper, EnResNets do not share weights. Direct ResNet ensemble counterpart of the PDE formalism needs to share weights. Table 1 below shows that, SWE also improves both natural and robust accuracies which verifies the efficacy of our PDE formalism. Moreover, to further improve the ensemble model's performance, we generalize SWE to non-share weights ensemble (NSWE) with the consideration of increasing the model capacity. As shown in Table 1 in our paper (En_2ResNet20 v.s. ResNet44), NSWE remarkably outperforms the vanilla ResNets with a similar capacity. We will point out this and include the results of SWE in the revision.

Table 1: Accuracy of the robustly trained $n\times\text{En}_1$ResNet20 which denotes the ensemble of n share-weights En_1ResNet20.

	ResNet20	$1\times$ En_1ResNet20	$2\times$ En_1ResNet20	$5\times$ En_1ResNet20
$\mathcal{A}_{\text{nat}}$	75.11%	77.21%	77.88%	77.99%
$\mathcal{A}_{\text{rob}}$ (IFGSM^{20})	46.03%	49.06%	49.17%	49.20%

Reviewer 3: Do the authors run the experiments for multiple times observe consistent gain over the baseline?

Reply: The reported accuracies are averaged over five runs, and the standard deviation is less than 0.5% among these runs. The accuracy of EnResNets is consistently better than the baseline over different runs.

Reviewers 3 & 4: Influence of ϵ for adversarial perturbation, and the standard deviation of the Gaussian noise.

Reply: We take the most-used ϵ, and different ϵ/noise will be discussed in the revision. Table 2 lists $\mathcal{A}_{\text{rob}}$ (IFGSM^{20}) of the robustly trained En_2ResNet20 with different noise. 0.1 gives a good trade-off between accuracy and variance.

Table 2: Robust accuracy of the PGD adversarially trained En_2ResNet20 with different Gaussian noise injection. (five runs)

Standard derivation (Gaussian noise)	0.05	0.1	0.4	0.8
$\mathcal{A}_{\text{rob}}$ (IFGSM^{20})	$50.05\% \pm 0.27\%$	$50.06\% \pm 0.35\%$	$50.51\% \pm 0.90\%$	$43.51\% \pm 3.78\%$

Reviewers 3 & 4: 1). Specify the details of ResNets. 2). Reorganize Sec.3.3. 3). Provide derivation of Eq. (10). 4). Line 126, better symbols for noise injected ResNet might be Resnet* etc. 5). Add the meanings of the symbols in Th 1.

Reply: We will modify our manuscript to include all these valuable suggestions.

Reviewer 4: The highest vanilla resnet accuracy in the paper is close to 85%, however, there are many implementations of resnets which achieve $\sim 95\%$ on CIFAR10. Why aren't the baseline numbers reported in the paper high enough?

Reply: The reported accuracies are that of the robustly trained models by solving the EARM (Eq.(1)). The proposed ResNets ensemble can also improve the natural accuracy of the naturally trained models, e.g., $\mathcal{A}_{\text{nat}}$ of naturally trained En_2ResNet20 is 92.60% which is remarkably better than that of ResNet20 reported in He et al., arXiv:1512.03385.