Noise-free Optimization in Early Training Steps for Image Super-Resolution

MinKyu Lee, Jae-Pil Heo*

Sungkyunkwan University
{bluelati98, jaepilheo}@g.skku.edu

Abstract

Recent deep-learning-based single image super-resolution (SISR) methods have shown impressive performance whereas typical methods train their networks by minimizing the pixel-wise distance with respect to a given high-resolution (HR) image. However, despite the basic training scheme being the predominant choice, its use in the context of ill-posed inverse problems has not been thoroughly investigated. In this work, we aim to provide a better comprehension of the underlying constituent by decomposing target HR images into two subcomponents: (1) the **optimal centroid** which is the expectation over multiple potential HR images, and (2) the **inherent noise** defined as the residual between the HR image and the centroid. Our findings show that the current training scheme cannot capture the ill-posed nature of SISR and becomes vulnerable to the inherent noise term, especially during early training steps. To tackle this issue, we propose a novel optimization method that can effectively remove the inherent noise term in the early steps of vanilla training by estimating the optimal centroid and directly optimizing toward the estimation. Experimental results show that the proposed method can effectively enhance the stability of vanilla training, leading to overall performance gain. Codes are available at github.com/2minkyulee/ECO.

1 Introduction

With the drastic development of deep-learning-based techniques, recent single image super-resolution (SISR) methods have shown promising performance against previous methods. Here, the two primary objectives of SISR are; achieving precise reconstruction at the pixel level (known as fidelity-oriented methods); and producing visually appealing (**??**) images (referred to as perceptual-quality-oriented methods). While perceptual-quality-oriented methods have become increasingly popular in recent years, fidelity-oriented methods still remain a mainstream of research due to the high demand for reliable reconstruction. Accordingly, we limit our focus to fidelity-oriented methods in this paper.

Typically, modern fidelity-oriented SISR networks adopt a very simple training strategy. In most cases, the only objective is to optimize the likelihood of the predicted image based on pairs of HR images and corresponding downscaled LR images. Here, with fair assumptions on the distribution of image spaces and empirical results (**?**), the majority decision of the objective function is narrowed down as the pixel-wise L_1 loss. However, although this basic training scheme is the predominant choice, its use and limitations have not been thoroughly investigated, particularly with regard to the ill-posed nature of image super-resolution.

In this paper, we aim to analyze the underlying components of vanilla training in the context of SISR tasks and systematically develop the current training process. We start our analysis by decomposing the original HR image into two key components: *optimal centroid* and *inherent noise*. Given the ill-posed nature of image SR (**??**), we define the optimal centroid as the expectation over multiple potential HR images that downsamples to an identical LR image instance. Additionally, we define the inherent noise term as the residual between the HR image sample and the optimal centroid, which is a fundamental component underlying in each HR image instance.

Our findings are that vanilla training neglects the ill-posed nature of inverse problems, which results as a residual noise term per sample. Consequently, the overall training procedure becomes highly dependent on each HR image sample within a mini-batch, leading to noisy and unstable training, especially in early training steps.

In order to tackle this issue, we take the ill-posed nature of SR into account and formulate a noise-free objective, which simplifies as minimizing the L_1 distance between the network's estimation and the expectation over all possible HR samples (i.e., the true centroid term). However, since direct usage of this objective is impossible due to the intractability of the centroid term, we utilize a surrogate objective that can effectively act as a substitute for the intractable objective. Specifically, we estimate the true centroid by an empirical centroid obtained from pretrained SR networks and define a tractable objective for noise-free optimization. Further, we show that Knowledge Distillation (KD) can be understood as a specific case of this noise-free optimization, but with apparent flaws: spatial inconsistency. We make a quick fix for the shortcomings of KD and construct a noise-free training objective that optimizes directly towards the empirical centroid while being both tractable and spatially aligned. It turns out that the proposed objective can lead to well-

*Corresponding author

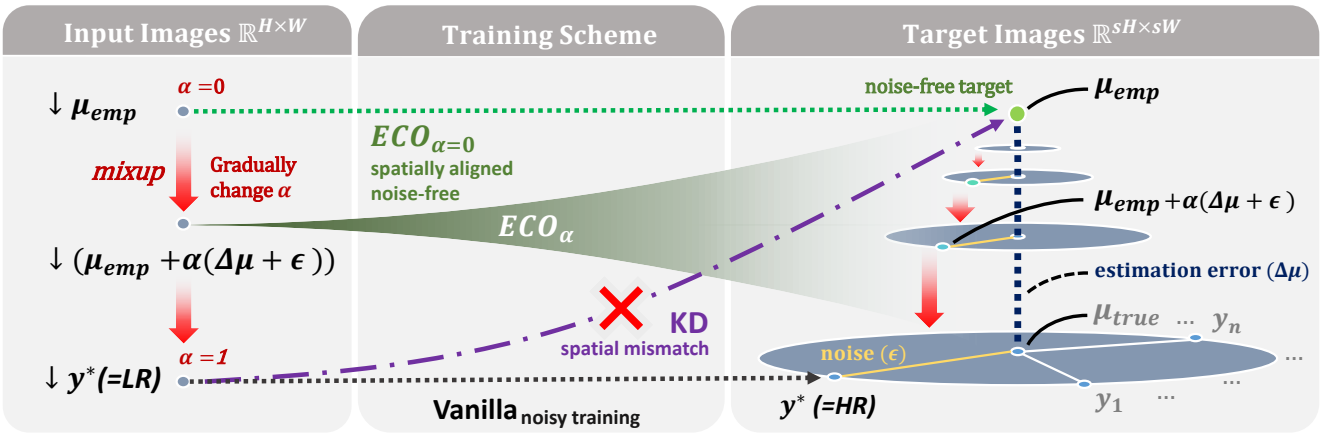

Figure 1: Visualization of our method (ECO) compared to vanilla training and knowledge distillation (KD). Data points indicated in gray text are not available during training. Vanilla training leads to noisy training since it is unaware of the inherent noise ϵ, which is defined as the difference of a given HR image y^* and the expectation over all possible HR images, μ_{true}. On the other hand, KD benefits from noise-free targets but suffers from spatial inconsistency between the input and target images as in Eq.(**??**). The proposed objective Eq.(**??**) benefits from noise-free training while being spatially aligned. Then, we overcome the limitations that arise by removing the estimation error term $\Delta\mu := \mu_{\text{true}} - \mu_{\text{emp}}$ with a smooth transition from the proposed objective to the original objective. Remarkably, the overall solution can be greatly simplified with the use of *mixup* strategy as in Eq.(**??**) (Section **??**). Starting from synthetic data pairs ($\alpha = 0$), gradually migrate to real data pairs ($\alpha = 1$). This way, we enjoy noise-free training during the early steps, and finetune the network with supervision from real data samples in later steps.

behaving loss values and gradients (i.e., better Lipschitzness) enabling stable optimization which is especially beneficial in the early steps of training. At last, we address the limitations that come from the estimation error and provide a simple method to overcome this. With a smooth transition between the proposed noise-free objective and the original loss, it is shown that the proposed training framework can benefit from stable training during early steps and minimize shortcomings of approximation errors in later steps.

To sum up, the major contribution of this work is in offering improved comprehension of the underlying processes involved in training neural networks for SISR tasks. This is further extended to a novel training framework, which we refer to as **Empirical Centroid-oriented Optimization (ECO)**. Experimental results show that ECO can lead to performance gain against vanilla training by enabling stable training and providing well-behaving optimization landscapes, especially helpful in the early training stages.

2 Probabilistic Modeling

2.1 Traditional objective function

With plausible assumptions of the HR image manifold, the widely used MLE strategy in low-level vision tasks are formulated as minimizing the L_p-norm. Here, the majority choice in SR tasks is the L_1-norm since it has been empirically shown to have better convergence than the L_2-norm (**?**). Accordingly, typical methods employ pixel-wise L_1 loss as the objective function where each HR image sample from the training dataset is treated as the sole ground truth image. Thus, it is a clear choice to construct the objective function in the form of a loss for a single data point as follows:

$$\begin{aligned} f &:= R^{H\times W} \rightarrow R^{sH\times sW} \\ L_1(HR, SR) &= ||y^* - f(x)||_1, \end{aligned} \tag{1}$$

where $f(\cdot)$ is the SR network with scale-factor s, which is piece-wise linear (**?**) with only ReLU (**?**) as the non-linearity, and y^*, x each corresponds to the HR, LR image sample in the training dataset, respectively.

2.2 Optimal centroid and inherent noise

Before we start our analysis, we define two fundamental components of ill-posed inverse problems: (1) the optimal centroid μ_{true} which is the expectation over multiple plausible solutions, and (2) the inherent noise ϵ which is the residual between the optimal centroid and a single data point. Here, the inherent noise term ϵ can be understood as a factor being highly random and indeterministic due to its ill-posed nature. In terms of SISR, we can define μ_{true} and ϵ with respect to an observed LR image x and the corresponding HR image sample y as following:

$$\mu_{\text{true}} := \int yp(y|x)dy, \tag{2}$$

$$\epsilon := y - \mu_{\text{true}}, \tag{3}$$

where, ϵ is expected to reside in high-frequency regions within every HR image sample, which makes exact pixel-wise reconstruction impossible. Accordingly, representing the vanilla L_1 loss in terms of the components defined above, the original objective function can be reformulated as follows:

$$||y^* - f(x)|| = ||\mu_{\text{true}} + \epsilon^* - f(x)||, \tag{4}$$

where ϵ^* is the inherent noise term for ground truth image y^* in the training dataset. In the following sections, we will provide a comprehensive analysis based on this formulation.

2.3 Modifying the objective function

Taking the ill-posed nature into account. Regarding the ill-posed nature of SISR, multiple HR images can correspond to a single LR image. Therefore, following general principles in machine learning, it is natural to maximize the likelihood over all plausible solutions. Accordingly, we begin our investigation by taking the posterior distribution into account and delve deeper into the underlying essentials of image super-resolution as below:

$$\begin{aligned} &\int ||y - f(x)||p(y|x)dy \\ =&\int ||\mu_{\text{true}} + \epsilon - f(x)||p(y|x)dy. \end{aligned} \tag{5}$$

Then given an LR image x, an ideal SR model should estimate μ_{true}, which is the optimal point of maximum likelihood, regarding that $\mathbb{E}_{p(y|x)}(y) = \mu_{\text{true}}$ by construction.

Vanilla training induces noisy training. It is worth noting that the original loss Eq.(**??**) is a specific case of Eq.(**??**). If we let $p(y|x)$ as a Delta function where $p(y|x) = 0$ for all points except for $y = y^*$, Eq.(**??**) is found to be identical to the original objective function. Based on this observation, we can conclude that the current training protocol, indeed, fails in capturing the ill-posed nature of inverse problems. Instead, it treats the given HR sample as a unique and well-defined solution. However, this assumption does not account for the non-deterministic mapping from LR to HR, which makes the use of the Delta function for $p(y|x)$ less appropriate. Moreover, this induces *inherent noise* ϵ per every HR image, which can potentially hinder the stability of the training procedure. However, in general, it is hard to disentangle the noise term since μ_{true} is intractable. In further sections, we provide systematic methods to remove the noise term and enable optimization towards the centroid.

3 Noise-free Objective Function

3.1 Removing the noise term

In this section, our goal is to remove the inherent noise term in Eq.(**??**), which can hinder the optimization, and only retain the centroid term. For any measurable and convex function $\phi(\cdot)$, we can obtain a lower bound of the expectation as $\mathbb{E}(\phi(\cdot)) \geq \phi(\mathbb{E}(\cdot))$ by Jensen's inequality. Since all L_p-norms are convex for $p \geq 1$; and μ_{true} and $f(x)$ are independent from y; and $\mathbb{E}_{y\sim p(y|x)}(\epsilon) = 0$ by definition, Eq.(**??**) can be simplified as following:

$$\begin{aligned} &\mathbb{E}_{y\sim p(y|x)}(||\mu_{\text{true}} + \epsilon - f(x)||) \\ \geq&||\mathbb{E}(\mu_{\text{true}}) + \mathbb{E}(\epsilon) - \mathbb{E}(f(x))|| \\ =&||\mu_{\text{true}} - f(x)||. \end{aligned} \tag{6}$$

By eliminating the *per sample* inherent noise, we obtain a noise-free lower bound of the original objective function.

3.2 Empirical centroid estimation

Although a noise-free objective has been obtained in Eq.(**??**), the true centroid term is still intractable and cannot be directly utilized since it involves taking the expectation over an infinite number of possible HR images. Here, pretrained networks serve as a remedy to the problem at hand. It has been observed that low-level vision methods with pixel-wise loss implicitly tend to estimate the average among all plausible estimations (**????**) . This phenomenon, which we refer to as *centroid-oriented optimization*, is acknowledged as a limitation of the training paradigm. However, by carefully integrating the retrospective centroid-oriented optimization phenomenon into the original training scheme in advance (i.e., by explicitly targeting the centroid), surprisingly, we can achieve favorable results. To this extent, we employ a pretrained super-resolution network as a centroid estimator. Thus, we refer to the estimation of a pretrained network as an *empirical centroid*, which can be simply defined as follows:

$$\mu_{\text{emp}} := \hat{f}(x), \tag{7}$$

where $\hat{f}$ is the pretrained SR network. Here, the empirical centroid μ_{emp} can be understood as the expectation, but with regard to the learned natural image prior obtained by the training dataset of the pretrained network.

4 Estimation Error of Empirical Centroids

In the previous section, we leveraged a pretrained network as an approximation of the centroid of the posterior distribution. However, even the state-of-the-art pretrained networks are followed by estimation errors, and thus should not be treated as ideal networks. Here, we examine the estimation errors from the perspectives of both (1) low-frequency (LF) components, which can be observed when SR images do not downsample to the original LR images, and (2) high-frequency (HF) components, which are the case when SR images only contain limited sharp details, below the theoretical upper-bound of pixel-wise reconstruction. Hence, we start this section by reformulating Eq.(**??**) as following:

$$||(\mu_{\text{emp}} + \Delta\mu) - f(\downarrow(\mu_{\text{emp}} + \Delta\mu + \epsilon))||, \tag{8}$$

where $\Delta\mu := \mu_{\text{true}} - \mu_{\text{emp}}$ is the estimation error and $\downarrow$ is the downsampling operation. We emphasize that these limitations of pretrained networks should be taken into account, which will be further discussed in the following sections.

Revisiting Knowledge Distillation. Here, we demonstrate that a well-known training technique, Knowledge Distillation (KD), can be simply represented in terms of the components derived in the previous sections as below:

$$\begin{aligned} &||\hat{f}(x) - f(x)|| \\ =&||\mu_{\text{emp}} - f(x)|| \\ =&||(\mu_{\text{emp}} + \Delta\mu) - f(\downarrow(\mu_{\text{emp}} + \Delta\mu + \epsilon))||, \end{aligned} \tag{9}$$

where the first row is the original formulation of KD and the others are equivalent objectives in terms of our observation. This can be understood as a special case of Eq.(**??**), with $\Delta\mu = 0$ only on the left term. In other words, the objective of KD (Eq.(**??**)) neglects the estimation error of the

teacher model in the target image but leaves it in the LR image. However, predictions of pretrained networks may not downsample to the original LR image precisely due to the LF components of $\Delta\mu$, and conversely, the given HR image will not align with the corresponding LR image. We refer to this discrepancy as spatial inconsistency between the input and target images, highlighting a critical limitation in the formulation of KD. Specifically, this spatial inconsistency hinders KD to provide proper supervision, thereby leading to potential instability in the training process. Additionally, since the estimation error term $\Delta\mu$ of the target image is ignored, this term will not be optimized which leads to limited performance bounded by the teacher network. Overall, while KD-based training may benefit from the noise-free objective and converge faster in the early steps of training, it will suffer from additional challenges by ignoring $\Delta\mu$ only in the target image.

5 Empirical Centroid-oriented Optimization

In this section, we make a quick fix on the limitations of conventional KD observed above. We construct a noise-free optimization objective in a spatially consistent manner, followed by a method to handle the estimation error.

5.1 Spatially consistent noise-free objective

Regarding that μ_{true} are linear combinations of plausible HR images, $f(\downarrow (y^*)) = f(\downarrow (\mu_{\text{true}}))$ holds if the network f and the downsampling operation $\downarrow$ are linear. By taking into account the piece-wise linearity (**?**) of f and the fact that "plausible" HR images downsample to identical images by construction, we make a fair approximation of Eq.(**??**) as follows:

$$||\mu_{\text{emp}} + \Delta\mu - f(\downarrow (\mu_{\text{emp}} + \Delta\mu))||. \tag{10}$$

Instead of assuming $\Delta\mu = 0$ only on the left side as in KD, we remove $\Delta\mu$ in **both** terms of the approximation and propose an objective as below:

$$\begin{aligned} &||\mu_{\text{emp}} + \cancel{\Delta\mu} - f(\downarrow (\mu_{\text{emp}} + \cancel{\Delta\mu}))|| \\ =&||(\mu_{\text{emp}}) - f(\downarrow (\mu_{\text{emp}})||. \end{aligned} \tag{11}$$

This way, we obtain a tractable noise-free objective function, which enables the proposed Empirical Centroid-oriented Optimization (ECO) without risking the optimization procedure from spatial inconsistency observed in KD.

5.2 Taking the estimation error into account

Trade-off of removing the error term. While it is important to prevent highly random and noisy HF components from disturbing the training, removing more HF components than required (i.e., over-smoothing) will lead to failure in providing sufficient supervision for necessary detail recovery. Regarding that pretrained networks can fail to generate sharp details, the problem of insufficient HF supervision still remains in the objective in Eq.(**??**). Thus, Eq.(**??**) has a trade-off between stable training and the limited capability of HF supervision. In practice, the impact of neglecting $\Delta\mu$ can empirically be larger than the benefit of noise-free objective after sufficient training iterations, where networks need to be fine-tuned. Overall, both our tractable noise-free objective Eq.(**??**) and the vanilla training objective Eq.(**??**) come with their own set of advantages and disadvantages.

Mixup as rescue. To this extent, we propose a simple and efficient workaround to capture the advantages of both Eq.(**??**) and Eq.(**??**). The proposed method starts by training the network with our tractable noise-free objective in Eq.(**??**). However, once adequate convergence is achieved, we switch the objective to the original objective Eq.(**??**) and obtain additional supervision on HF components. Remarkably, it turns out that this type of approach can be formulated with a well-known data augmentation method, *mixup* (**?**). As the first step, we reformulate the original loss function Eq.(**??**) as follows:

$$\begin{aligned} &||y^* - f(\downarrow (y^*))|| \\ =&||(\mu_{\text{true}} + \epsilon) - f(\downarrow (\mu_{\text{true}} + \epsilon))|| \\ =&||(\mu_{\text{emp}} + 1(\Delta\mu + \epsilon)) - f(\downarrow (\mu_{\text{emp}} + 1(\Delta\mu + \epsilon))||. \end{aligned} \tag{12}$$

Equally, the objective function based on mixup can be interpreted as an additive term of a single data pair and another as below:

$$\begin{aligned} &L(\alpha Y_1 + (1-\alpha)Y_2, \phi(\alpha X_1 + (1-\alpha)X_2)) \\ =&L(Y_2 + \alpha(Y_1 - Y_2), \phi(X_2 + \alpha(X_1 - X_2)), \end{aligned} \tag{13}$$

where $L(\cdot,\cdot)$ is an arbitrary loss function with inputs X_1, X_2, targets Y_1, Y_2 and the network to optimize as ϕ. Here, if we let $L(\cdot,\cdot)$ as the pixel-wise norm, $\phi = f$, (X_1, Y_1) as the original data pair (x, y^*) and (X_2, Y_2) as the synthetic data pair $(\downarrow (\mu_{\text{emp}}), \mu_{\text{emp}})$, we can obtain our final objective function as follows:

$$\begin{aligned} &||(\mu_{\text{emp}} + \alpha(y^* - \mu_{\text{emp}})) - f(\downarrow (\mu_{\text{emp}} + \alpha(y^* - \mu_{\text{emp}})))|| \\ =&||(\mu_{\text{emp}} + \alpha(\Delta\mu + \epsilon^*)) - f(\downarrow (\mu_{\text{emp}} + \alpha(\Delta\mu + \epsilon^*)))||. \end{aligned} \tag{14}$$

With a smooth transition of $\alpha = 0$ to $\alpha = 1$, we can easily balance through the spatially aligned tractable noise-free objective ($\alpha = 0$) and the vanilla objective ($\alpha = 1$). It should be noted that the inherent noise will be reintroduced back into the training as α increases. However, our empirical findings in Sec.**??** reveal that the early stages of training play a crucial role in overall performance. In later steps, networks become relatively stabilized, allowing them to tolerate the reintroduced noise while benefiting from enhanced high-frequency (HF) supervision. Overall, this balanced approach allows for the advantages of noise-free training in the early stages without sacrificing the benefits of HF supervision in later training. By preprocessing synthetic images and parallelizing mixup with separate CPU processes, the proposed method can be implemented in just a few lines of code. The overall framework of our method is illustrated in Fig.**??**. Unless specified otherwise, the term 'ECO' throughout this paper refers to our proposed method together with the usage of the mixup strategy described in Eq.(**??**).

Difference with conventional mixup. The proposed method is a mixture of the original (HR, LR) image pairs and synthetic reconstruction of the *identical* images. On the other hand, conventional mixup refers to blending between

different data samples in order to augment limited data samples. Note that these two methods are fairly orthogonal and can be applied simultaneously.

6 Experiments

6.1 Analyzing the impact of noise-free training

We use EDSR-baseline (**?**) as the representative model and investigate the impact of the noise-free objective obtained in Eq.(**??**), without mixup.

Exploring the optimization landscape. Following (**?**), we identify the impact of the proposed noise-free objective within the training process by investigating the optimization landscape and the Lipschitzness of the loss function. At each specific training point, we move through the gradient direction and observe the loss variation and the maximum gradient difference in terms of L_2-norm, as illustrated in Fig.**??**. Through the use of the noise-free objective, we observe well-bounded loss values, which aligns with our theoretical analysis. Moreover, of greater importance is that while vanilla training leads to sharp spikes during early training steps, noise-free training shows well-bounded gradients. In other words, noise-free training demonstrates a notably improved level of effective β-smoothness (**??**). In the context of gradient-based training methods, it is clear that the overall training procedure can be significantly influenced by gradient behaviors. Specifically, vanishing or exploding gradients can raise additional challenges when training deep networks. Thus, by having a well-behaving and predictable gradient with the proposed noise-free objective, we can alleviate these issues and obtain faster convergence with improved stability. This observation underlines the significance of noise-free training during the early stages, as it minimizes fluctuations and instabilities that could hinder the learning process. By enhancing stability in these crucial initial steps, our method can lead to an overall performance gain, setting a strong foundation for later stages of training.

Comparison against vanilla training and KD. In Fig.**??**, we provide training curves of noise-free training (w/o mixup) against vanilla training and knowledge distillation (KD). It demonstrates that KD can also lead to slightly faster convergence during early training since the formulation of KD is also expected to have noise-free targets. However, we have shown that it is followed by a fundamental limitation: spatial inconsistency between input and target images. Accordingly, the final performance turns out to be worse than that of vanilla training, while the proposed spatially aligned noise-free objective obtains overall performance gain. Remarkably, despite the only changes being the construction of LR images, it shows significant improvement.

Comparison over various batch-size. With smaller mini-batch sizes, each gradient step becomes more reliant on every individual data point within the batch. In the case of vanilla training, the training procedure becomes more susceptible to *per sample* noise originating from each image instance. Comparatively, the proposed method is relatively free from per-sample noise, which enables additional robustness to smaller mini-batch size selection. To validate

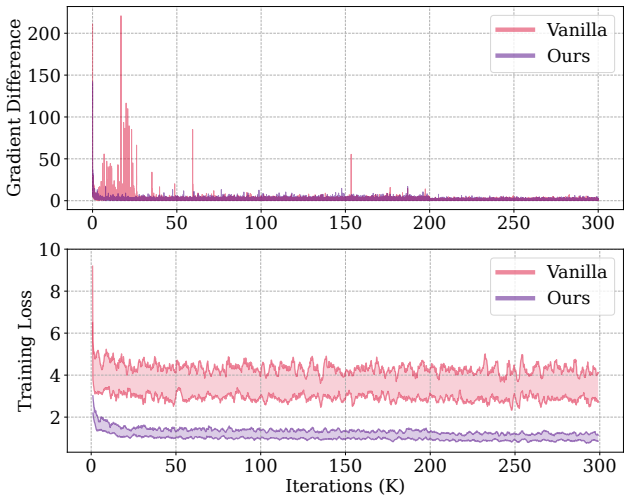

Figure 2: Visualization of maximum gradient difference and the loss variation. Spikes of gradient differences indicate that the gradients are not well-bounded (i.e., not Lipschitz).

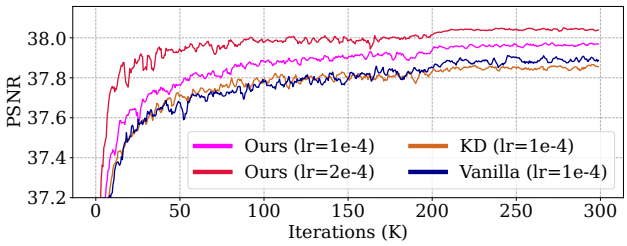

Figure 3: Comparison of our method (w/o mixup) with KD and vanilla training on Set5. It verifies the impact of spatial inconsistency in training image pairs.

the statement, we perform extensive experiments over various selections of smaller mini-batch sizes as in Fig.**??**. The mini-batch size is chosen as 2, 4, 8, and 16 where 16 is the default setting for most works. As demonstrated in Fig.**??**, vanilla training shows fluctuating PSNR scores with small mini-batch sizes, especially in early training steps, while our method provides increased stability and faster convergence over various mini-batch size choices.

Empirical impact of the estimation error. Fig.**??** illustrates the empirical trade-off between smoother gradients and the ignorance of the estimation error. In the early stages, we can observe clear improvement when training with the proposed noise-free objective. However, the impact of the estimation error empirically increases, and the final performance turns out to be lower than that of the original training scheme if the mixup strategy is not used. Together with mixup, it is shown that we can obtain superior performance over the entire training steps. We have further analyzed the mixup strategy by shifting the scheduling hyperparameter α in Eq.(**??**) but did find it to be significant.

Scale	Model	Method	Set5	Set14	BSD100	Urban100	Manga109
×2	EDSR (**?**)	Vanilla	38.18 / 0.9612	33.82 / 0.9197	32.33 / 0.9016	32.83 / 0.9349	39.05 / 0.9777
	EDSR (**?**)	ECO (ours)	**38.29 / 0.9615**	**34.07 / 0.9210**	**32.37 / 0.9022**	**33.07 / 0.9369**	**39.26 / 0.9782**
	RCAN (**?**)	Vanilla	38.26 / **0.9615**	34.04 / **0.9215**	32.35 / 0.9019	33.05 / 0.9364	39.34 / **0.9783**
	RCAN (**?**)	ECO (ours)	**38.28 / 0.9615**	**34.07 / 0.9215**	**32.39 / 0.9023**	**33.22 / 0.9378**	**39.39 / 0.9783**
×3	EDSR (**?**)	Vanilla	34.70 / 0.9294	30.58 / 0.8468	29.26 / 0.8095	28.75 / 0.8648	34.17 / 0.9485
	EDSR (**?**)	ECO (ours)	**34.80 / 0.9302**	**30.64 / 0.8476**	**29.32 / 0.8108**	**28.95 / 0.8679**	**34.36 / 0.9496**
	RCAN (**?**)	Vanilla	34.80 / 0.9302	30.62 / 0.8476	29.32 / 0.8107	29.01 / 0.8685	34.48 / 0.9500
	RCAN (**?**)	ECO (ours)	**34.86 / 0.9306**	**30.68 / 0.8484**	**29.33 / 0.8111**	**29.09 / 0.8700**	**34.56 / 0.9504**
×4	EDSR (**?**)	Vanilla	32.50 / 0.8986	28.81 / 0.7871	27.71 / 0.7416	26.55 / 0.8018	30.97 / 0.9145
	EDSR (**?**)	ECO (ours)	**32.59 / 0.8998**	**28.90 / 0.7892**	**27.78 / 0.7432**	**26.77 / 0.8064**	**31.32 / 0.9182**
	RCAN (**?**)	Vanilla	**32.71** / 0.9008	28.87 / 0.7887	27.77 / 0.7434	26.83 / 0.8078	31.31 / 0.9168
	RCAN (**?**)	ECO (ours)	32.70 / **0.9011**	**28.91 / 0.7895**	**27.80 / 0.7437**	**26.88 / 0.8086**	**31.38 / 0.9174**

Table 1: Quantitative comparison of the proposed method ECO (w/ mixup) against vanilla training. We report PSNR (dB) and SSIM scores for ×2, ×3, and ×4 SR over standard benchmark datasets. The best result are highlighted in **bold**.

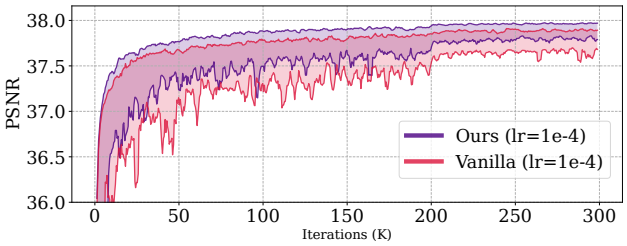

Figure 4: Validation results are reported for both vanilla training and the proposed method (without mixup) across mini-batch sizes of 2, 4, 8, and 16. The shaded regions indicate the minimum and maximum PSNR values at each iteration across all settings. Noise-free optimization enables additional stability throughout various batch-size choices.

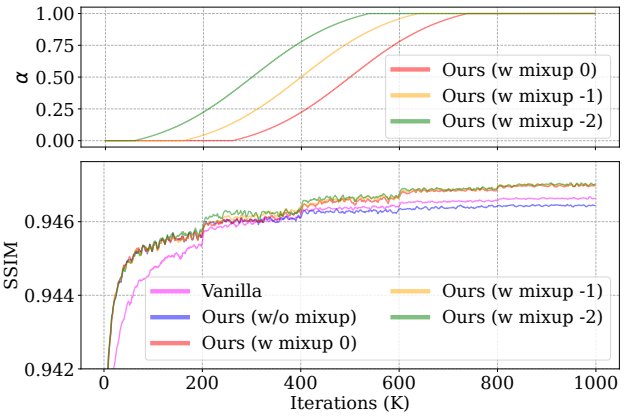

Figure 5: Validation results over various configurations of mixup. Without mixup, the performance is limited due to neglecting the estimation error factor $\Delta\mu$ as in Eq.(**??**).

6.2 Evaluation on the state-of-the-art methods

Experimental Setup. We validate the effectiveness of our method on benchmark datasets: Set5 (**?**), Set14 (**?**), BSD100 (**?**), Urban100 (**?**) and Manga109 (**?**). We reproduce all methods and mixup is used for our method. For Tab.**??**, we follow (**?**) and train networks with larger mini-batch size and fewer iterations in order to reduce the overall training time. See the supplementary materials for details.

Benchmark comparison. In Tab.**??**, we compare the proposed training scheme against vanilla training in standard SR settings. Specifically, evaluation is performed for ×2, ×3 and ×4 SR tasks with bicubic downsampling. It demonstrates that our method leads to sustainable performance gain in terms PSNR and SSIM over standard benchmark datasets. In qualitative comparison (Fig.**??**) for ×4 SR, we can clearly see that the proposed method provides more visually pleasing results, successfully recovering high-frequency details.

Larger scale factor and adaptation to real-world. We further perform extensive experiments comparing our method against vanilla training in ×8 SR task and real-world ×2 SR settings. In the case of the real-world setting, LR images with additive color Gaussian noise were used for both training and evaluation and the average score of 10 different runs is reported. Tab.**??**.(c) and Tab.**??**.(d) indicate that the proposed training framework leads to performance gain in both real-world ×2 SR and bicubic ×8 SR. Remarkably, we reach comparable performance to vanilla training with only 20% of the total iterations for ×8 SR. It verifies the higher benefits of noise-free training when the inherent noise term is expected to exhibit greater randomness.

Independence of architecture and loss. In Tab.**??**.(a-b), we further validate the proposed training framework with SwinIR (**?**) and with the L_2 loss, respectively. Experimental results verify that the application of the proposed method is not limited to only CNN architectures or the L1 loss.

Figure 6: Visual comparison of the proposed method and vanilla training for $\times 4$ SR. **Zoom in for best view.**

	Set5	Set14	Urban100	Manga109
(a) $\times 2$ SR on SwinIR-small				
Vanilla	38.19/**.9613**	33.93/.9203	32.74/.9338	39.11/**.9781**
ECO	**38.21/.9613**	**33.96/.9209**	**32.78/.9345**	**39.16/.9781**
(b) $\times 2$ SR on EDSR-baseline with L2 loss				
Vanilla	37.89/.9601	**33.47**/.9167	31.74/.9246	38.12/.9759
ECO	**37.94/.9602**	**33.47/.9170**	**31.77/.9249**	**38.34/.9764**
(c) $\times 2$ SR on EDSR-baseline, on real-world dataset				
Vanilla	33.46/.9074	30.58/.8412	29.37/.8744	34.08/.9399
ECO	**33.49/.9078**	**30.60/.8416**	**29.38/.8748**	**34.16/.9403**
(d) $\times 8$ SR on EDSR-baseline				
Vanilla	26.88/.7712	24.85/.6370	22.30/.6089	24.34/.7696
ECO*	26.90/.7700	24.91/.6378	22.37/.6091	24.40/.7697
ECO	**27.00/.7743**	**24.94/.6398**	**22.41/.6132**	**24.52/.7749**

Table 2: ECO (ours) compared to vanilla training. PSNR (dB) and SSIM scores are reported, and the best and second-best results are highlighted in **bold** and underlines. ECO* indicates training only up to 20% of the total iterations.

7 Related Work

Starting with the pioneering work (**?**), CNN base networks (**??????**) aiming for high fidelity reconstruction has shown drastic development. Later, ViT and Swin-based networks (**????**) have achieved the state-of-the-art performances revealing the effectiveness of self-attention in context of image reconstruction. Several works investigate the objective function of SISR (**??**) and empirical results of (**?**) demonstrate that the L_1 loss can lead to better convergence against the widely used L_2 loss. Knowledge distillation methods (**????**) have shown their efficiency on small SR networks where (**?**) uses privileged information to boost the teacher network's performance. Meanwhile, (**???**) aims to model the complex degradations explicitly. To tackle the ill-posed nature of SISR, several methods (**???**) obtain enhanced visual quality by utilizing the adversarial loss and the perceptual loss (**?**). Further, (**?**) generates adaptive targets, and (**??**) enables the generation of multiple plausible SR samples.

8 Limitation

It should be noted that Eq.(**??**) cannot disentangle the inherent noise term and the estimation error term. Thus, it reintroduces the inherent noise back into the training in later steps. Despite this, experiments emphasize the critical role of stability during the initial steps, setting a strong foundation that leads to overall performance gains. However, we acknowledge the opportunity for further advancement especially for the later training steps, which we leave for future work.

9 Conclusion

In this work, we have analyzed the underlying components of vanilla training and systematically developed the current training process. As a first step, we have disentangled the original loss function into two fundamental components; the centroid and the noise term. It turns out that the inherent noise term, induced by the ill-posed nature, can potentially raise additional difficulty in vanilla training. To overcome this issue, we estimate the centroid of all possible high-resolution images and obtain a noise-free lower bound of the original loss function which leads to a well-behaving optimization landscape with enhanced Lipschitzness. We further provide an effective method to overcome the limitation of estimation errors, which can be simply adapted into current methods within a few lines of code. Experimental results lead us to conclude that the proposed training framework can indeed lead to favorable results.

Acknowledgments

This work was supported in part by MSIT/IITP (No. 2022-0-00680, 2019-0-00421, 2020-0-01821, 2021-0-02068), and MSIT&KNPA/KIPoT (Police Lab 2.0, No. 210121M06).

Supplementary Material

Noise-free Optimization in Early Training Steps for Image Super-Resolution

MinKyu Lee, Jae-Pil Heo*
Sungkyunkwan University
{bluelati98, jaepilheo}@skku.edu

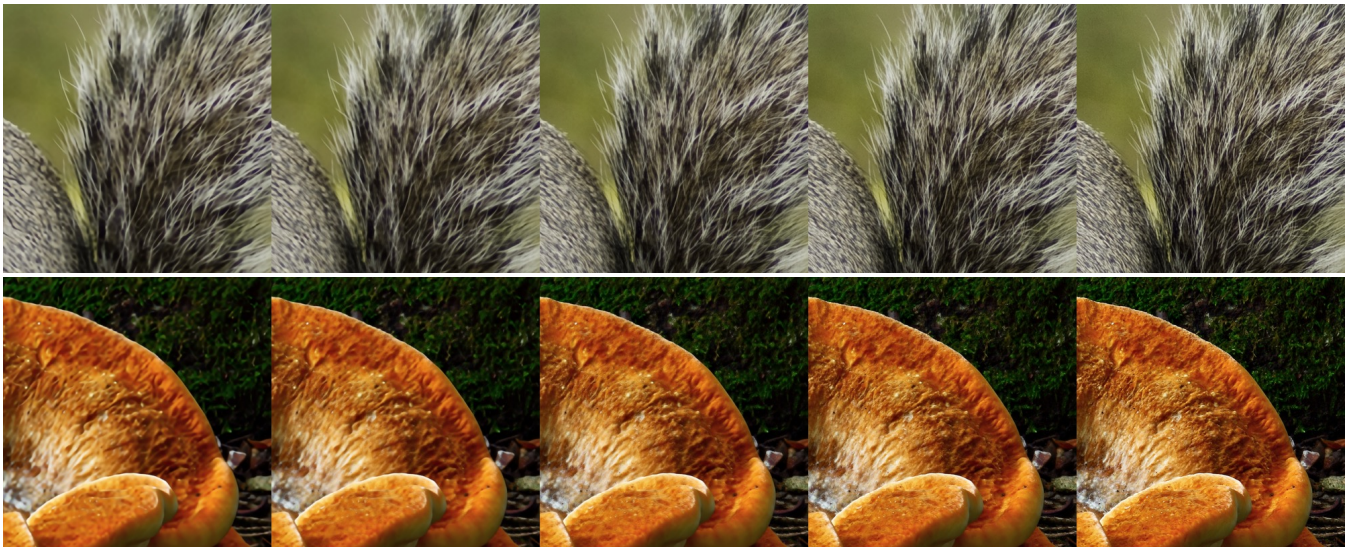

Figure 7: Visualization of target images in Eq.(**??**) as α gradually changes. Unpredictable high-frequency components that can lead to unstable optimization are removed when $\alpha = 0$. **Zoom in for best view.**

A Visual Examples of Target Images

In order to perform noise-free training, the proposed training framework utilizes different target images as training proceeds. Specifically, the HR image and the SR image of a pretrained network are blended based on a scheduling parameter α. In Fig.**??**, we provide visual examples of the target images in Eq.(**??**) as α gradually changes from 0 to 1. As α increases, images become sharper but contain unpredictable high-frequency components, which can potentially lead to noisy and unstable training.

B Regressing the Inherent Noise

To determine the inherent noise ϵ^* of an HR image, a naive approach might involve training a network to regress the error term. Here we compare this naive approach of regressing the error against the proposed method ECO. The key distinction lies in the way each method approximates the noise. Notably, the consequence of the regression is approximating the *expectation* of the error $\mathbb{E}(\epsilon^*)$, given an **LR**. In contrast, ECO estimates ϵ^* directly, by utilizing **HR** at training time. In Fig.**??**, we visualize estimated $\mathbb{E}(\epsilon^*)$ and ϵ^*. Here, $\mathbb{E}(\epsilon^*)$ is obtained by training an RRDB that is trained to regress ϵ^* instead of the SR image. It can be observed that $\mathbb{E}(\epsilon^*)$ results in a flat uncertainty map over the entire uncertain region. In contrast, ϵ^* better spots fine-grained noise factors, including almost invisible noise factors in the flat background. Note that we have shown how this noise can harm the training, underscoring the critical need for precise per-instance estimation of ϵ^*. **In a practical view**, estimating $\mathbb{E}(\epsilon^*)$ leads to significantly increased computational cost during training since it requires an additional network, whereas ECO only requires negligible cost. Specifically, the pretrained network $\hat{f}$ can be any off-the-shelf SR network for practical use cases, and μ_{emp} can be preprocessed before the actual training.

*Corresponding author

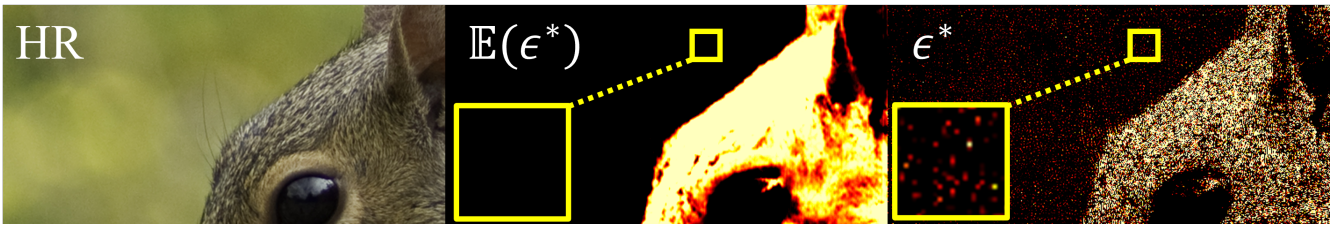

Figure 8: Visualization of estimated $\mathbb{E}(\epsilon^*)$ and ϵ^*. It can be seen that ϵ^*, which corresponds to the proposed method ECO, can better spot fine-grained noise factors including almost invisible noise in the flat background region. Values are scaled for better visualization. **Zoom in for best view**.

C Analysis in the Spectral Domain

We provide further analysis to identify the specific components of image instances that affect the optimization procedure. To achieve this, we applied Fast Fourier Transform (FFT) directly followed by inverse FFT (IFFT) to the images before feeding them into the super-resolution network. Fig.**??**.(a-b) illustrates HR, LR image pairs in the spectral domain and the gradients of losses in the spectral domain, both at $\alpha = 0$. Here, high activation on specific frequency regions indicates that the corresponding components are responsible for the loss values. In the case of vanilla training, the gradients exhibit strong activation, particularly on the regions where *very* high-frequency components exist. On the other hand, in ECO, while it is well activated on the major frequency components required for recovery, it shows relatively lower activation on very high-frequency components. We interpret this as indicating the presence of inherent noise terms in the frequency domain. By this observation, we conclude that ECO, indeed, has a powerful impact on the gradients, especially in regions where inherent noise terms are expected to reside.

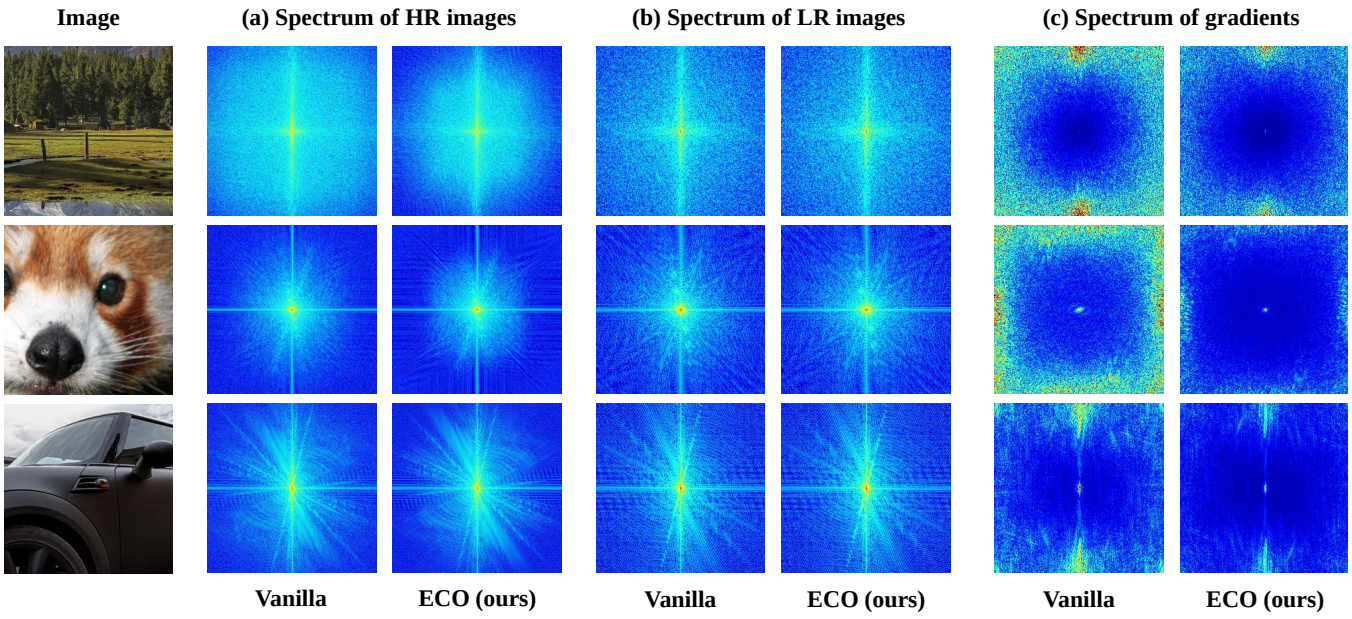

Figure 9: Spectral analysis of training data pairs and gradients at $\alpha = 0$. (a) High-resolution images in the spectral domain. (b) Low-resolution images in the spectral domain. (c) The gradient of the loss in the spectral domain.

D Experimental Details

Dataset details. For all datasets, we generate LR images with the bicubic function in MATLAB with the antialiasing option set as true. We have verified that all LR and HR images match the images provided in the prior work (**?**). In the case of real-world SR, we add zero-mean color Gaussian noise with $\sigma = 0.01$ to synthesize real-world images on flight. HR images were cropped modulo the scale factor in order to ensure that HR images match the output size of SR images.

Evaluation details. We have compared the proposed training scheme against vanilla training over standard benchmark datasets: Set5 (**?**), Set14 (**?**), BSD100 (**?**), Urban100 (**?**) and Manga109 (**?**). EDSR (**?**), RCAN (**?**) and SwinIR (**?**) were used

as the representative baseline methods. Performances were evaluated in terms of PSNR and SSIM indices on the Y channel (luminance channel) in the YCbCr space and pixels up to the scale factors in the border were ignored. In the case of real-world SR, we provide average results of 10 different evaluation runs, where the test images were preprocessed for fair comparison.

Implementation details. For all experiments, we reproduce representative baseline networks EDSR-baseline, EDSR, RCAN and SwinIR with both vanilla training and our method. To train the networks, we use 800 RGB images from DIV2K (**?**) and images were preprocessed as sub-patches for faster I/O. Note that we have only used the DIV2K dataset (instead of the DF2K) also for SwinIR, and do not use exponential moving averaged weights for both the baseline method and our method. The patch size of low-resolution images was kept as 48×48 for all scale factors as in prior works. Random horizontal and vertical flips were used, together with 90°, 180°, 270° random rotations as basic training augmentation. In Table.(1), we follow (**?**) which demonstrated comparable performance while significantly reducing the required training time. We increase the learning rate by $\times16$ and the mini-batch size by $\times8$, decreasing the total training iteration by $\times8$. Specifically, the mini-batch size is 128, the learning rate is 0.0016, and the total training iteration is 125K. We also substitute the scheduler as cosine annealing (**?**) and utilize the Lamb (**?**) optimizer which is known to work better on larger batch sizes. We train our networks on two NVIDIA TITAN RTX GPUs and the batch size was selected to fit GPU memory. We train networks from scratch for $\times2$ SR. For $\times3$ SR and $\times4$ SR, we start from the pretrained weight of the $\times2$ SR network. However, for $\times4$ RCAN (both vanilla training and ours), we kept the setting of the original works (**?**) since the baseline models produced undefined numbers (NaN) with larger learning rates. Specifically, the mini-batch size is 16, the learning rate is 10^{-4}, the total training iteration is 1000K and the Adam optimizer was used and the learning rate was reduced by half every 200K iterations. All networks in Tab.(1) were trained with two NVIDIA A6000 and all other networks were trained with one NVIDIA TITAN RTX.