Fast Bayesian Uncertainty Estimation and Reduction of Batch Normalized Single Image Super-Resolution Network

Aupendu Kar, Prabir Kumar Biswas
Indian Institute of Technology Kharagpur, WB, India
Webpage: aupendu.github.io/sr-uncertainty

mailtoaupendu@gmail.com, pkb@ece.iitkgp.ac.in

Abstract

Convolutional neural network (CNN) has achieved unprecedented success in image super-resolution tasks in recent years. However, the network's performance depends on the distribution of the training sets and degrades on out-of-distribution samples. This paper adopts a Bayesian approach for estimating uncertainty associated with output and applies it in a deep image super-resolution model to address the concern mentioned above. We use the uncertainty estimation technique using the batch-normalization layer, where stochasticity of the batch mean and variance generate Monte-Carlo (MC) samples. The MC samples, which are nothing but different super-resolved images using different stochastic parameters, reconstruct the image, and provide a confidence or uncertainty map of the reconstruction. We propose a faster approach for MC sample generation, and it allows the variable image size during testing. Therefore, it will be useful for image reconstruction domain. Our experimental findings show that this uncertainty map strongly relates to the quality of reconstruction generated by the deep CNN model and explains its limitation. Furthermore, this paper proposes an approach to reduce the model's uncertainty for an input image, and it helps to defend the adversarial attacks on the image super-resolution model. The proposed uncertainty reduction technique also improves the performance of the model for out-of-distribution test images. To the best of our knowledge, we are the first to propose an adversarial defense mechanism in any image reconstruction domain.

1. Introduction

Single image super-resolution (SISR) is an ill-posed low vision problem, where we upscale the image to increase the image's spatial resolution. Besides improving the perceptual quality of images for human interpretation, SISR boosts other computer vision tasks' performance [22]. Due to the
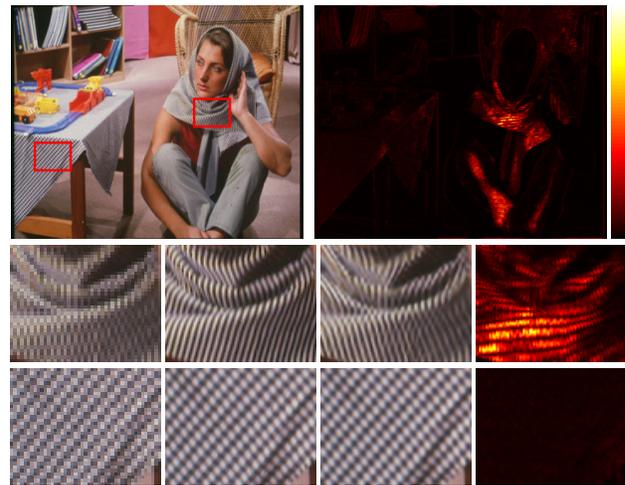
(a) LR Image   (b) HR Image   (c) SR Image   (d) Uncertainty

Figure 1. In the first row images, the left image is an LR image, and the model's uncertainty during upscaling is in the right. Two cropped regions have different textures. We can observe from two patches in the second and third-row that whenever the model fails to reconstruct the texture correctly (second-row image), it leads to higher uncertainty.

advancement of deep learning techniques, the community has developed a state-of-the-art SISR network using different topology-based deep neural architectures [26]. However, due to deep learning models' black-box nature, it is always hard to trust CNN's outcome and the limitations of the models are unknown.

Understanding the model's limitations is a crucial part of many machine learning systems. Deep learning models learn powerful abstract representations from high-dimensional images to map it to the outputs. The output results on the unknown test data are often considered blindly and believed to be reliable, which is not always true. Bayesian uncertainty plays a vital role in determining the confidence of the model during testing. Uncertainty

is a powerful tool for any prediction and reconstruction system, and the confidence of the system's output helps in the decision-making process. We use the concept of uncertainty for image super-resolution. SISR techniques using deep learning (DL) models learn features from the training image set distribution. However, real-world pictures are entirely different and contain more complex textures, which may differ from the training set. Unseen low-resolution (LR) textures during test time can produce an inappropriate reconstruction. We also have witnessed that some artifacts, blurriness, or distortions in an image due to adversarial perturbation can significantly degrade the performance of DL based models [16, 4]. Some deformation in reconstructing LR facial images may lead to wrong output in a recognition system. Any deformed reconstruction in tumor image may lead to the incorrect estimation of tumor size. Therefore, uncertainty in DL-based reconstruction models improves the transparency and trustability of the reconstruction system.

Bayesian approaches in super-resolution provide the posterior distribution of the reconstructed high-resolution (HR) image. Recent progress in Bayesian DL approaches uses Monte-Carlo (MC) samples that come from a posterior distribution via dropout [23] or batch-normalization [10]. Dropout during testing or stochastic batch mean-variance during testing helps to generate MC samples [7, 24]. Monte Carlo methods for deep learning model uncertainty estimation is successfully applied to classification, segmentation [11, 21], camera relocalization [12] problems. In our work, we use batch-normalization uncertainty to analyze SISR model uncertainty.

In this article, we propose a Bayesian approach for SISR. For this purpose, we use a widely used batch-normalization layer in our SISR network to generate MC samples. Those samples are different possible super-resolved images from a single LR image. We use the mean of those images to get the reconstruction, and the standard deviation between those images gives the uncertainty of reconstruction. We also propose a faster approach for generating MC samples, where we generate MC samples in a single forward pass instead of multiple. Due to this, it is useful in real-time applications. We also show the importance of uncertainty in adversarial perturbation and non-ideal LR image premises, and we have made an effort to understand the SISR model limitations. We also introduce a signed gradient-based uncertainty reduction technique for any test image. We perform signed gradient dependent controlled perturbation on LR images and successfully defended adversarial attacks on the SISR model. We also achieved performance improvement on non-ideal LR images.

The key contributions of our work can be summarised as follows.

- We propose a faster implementation of Monte Carlo

batch-normalization uncertainty to generate MC samples and overcome the hurdle of variable image size.
- We address the uncertainty of deep SISR models using Bayesian approach to measure it. To the best of our knowledge, we are the first to estimate uncertainty in deep learning models for image reconstruction. We also discuss the advantages and implications of uncertainty in SISR and its usefulness in understanding the model's limitations and outcome.
- We also propose an approach to reduce the model uncertainty for a test image. The uncertainty reduction method acts as a defense against the adversarial attack on the SISR model and proves to be beneficial for out-of-distribution noisy low-resolution images.

2. Prior Art

Single image super-resolution: SISR has extensive literature due to different studies in the last few decades. Recent advancement of deep learning (DL) methods has achieved significant improvement in that field. VDSR [13] proposed a deeper architecture and showed performance improves with the increase of network depth and converge faster using residual learning. After that, different DL based approaches [29, 17, 2, 5, 8, 14] have been proposed and achieved state-of-the-art performance in the standard datasets. In our work, we used VDSR architecture for uncertainty analysis as it is the first deep architecture for SISR.

Bayesian uncertainty: Bayesian models are generally used to model uncertainty, and different approaches have been developed to adapt NNs to Bayesian reasoning, like placing a prior distribution over parameters. Due to difficulty in inferencing [6] of Bayesian neural networks (BNNs), some approaches [7, 24] have been taken to approximate BNNs. Bayesian deep learning approaches utilized MC samples generated via dropout [23] or batch-normalization (BN) [10] to approximate the posterior distribution. Dropout can be treated as an approximate Bayesian model with multiple predictions through the trained model by sampling predictions using different dropout masks, and In the case of BN, stochastic parameters batch mean and batch variance are used to generate multiple predictions. Thus, the batch-normalized neural network can be approximated to the Bayesian model [24]. We use it for SISR as it is generally common in image reconstruction applications.

3. Methodology

We introduce a Bayesian approach on SISR that produces high-resolution images and a confidence map of the reconstruction. In this regard, we discuss a brief background of Bayesian inference in this section. After that, our network architecture and its modifications are discussed for Bayesian approximation. We also present a faster approach

to overcome the difficulties of estimating uncertainty in SISR applications. In the end, we introduce an uncertainty reduction technique for an unknown test image.

3.1. Bayesian Inference

We estimate a probabilistic function $F_W(I) : I_{LR} \rightarrow I_{HR}$ from a training set $D = \{\tilde{I}_{LR}, \tilde{I}_{HR}\}$ where $\tilde{I}_{LR} = \{\tilde{I}_{LR_1}, ..., \tilde{I}_{LR_n}\}$ are LR image set and its corresponding HR image set $\tilde{I}_{HR} = \{\tilde{I}_{HR_1}, ..., \tilde{I}_{HR_n}\}$. This function is approximated to generate most likely high-resolution image I'_{HR} from a low-resolution test image I'_{LR}. So the probabilistic estimation of HR test image is described as

$$p(I'_{HR}|I'_{LR}, D) = \int p(I'_{HR}|I'_{LR}, W)p(W|D)dW \quad (1)$$

where W is weight parameters of a function $F_W(I)$. We use variational inference to approximate Bayesian modeling. Most common approach is to learn approximate distribution of weights $q_\theta(w)$ by minimizing the Kullback–Leibler divergence $KL(q_\theta(w) \parallel p(W|D))$. This yields approximate distribution

$$q(I'_{HR}|I'_{LR}, D) = \int p(I'_{HR}|I'_{LR}, w)q_\theta(w)dw \quad (2)$$

In a batch-normalized neural network for bayesian uncertainty estimation, model parameters are $W_L, \gamma_L, \beta_L, \mu_L, \sigma^2{}_L$. $\theta = \{W_L, \gamma_L, \beta_L\}$ is learnable model weight parameters and $w = \{\mu_L, \sigma^2{}_L\}$ are stochastic parameters which are mean and variance of each layer. $q_\theta(w)$ is a joint distribution of weights and stochastic parameters w. w_i is mean and variance of ith sample.

3.2. Network Architecture

In this paper, we use the very deep super-resolution (VDSR) network [13] as a base architecture for an experimental purpose to analyze uncertainty. Our method is a generalized approach, and any other super-resolution network can adopt without any modifications. We have used batch-normalization (BN) for uncertainty estimation, but VDSR paper has not used batch-normalization. So we introduce it in the VDSR architecture and use this after each convolutional layer except the first and last layer. In our experiments, we achieve similar results like VDSR in our batch-normalized VDSR. We also perform same experiments on another super-resolution model and for this purpose we use SR-Resnet [18] architecture. All the detailed experiments on SR-Resnet are described in the supplementary.

3.3. Bayesian VDSR for Uncertainty Estimation

We use BN to estimate the uncertainty of the SISR network, where random batch members are selected to estimate mini-batch statistics for training. We use this stochasticity to approximate Bayesian inference, and it allows us to make a meaningful estimate of uncertainty, and it is termed

as Monte-Carlo Batch Normalization (MCBN) [24]. Generally, estimated running batch mean and batch variance are used in each BN layer during testing, but here we use stochastic batch mean and variance. We have learnable model parameters that are optimized during training, and stochastic parameters like batch mean and variances help to generate MC samples from the posterior distribution of the model. We feed-forward a test image along with different randomly sampled training batches for multiple times, and due to the stochasticity of batches, it creates various reconstructed HR images. We take the mean of those MC samples to estimate reconstruction and standard deviation for uncertainty map.

3.4. Faster approach

The main drawback of standard MCBN uncertainty estimation is that we need to process test images with different random batches to generate MC samples, and computation time increases exponentially with the increase of the number of samples in a single batch or spatial dimension of the batch. The main challenge is that in the case of SISR, test

---

Algorithm 1: Layer-wise batch mean and variance estimation for single shot MC samples generation

---

Input: training image set I
Output: layer-wise batch mean and variance set
    $\widehat{w}_L^T = \{\mu_L^T, \sigma^2{}_L^T\}$ of trained network.
    Where L is layer no and T is maximum
    number of MC samples required.
Training SR Network:
for total iterations **do**
    forall batches B of image set I **do**
        train();
    end
end
$\widehat{w}_L^T$ ***estimation:***
for T batches of B **do**
    forward pass;
    estimate $\widehat{w}_L^T$;
end

---

image size varies from thousands to millions of pixels. We can not make a larger spatial batch size during training as it takes longer computation time. While we train our model using a small patch size due to the computational constraint, we have to break larger images during testing for batch processing to keep the stochastic behavior, and it can create a patchy effect in the output images. Due to this, we propose a different approach to generate MC samples in a single batch. After training, we estimate stochastic parameters w_L of each layer using different random training batches, as shown in Algorithm 1. We use the same batch shape

during training and stochastic parameter estimation. We estimate these parameters in each BN layer for a batch, and like this, we create several stochastic parameters set for different batches. We use these stochastic parameters during testing to generate MC samples. One stochastic parameter set generates one MC sample. During testing, we concatenate the same test image based on the required number of MC samples. In the BN layer, we normalize each image separately using different stochastic parameters, as shown in Algorithm 2. Due to this, it produces various super-resolved /HR images as MC samples, which come from a posterior distribution learned from the training dataset.

Algorithm 2: Our MCBN algorithm for SISR

Input: test image I_{LR}, number of MC samples N,
      batch mean and variance of layer L
      $\widehat{w}_L^N = \{\mu_L^N, \sigma^2{}_L^N\}$
Output: image mean prediction $\hat{I}_{SR}$, predictive
      image uncertainty σ
concat I_{LR} for T times ($I^T{}_{LR}$);
$I^N{}_{SR} = F_W(I^N{}_{LR}, \widehat{w}_L^N)$;
$\hat{I}_{SR} = mean(I^N{}_{SR})$;
$\hat{\sigma} = std(I^N{}_{SR})$;

3.5. Uncertainty Reduction

Stochastic parameters of the batch-normalization layers generate different MC samples, and deviations between those samples give an uncertainty map associated with reconstructed LR image. A higher value of uncertainties mainly appears in those regions where the model fails to achieve high confidence reconstruction from LR images. Those regions are mostly the textures that are far from the training sets. Adversarial attack searches for those out-of-distribution samples where the model fails miserably. In the SISR domain, adversarial attacks perturb the LR image and push the image from the training dataset distribution of the LR image in which the model is trained. Small and visually indistinguishable perturbations on LR images lead to the visually unrealistic and unwanted synthesized texture on the reconstructed image. Even a small amount of random Gaussian noise on the Bicubic downsampled LR image can heavily degrade the performance as the degraded test image does not follow the training set distribution. Whenever a test image or specific patch of the image is far from the training set distribution, it leads to higher uncertainty.

Therefore, to bring back the sample far from the training set distribution, which eventually leads to higher uncertainty, we propose an inverse perturbation mechanism that will push the test LR image in those directions, which will subsequently lead to lowering the uncertainty for a test image. Algorithm 3 shows an uncertainty reduction algorithm

by perturbing a test image. There are two stages in this algorithm. In the first stage, average gradient directions are determined to lower the uncertainties. The second stage is the perturbation level selection.

Algorithm 3: Uncertainty reduction for an image

Input: test image I_{LR}, number of MC samples N,
      batch mean and variance of layer L
      $\widehat{w}_L^N = \{\mu_L^N, \sigma^2{}_L^N\}$
Output: perturbed test image I'_{LR}
for total updates on I_{LR} **do**
    Gradient directions for I_{LR}:
    for total iterations, T **do**
      Random Select t_1 and t_2 ;
      $I^{t_1}{}_{SR} = F_W(I_{LR}, \widehat{w}_L^{t_1})$;
      $I^{t_2}{}_{SR} = F_W(I_{LR}, \widehat{w}_L^{t_2})$;
      $\mathfrak{L} = \frac{1}{CWH} \sum (I^{t_1}{}_{SR} - I^{t_1}{}_{SR})^2$;
      $\mathfrak{N} = \mathfrak{N} + sign(\nabla \mathfrak{L})$
    end
    Perturbation Level Selection
    for different levels of perturbation, β **do**
      $I'_{LR} = I_{LR} - \beta \cdot \frac{\mathfrak{N}}{T}$;
      Calculate Uncertainty for I'_{LR}, $U(\beta)$;
      if $U(\beta) > U(\beta - 1)$ **then**
        $I'_{LR} = I_{LR} - (\beta - 1) \cdot \frac{\mathfrak{N}}{T}$;
        break ;
      end
    end
end

Gradient Direction Calculation We use the trained model F_W and the corresponding stochastic parameters set $\widehat{w}_L^N$ for uncertainty reduction. For a test image I_{LR}, we randomly select a pair of the stochastic parameter $(\widehat{w}_L^{t_1}, \widehat{w}_L^{t_2})$ from the parameter sets and use those to reconstruct two different super-resolved images $I^{t_1}{}_{SR}$ and $I^{t_2}{}_{SR}$. Our main objective is to reduce the difference between those two images, and it can be defined by minimizing pixel-wise mean-squared loss, as shown in Algorithm 3. The negative sign gradient at each pixel $\mathfrak{N}$ of the input image is the test LR image update direction to reduce uncertainty. Instead of taking a single random pair of stochastic parameters, we consider multiple pairs for better estimation of gradient directions, which eventually reduces uncertainty.

Perturbation Level Selection After obtaining the gradient directions for a test image, the main challenge is selecting the proper amount of perturbation level, which will reduce the uncertainty. If we increase the value of β, the uncertainty will decrease, and the perturbed LR image I'_{LR}

will come closer to the training set distribution. However, after a certain point, due to heavy perturbation, uncertainty can start increasing again. Therefore, we apply a simple technique for the proper selection of β. We gradually increase the perturbation level on the LR image and calculate the uncertainty using Algorithm 2 for each increment. We stop adding perturbation whenever the uncertainty starts increasing.

4. Results and Discussion

4.1. Training Details

We use the DIV2K dataset [1, 25] for training, which contains 800 training images and 100 images for validation. Five standard benchmark testing datasets, namely Set5 [3], Set14 [27], BSD100 [19], Urban100 [9], Manga109 [20] are used for performance analysis. We randomly extract patches of size 64×64 from each HR and bicubic interpolated LR image during training for a batch update. Each batch contains 16 patches. We augment the patches by horizontal flip, vertical flip, and 90-degree rotation and randomly choose each augmentation with a 50% probability. Each input patch is normalized into $[0, 1]$ before feeding to the network. We train each model with the PyTorch framework for 1000 epochs, where a single epoch constitutes 1000 batch updates. Adam optimizer [15] is used to update the weights. The learning rate is initialized to 10^{-4} and reduced to half after every 200 iterations. We use meansquared error to optimize model parameters.

4.2. Monte-Carlo Samples

4.2.1 Number of MC Samples

We get a better estimate of uncertainty and reconstruction with the increase of MC samples, with increased inference time. Hence, a proper choice of the number of MC samples is necessary due to this trade-off. The minimum number of MC samples should produce comparable results compared to batch-normalization without stochastic meanvariance and provide a stable uncertainty estimation. Figure 2(a) shows the changes in reconstructed image quality in terms of peak signal-to-noise ratio (PSNR), and Figure 2(b) presents uncertainty associated during reconstruction with the increase in the number of MC samples. The experiments performed for those plots use all the images from the BSD100 dataset. Results show that PSNR and uncertainty increase with the increase of MC samples, and later it settles to some stable values. We observe that around 35 MC samples are sufficient to produce stable results.

4.2.2 Fast MC Sample Generation

We benchmark our faster approach against standard MCBN uncertainty estimation. The time required for generating

MC Samples	MCBN	Our approach
5	14.28	1.0
10	33.48	1.97
15	52.67	2.96

Table 1. All values correspond to 'face' image of Set14. Five MC samples generation using our approach is used as a reference point (taken as 1.0) and other numbers represent how much more time is required to generate MC samples.

MC samples in standard MCBN uncertainty mainly depends on the size of the image in the dataset and the number of MC samples. We overcome these two difficulties. Our approach is much faster than conventional, as shown in Table 1. We consider 5 MC sample generation for an image using our method as a baseline. Other values in the table exhibit how many times more GPU time is required for inference. Our approach takes 14.28 times lesser execution time to generate 5 samples for an image of size 276×276.

(a) PSNR vs MC Samples   (b) Uncertainty vs MC Samples

(c) PSNR vs Uncertainty   (d) Uncertainty vs Adv. Noise Level

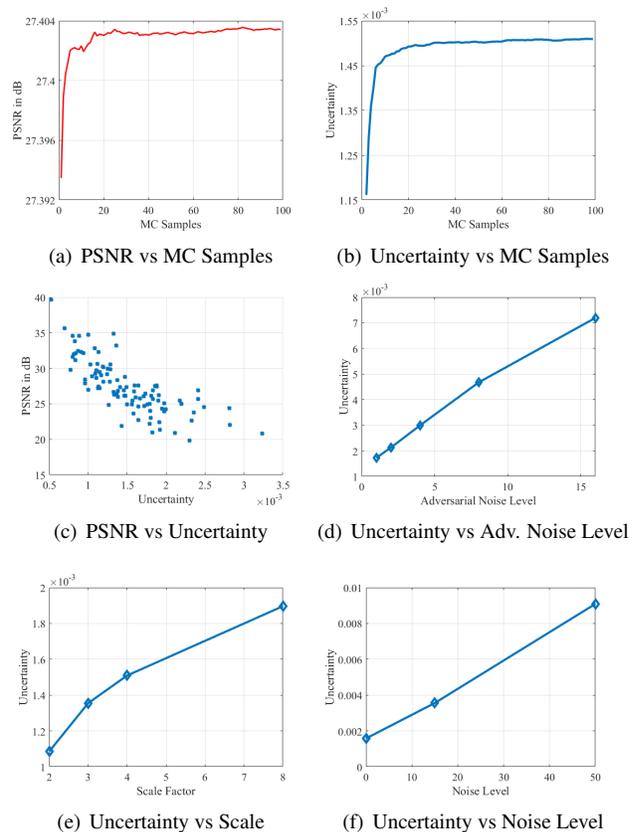(e) Uncertainty vs Scale   (f) Uncertainty vs Noise Level

Figure 2. (a)-(b) present effect of PSNR and uncertainty with the increase of MC samples and (c)-(d) present impact of PSNR and adv. noise with uncertainty. (e)-(f) present the influence of Bayesian uncertainty with different Scale factor models and random noise on LR images. (zoom for the best view.)

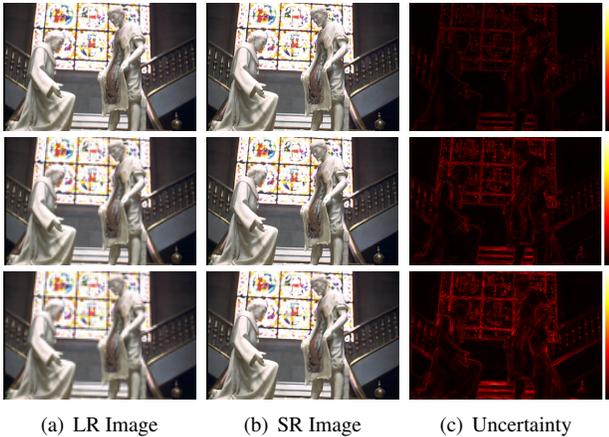| (a) LR Image | (b) SR Image | (c) Uncertainty |

Figure 3. Visual representation of the reconstructed image and corresponding uncertainty map for different scale factor models. The first-row image is for scale factor $\times 2$, the second-row image is for scale factor $\times 3$, and the third-row image is for scale factor $\times 4$.

4.3. Understanding Uncertainty

4.3.1 Behaviour of Uncertainty

We compare the average uncertainty with the quality of reconstruction, and we use PSNR as an image quality metric. Figure 2(c) shows a strong relationship between uncertainty and image quality using 100 images of BSD100 dataset. PSNR of the images decreases with the increase of uncertainty. Figure 2(e) shows that average uncertainty on BSD100 dataset increases with the increase of scale factor. However, we cannot conclude anything as we do not observe the same trend in SR-ResNet architecture as shown in supplementary. Figure 3 shows visual representations of the uncertainty for different scale factors. Uncertainty also increases with the increase of random Gaussian noise perturbation, as shown in Figure 2(f). As models are trained using bicubic downsampled noiseless images, the random noise in LR images does not match the training sets' distribution. Therefore, it leads to an increase in uncertainty. The test image moves from the ideal condition with the increase of noise in the LR image, leading to more uncertainty.

4.3.2 Uncertainty in Adversarial attack

Adversarial images are those strategically perturbed images where the model fails to perform. Adversarial images or patches can degrade performance significantly, and it makes deep learning models lesser trustworthy. We should know the limitations of the model and be aware of failure cases for high-risk applications like medical imaging.

The modified version of Iterative fast gradient sign method (I-FGSM) based adversarial small perturbation in the LR image can drastically degrade the model's performance and create fake details in the image [4]. We ana-

lyzed the performance of the SR network under adversarial attack and found that uncertainty of the model increases with the increase of adversarial perturbation, as shown in Figure 2(d). Therefore, uncertainty is a crucial factor in determining black-box deep learning models' performance in any test case scenario. We generally observe uncertainty in high-frequency regions like edges of the image, which are difficult to reconstruct, as shown in the first row of Figure 4. We can witness higher uncertainty in the airplane's edges from the first-row image without any adversarial perturbation. But, under adversarial attacks, as shown in the second row of Figure 4, we observe higher uncertainty in the smooth regions where fake artifacts are generated. In the third row image of Figure 4, we have shown partial adversarial attack results and the uncertainty map. We observe the partial attack region, a square box in the center of the image, produces higher uncertainty. We can conclude from here that uncertainty works on patch level. If any real-world scenario has adversarial patches in certain regions, it can be easily detected using the reconstructed image's prediction uncertainty. Hence uncertainty act as a tool to measure the capacity of the deep SISR model.

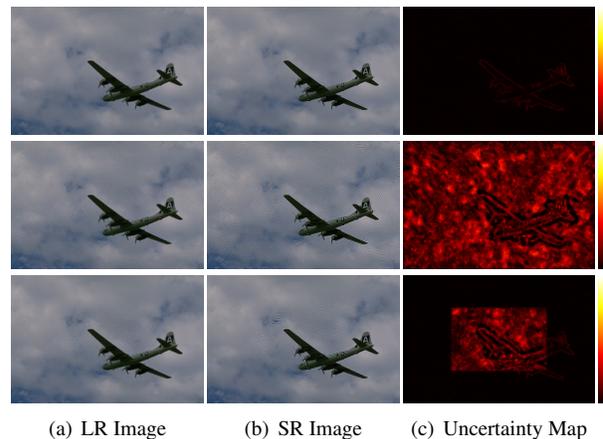| (a) LR Image | (b) SR Image | (c) Uncertainty Map |

Figure 4. This image is taken from the BSDS100 dataset. The first-row represents the LR image without any adversarial perturbation and its corresponding SR image and uncertainty map. The adversarial LR image and its outputs are shown in the second-row. The third-row image presents the LR image, which is partially perturbed by the adversarial attack, and the perturbed location is a square box at the center of the image.

4.4. Uncertainty Reduction

4.4.1 Defense Against Adversarial Attack

In the earlier section, we show that uncertainty holds a strong relation with different adversarial attack levels. The model's performance degrades severely in the adversarial attacked images, and the corresponding uncertainty map produces more uncertainties compared to non-adversarial

(a) HR   (b) LR   (c) Adversarial LR   (d) SR (No Def.)   (e) SR (After Def.)   (f) UN (No Def.)   (g) UN (After Def.)

Figure 5. Performance of our proposed adversarial defense mechanism using the Bayesian uncertainty reduction technique. The cropped region shows undesired artifacts in the SR image without any defense, and our defense mechanism successfully suppresses those artifacts. Zoom for the best view.

Type	Level of Attack	Scale Factor ($\times 2/ \times 3/ \times 4$)			
		PSNR		Uncertainty	
		No Defense	After Defense	No Defense	After Defense
No Attack	-	32.02/ 28.95/ 27.43	31.94/ 28.87/ 27.32	0.0011/ 0.0014/ 0.0015	0.0009/ 0.0011/ 0.0013
Attack	1	31.16/ 28.36/ 27.03	31.55/ 28.63/ 27.16	0.0014/ 0.0017/ 0.0017	0.0010/ 0.0013/ 0.0013
	2	29.54/ 27.27/ 26.25	30.86/28.25/ 26.89	0.0019/ 0.0022/ 0.0021	0.0013/ 0.0014/ 0.0015
	4	26.30/ 25.03/ 24.63	29.73/ 27.67/ 26.43	0.0034/ 0.0035/ 0.0030	0.0018/ 0.0018/ 0.0018
	8	21.97/ 21.78/ 22.02	27.78/ 26.60/ 25.59	0.0058/ 0.0055/ 0.0047	0.0026/ 0.0024/ 0.0023
	16	18.04/ 18.37/ 18.82	24.78/ 24.64/ 23.94	0.0086/ 0.0084/ 0.0072	0.0040/ 0.0037/ 0.0035

Table 2. Quantitative evaluation of our proposed Bayesian uncertainty reduction technique based adversarial defense mechanism.
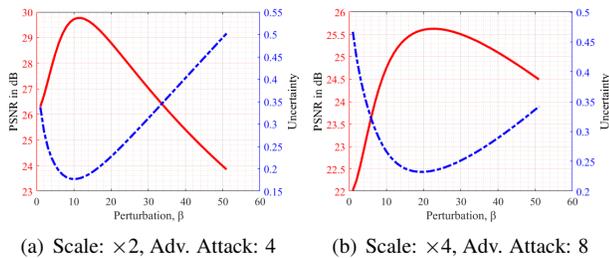
(a) Scale: ×2, Adv. Attack: 4   (b) Scale: ×4, Adv. Attack: 8

Figure 6. Effect of perturbation level for adversarial defense on quality of the image and mean uncertainty of the corresponding image.

images. As mentioned in Algorithm 3, we propose an uncertainty reduction technique that reduces pixel-wise deviations between MC samples, and we successfully defend the adversarial attack by reducing uncertainties.

In our experiments, we apply the I-FGSM attack [16] on the single image super-resolution model, as shown in [4] and it is also discussed in supplementary. We use the proposed uncertainty reduction based defense mechanism on five different levels of adversarial attacked images. The experimental results are presented in Table 2. We observe from the table that our model reduces uncertainties between MC samples and increase the model's reconstruction performance in terms of PSNR. All the experiments in Table 2 are performed on the BSD100 dataset. The table itself depicts that an increase in the level of attacks drastically reduces the reconstruction performance. Our defense mechanism successfully prevented the free fall of reconstructed image quality under adversarial attacks. We also observe that the defense mechanism does not affect non-adversarial images and posses negligible performance drop. The minute drop may be due to slightly blurring the higher uncertainty regions in the non-adversarial images like edges during uncertainty reduction. Figure 5 shows a subjective evaluation of our proposed uncertainty reduction based defense mechanism. We observe that low-resolution (LR) and adversarial LR images do not significantly differ in perception. However, the ×2 scaled super-resolved (SR) adversarial image without defense exhibits many artifacts in the image and produces more uncertainty, as shown in the uncertainty (UN) map. Our proposed defense mechanism significantly reduces those unwanted artifacts, and uncertainty

Noise Level	Scale Factor (×2/ ×3/ ×4)				
	RCAN [28]	SAN [5]	DRLN [2]	BayesianVDSR	BayesianVDSR (Uncertainty Reduction)
5	33.10/ 31.87/ **30.33**	33.70/ 31.76/ 30.24	33.73/ **31.94**/ 30.29	33.44/ 31.28/ 29.70	**34.08**/ 31.69/ 29.78
10	28.94/ 28.36/ 27.46	29.41/ 28.08/ 27.11	29.44/ 28.45/ 27.30	29.32/ 28.06/ 27.18	**31.68/ 29.73/ 28.17**
15	26.11/ 25.49/ 24.94	26.26/ 25.08/ 24.53	26.27/ 25.65/ 24.52	26.43/ 25.61/ 25.04	**29.65/ 28.31/ 27.12**

Table 3. Zero-mean Gaussian noise with different variances is added to the low-resolution images of the Set5 dataset. We compare the performance of state-of-the-art models with Bayesian VDSR and its uncertainty reduction counter-part. Note that none of the models is designed to handle the noise. Best results are in bold.

Noise Type	Scale Factor (×2/ ×3/ ×4)			
	PSNR		Uncertainty	
	Before UN Reduction	After UN Reduction	Before UN Reduction	After UN Reduction
Poisson	29.16/ 28.04/ 27.10	31.28/ 29.49/ 28.10	0.0023/ 0.0026/ 0.0026	0.0007/ 0.0010/ 0.0013
Speckle	27.80/ 26.83/ 26.22	29.73/ 28.58/ 27.45	0.0025/ 0.0029/ 0.0030	0.0010/ 0.0015/ 0.0013

Table 4. Performance of our uncertainty reduction technique on noisy LR images.

also reduces.

In our uncertainty reduction technique, we perturb a LR test image during uncertainty reduction so that uncertainty reduces for that test image. The perturbation level β plays an essential role. As we increase the perturbation level, the uncertainty reduces, and reconstructed image quality increases, as shown in Figure 6. After the optimum perturbation level, the uncertainty starts to increase due to excessive perturbation. The optimum perturbation level depends on the attack level and scale factor model we use. After getting gradient directions, we choose the perturbation level, which produces minimum uncertainty during reconstruction.

4.4.2 Noisy Low-resolution Image

Most of the deep super-resolution models handle bicubic downsampled images as input and is trained to map high-resolution counterparts of inputs. However, the real-world LR images are not the ideal bicubic downsampled version of high-resolution images. Noise is one of the critical degradation factors incorporated with real-world LR images. Some approaches consider that noise in the form of Gaussian noise and train the network. It performs well on the images that follow the Gaussian distribution. However, there is no guarantee that it will work on real-world images as real-world noise distribution varies based on the camera sensor and environmental effects. In the earlier section, we have observed that small noise degrades the Bayesian VDSR network's performance and increases the uncertainty. Therefore, we apply the proposed uncertainty reduction technique to reduce the uncertainty induced due to noisy LR images. Our experimental findings in Table 3 show that the uncertainty reduction technique improves the quality of the SR images from noisy LR counterparts where noise follows Gaussian distribution. As our model is not trained on noisy LR images, therefore for a fair comparison,

we compare state-of-the-art deep learning models trained on bicubic LR images. We observe from the table that all the models' performance degrades due to the increase of noise, and our proposed uncertainty reduction technique can prevent degradation to some extent. In our experiment, we perform a single update on the LR image for zero-mean Gaussian noise with variance 5, and we update the LR image 10 times for noise variance 10 and 15. We also experiment with other different types of noisy LR images. We consider Poisson and Speckle noise in the LR images and the uncertainty reduction performance is shown in Table 4. We witness a consistent improvement of image quality in those noisy LR images due to uncertainty reduction.

5. Conclusion

In this work, we introduced a faster Bayesian approach to estimate uncertainty in batch normalized super-resolution network. We experimentally found that the uncertainty map is like the signature map of the SISR model, which indicates how good is the model in reconstructing images. We experimentally found the small adversarial noises and even random Gaussian noises on LR images increase the uncertainty as those images do not follow bicubic downsampled training image distribution. It suggests that uncertainty represents the limitation of the model. We propose an uncertainty reduction method between MC samples by perturbing the test LR image, and it eventually prevents the adversarial attack. We have also shown that the uncertainty reduction technique is a useful tool for reconstruction performance improvement on noisy LR images. We believe that this work will set a important gateway into the study of Bayesian deep learning. A more efficient way to reduce uncertainty will make the neural network work admirably for out-of-distribution samples, and it will benefit in handling unknown degradations.

References

[1] Eirikur Agustsson and Radu Timofte. Ntire 2017 challenge on single image super-resolution: Dataset and study. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 126–135, 2017. 5

[2] Saeed Anwar and Nick Barnes. Densely residual laplacian super-resolution. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020. 2, 8

[3] Marco Bevilacqua, Aline Roumy, Christine Guillemot, and Marie Line Alberi-Morel. Low-complexity single-image super-resolution based on nonnegative neighbor embedding. 2012. 5

[4] Jun-Ho Choi, Huan Zhang, Jun-Hyuk Kim, Cho-Jui Hsieh, and Jong-Seok Lee. Evaluating robustness of deep image super-resolution against adversarial attacks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 303–311, 2019. 2, 6, 7

[5] Tao Dai, Jianrui Cai, Yongbing Zhang, Shu-Tao Xia, and Lei Zhang. Second-order attention network for single image super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 11065–11074, 2019. 2, 8

[6] Yarin Gal. *Uncertainty in deep learning*. PhD thesis, PhD thesis, University of Cambridge, 2016. 2

[7] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *International Conference on Machine Learning*, pages 1050–1059, 2016. 2

[8] Muhammad Haris, Gregory Shakhnarovich, and Norimichi Ukita. Deep back-projection networks for super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1664–1673, 2018. 2

[9] Jia-Bin Huang, Abhishek Singh, and Narendra Ahuja. Single image super-resolution from transformed self-exemplars. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5197–5206, 2015. 5

[10] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning*, pages 448–456, 2015. 2

[11] Alex Kendall, Vijay Badrinarayanan, and Roberto Cipolla. Bayesian segnet: Model uncertainty in deep convolutional encoder-decoder architectures for scene understanding. *Proceedings of the British Machine Vision Conference (BMVC)*, 2017. 2

[12] Alex Kendall and Roberto Cipolla. Modelling uncertainty in deep learning for camera relocalization. In *2016 IEEE international conference on Robotics and Automation (ICRA)*, pages 4762–4769. IEEE, 2016. 2

[13] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Accurate image super-resolution using very deep convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1646–1654, 2016. 2, 3

[14] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Deeply-recursive convolutional network for image super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1637–1645, 2016. 2

[15] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *ICLR*, 2014. 5

[16] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial examples in the physical world. *ICLR Workshop track*, 2017. 2, 7

[17] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4681–4690, 2017. 2

[18] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4681–4690, 2017. 3

[19] D Martin, C Fowlkes, D Tal, and J Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*, volume 2, pages 416–423. IEEE, 2001. 5

[20] Yusuke Matsui, Kota Ito, Yuji Aramaki, Azuma Fujimoto, Toru Ogawa, Toshihiko Yamasaki, and Kiyoharu Aizawa. Sketch-based manga retrieval using manga109 dataset. *Multimedia Tools and Applications*, 76(20):21811–21838, 2017. 5

[21] Abhijit Guha Roy, Sailesh Conjeti, Nassir Navab, and Christian Wachinger. Inherent brain segmentation quality control from fully convnet monte carlo sampling. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 664–672. Springer, 2018. 2

[22] Aditya Sharma, Prabhjot Kaur, Aditya Nigam, and Arnav Bhavsar. Learning to decode 7t-like mr image reconstruction from 3t mr images. In *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, pages 245–253. Springer, 2018. 1

[23] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958, 2014. 2

[24] Mattias Teye, Hossein Azizpour, and Kevin Smith. Bayesian uncertainty estimation for batch normalized deep networks. In *International Conference on Machine Learning*, pages 4914–4923, 2018. 2, 3

[25] Radu Timofte, Shuhang Gu, Jiqing Wu, Luc Van Gool, Lei Zhang, Ming-Hsuan Yang, Muhammad Haris, et al. Ntire 2018 challenge on single image super-resolution: Methods and results. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2018. 5

[26] Zhihao Wang, Jian Chen, and Steven CH Hoi. Deep learning for image super-resolution: A survey. *arXiv preprint arXiv:1902.06068*, 2019. 1

[27] Roman Zeyde, Michael Elad, and Matan Protter. On single image scale-up using sparse-representations. In *International conference on curves and surfaces*, pages 711–730. Springer, 2010. 5

[28] Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun Fu. Image super-resolution using very deep residual channel attention networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 286–301, 2018. 8

[29] Yulun Zhang, Yapeng Tian, Yu Kong, Bineng Zhong, and Yun Fu. Residual dense network for image super-resolution. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 2