Table 1: Average results on the three cases of NIID Gaussian Noise (BSD68).

Methods	DnCNN	FFDNet	MemNet	NLRN	VDN
PSNR	28.43	28.55	28.52	28.61	**28.72**

Table 2: PSNR results under different ε_0^2 values on Renoir Dataset (p=11).

ε_0^2	1e-4	1e-5	1e-6	1e-7	1e-8	0
PSNR	38.81	39.39	**39.45**	39.42	39.27	39.18

Table 3: PSNR results of different p values on Renoir Dataset (ε_0^2 =1e-6).

p	7	11	15	19	23
PSNR	39.36	**39.45**	39.39	39.20	39.06

Table 4: PSNR results of different architecture combinations on Renoir Dataset.

Combinations	D-0	D-U	D-D	U-0	U-D	U-U
PSNR	38.51	**38.80**	38.67	39.18	**39.45**	39.35

Table 5: PSNR and SSIM results on DND and SIDD real Benchmarks.

Datasets	Metrics	Methods			
		DnCNN	FFDNet	CBDNet	VDN
DND	PSNR	37.90	37.61	38.06	**38.35**
	SSIM	0.9430	0.9415	0.9421	**0.9514**
SIDD	PSNR	38.65	-	38.68	**39.04**
	SSIM	0.9089	-	0.9093	**0.9151**

Figure 1: Estimated variance map of one typical real image in SIDD Dataset.

Table 6: PSNR results of different methods under the IID Gaussian noise on the gray BSD68 Dataset.

Metric	$\sigma = 15$					$\sigma = 25$					$\sigma = 50$				
	BM3D	DnCNN	FFDNet	UDNet	VDN	BM3D	DnCNN	FFDNet	UDNet	VDN	BM3D	DnCNN	FFDNet	UDNet	VDN
PSNR	31.06	31.60	**31.63**	31.39	31.61	28.56	29.18	29.16	28.84	**29.23**	25.66	26.28	26.35	26.02	**26.43**

To Reviewer 1:

Q1.1 Compared with other DL algorithms: As suggested, we have re-trained additional STOA DL methods MemNet and NLRN on our datasets, and listed the PSNR results in Tabel 1. We'll additionally cite the related references and add results in revision.

Q1.2&1.5 Pixel illuminance related noise, noise variance visualization for real dataset: Actually, this is one reason why we drop conventional i.i.d. noise assumption. Even not explicitly modeling such signal dependent property, our method can finely fit spatially variant noise, as explained in **Q1.3**, more or less delivering this noise property. As shown in Fig. 1 (noise variance by our method), our method is capable of estimating signal dependent noises, complying with practical understanding for real image noises.

Q1.3 Good estimation of local noise: Thanks for understanding and illuminating this point. On one hand, the mode of the inverse Gamma distribution in Eq. (4) is locally estimated in different image space. On the other hand, *S-Net* predicts the noise variance for each pixel via the those located in its local receptive field. Compared with conventional filtering-based variance estimation methods with pre-designed filters, our method employs a learnable CNN to adaptively fit filters for different local areas. This naturally conducts its capability on spatially variant noise estimation and robust denoising effect.

Q1.4 Sensitivity to ε_0: We have tested the sensitivity of ε_0^2 (Table 2 shows results on Renoir data), showing that our method performs stably well when setting it in around [1e-7,1e-5]. $\varepsilon_0^2 = 0$ denotes the network directly trained under the MSE loss as conventional.

To Reviewer 2:

Q2.1 Choice of inverse Gamma (IG) prior. IG is adopted because it is the conjugate prior for the variance of Gaussian distribution, enabling the posterior of $\boldsymbol{\sigma}^2$ analytically calculated. The function of the Gaussian filter with $p \times p$ window is to make IG parameters capable of being estimated locally in different image space, so as to enable the method deliver spatial noise variations, as discussed in **Q1.3**. Empirically, our method can perform consistently well for $p \in [7, 15]$. A typical example is given in Table 3.

Q2.2 Effect of network architectures. As suggested, we have tested different network architecture combinations on Renoir Dataset (Table 4), including those obtained by both with U-Net (U-U) or DnCNN (D-D), mixture of the two (U-D and D-U), and only training one under MSE loss (U-0 and D-0). It is seen that our method is not too sensitive to the chosen network architectures, and using the designed loss function can evidently improve the denoising effect beyond directly training the network as conventional.

Q2.3 Signal dependent noise. As discussed in **Q1.2**, the real-world noisy images (e.g., Fig. 1) used in the paper actually contain signal dependent noise, and the better performance of our method on them verifies its effectiveness on such typical real noises.

Q2.4 SSIM. We list SSIM comparison in Table 5, corresponding to Table 4&5 of the paper. Superiority of our method is also evident.

Q2.5 Usable in other scenarios. Yes. Just as other known denoising methods, our method can also be easily embedded into other low-level tasks, like super-resolution and deblurring. We will extend our model to other scenarios in our future investigations.

To Reviewer 3:

Q3.1 How can obtain x. Please kindly note that our problem setting is exactly the same as current supervised deep learning (DL) methods, representing the present STOA methodology for the image denoising task. All of them are trained on a pre-collected supervised dataset composing of noisy-clean image pairs $\{y_i, x_i\}$s, which are either simulated according to the in-camera pipeline, or elaborately collected in the recent real-world image denoising datasets, like Renoir and SIDD we have employed. We thus just follow what the other DL methods did. In the paper, we have actually compared with several direct supervised DL methods and two additional ones MemNet and NLRN (**Q1.1**). The advantage of our method is clear, especially in more practical non-i.i.d. noise cases. The superiority of our method actually just lies in the designing of the loss function, i.e., a variational lower bound for posterior inference (just like that of VAE compared with conventional autoencoder). It makes our model a generative one constructed under Bayesian framework with better interpretability and generality, as validated by our experiments and also evidently shown in **Q2.2**. We thus strongly believe that such a variational formulation (the main contribution of this work) should be necessary and important. We sincerely ask the reviewer to further read our descriptions on the motivation and insights of this work (Sec. 3) and the more explanations on our model provided in **Q1.3** and **Q2.2**, and reconsider the rating on this work. Thanks.

Q3.2 How S-net predicts new noise. Our method does not need additional fine-tune or other processing in the test stage. All noise predictions, including those never seen in training (as shown in Fig.2), are directly got using the fixed *S-Net* obtained in the training stage. Such generalization capability is naturally attributed to the non i.i.d. noise modeling mechanism and generative insights of our model under the Bayesian framework, in which the noise of each pixel is locally estimated by the learned *S-Net* as discussed in **Q1.3**.

Q3.3 High PSNR for BSD68. Results listed in Table 2 are obtained on sRGB images while not gray ones. Due to mutual correction among RGB channels, the results tend to be higher than those obtained on gray images. To make this clearer, we also list typical performance of our method on gray images of BSD68 Dataset in Table 6, more complying with the results reported in other works.