Revisiting Implicit Neural Representations in Low-Level Vision

Wentian Xu, Jianbo Jiao
School of Computer Science, University of Birmingham
{wxx155@alumni., j.jiao@}bham.ac.uk

Abstract

Implicit Neural Representation (INR) has been emerging in computer vision in recent years. It has been shown to be effective in parameterising continuous signals such as dense 3D models from discrete image data, e.g. the neural radius field (NeRF). However, INR is under-explored in 2D image processing tasks. Considering the basic definition and the structure of INR, we are interested in its effectiveness in low-level vision problems such as image restoration. In this work, we revisit INR and investigate its application in low-level image restoration tasks including image denoising, super-resolution, inpainting, and deblurring. Extensive experimental evaluations suggest the superior performance of INR in several low-level vision tasks with limited resources, outperforming its counterparts by over 2dB. Code and models are available at: https://github.com/WenTXuL/LINR.

1 Introduction

Implicit neural representation (INR) has been emerging in recent years. INR is a special type of continuous signal representation. It uses neural networks (typically multilayer perceptron, i.e. MLP) to parameterise complex signals such as 3D shapes as in NeRF (Mildenhall et al., 2021). This representation has good storage efficiency and is widely used in 3D reconstruction because it can represent continuous output signals from discrete forms. Most of existing research on INR has been focused on 3D tasks, while it is under-explored in 2D low-level vision problems. In this paper, we are interested in the question: *How does INR perform in low-level vision?*

Low-level vision has always been a fundamental problem in computer vision. It mainly focuses on pixel-level problems such as image denoising, super-resolution, inpainting, and deblurring, to name a few. These image restoration tasks aim to obtain high-quality images from their corrupted versions. With the development of computational resources, deep learning has been shown to be effective in addressing these tasks (Zhang et al., 2017; 2018; Lai et al., 2017; Yu et al., 2018), while it usually requires large amounts of data with paired clean images for training the deep model. However, it is often difficult to obtain large-scale data with ground-truth clean images, and such methods tend to result in poor performance when the distribution of training data differs from that of the test data.

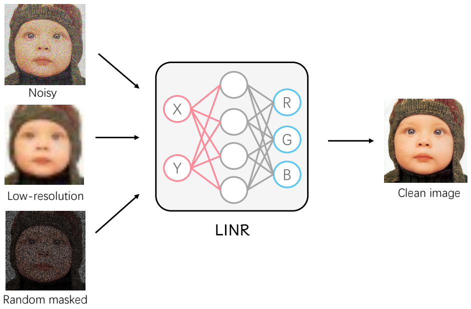


Figure 1: The proposed *LINR* aims to restore the original latent clean image (in terms of the RGB values of each pixel) from different types of corruptions, by only looking at the coordinates of the pixels within the input images.

Therefore, image restoration based on a single image, i.e. obtaining a clean image from only one corrupted image without the ground-truth clean image, has become a promising solution in this field. Deep image prior (DIP) (Ulyanov et al., 2018) investigates this solution by showing the potential of deep convolutional networks in addressing image restoration tasks without dataset-scale training. Some denoising works (Krull et al., 2019; Lehtinen et al., 2018) also get rid of the ground-truth clean image and only need noisy images, but they still need a dataset to train the model. Self2Self (Quan et al., 2020) follows these works by revising the network with dropout and mask, obtaining a high-quality clean image from a single noisy image. Zero-shot super-resolution (ZSSR) (Shocher et al., 2018) accomplishes super-resolution tasks with only one low-resolution image based on a fully convolutional network.

All these methods follow the conventional convolutional neural architecture, while the MLP (multi-layer perceptron) architecture that supports the INR is under-explored for low-level vision tasks. As a result, we are interested in its capability and performance in addressing low-level tasks, when given only one single image, similar to the aforementioned setting.

In our work, we propose to investigate the effectiveness of INR in low-level vision problems (*LINR*). We show the advantages of *LINR* on four low-level image restoration tasks, including image denoising, super-resolution, inpainting, and deblurring. When given the same amount of limited resources, *LINR* shows significantly better performance than other alternative approaches, validating the efficiency and effectiveness of the INR-based image restoration approach.

We further study the case where there are multiple corruptions by joint training the INR-based method. Surprisingly, *LINR* shows much better performance than training with only one corruption, suggesting its effectiveness in extracting beneficial features from both corruptions during joint optimisation. The main contributions of this work are summarised below:

- By revisiting INR in low-level vision tasks, we show the value and significant advantages of INR for image restoration tasks with limited resources.
- The proposed *LINR* is not restricted to one particular task, but generalises well to several different low-level vision tasks.
- We show that INR-based approaches can significantly benefit from joint training with multiple corruptions. This is a bit counter-intuitive as conventional image restoration methods are usually confused given multiple types of corruption.

2 Related Work

Implicit neural representation. Implicit neural representation (INR) has seen significant research interest in recent years. It focuses on parameterising a continuous differentiable signal with a neural network. This idea has been widely applied in 3D reconstruction tasks (Mildenhall et al., 2021; Mescheder et al., 2019; Park et al., 2019; Saito et al., 2019; Sitzmann et al., 2019; Oechsle et al., 2019; Michalkiewicz et al., 2019). Whereas there is relatively fewer INR work on 2D image tasks, mainly focusing on image generation (Dupont et al., 2021; Shaham et al., 2021; Skorokhodov et al., 2021; Anokhin et al., 2021), and super-resolution (Chen et al., 2021; Xu et al., 2021). For super-resolution tasks, We want to highlight that these methods require a large amount of data for training, whereas our *LINR* is concerned with cases where additional data is not required. In addition, the traditional MLP with the ReLU activation function (Nair & Hinton, 2010) is difficult to parameterise image signals. Sitzmann et al. (2020) proposed SIREN to solve this problem.

Image restoration and Zero-shot image restoration. As a low-level vision task, image restoration aims at obtaining a high-quality image from a corrupted image. Our work focuses on denoising, super-resolution, inpainting and deblurring. Most of the existing methods (Lehtinen et al., 2018; Krull et al., 2019; Batson & Royer, 2019; Lai et al., 2017; Shocher et al., 2018; Pathak et al., 2016; Gao & Grauman, 2017; Mataev et al., 2019) require large datasets for training, which in many cases are difficult to obtain. Researchers therefore propose methods that do not require extra data, also known as zero-shot methods. Self2self (Quan et al., 2020) extends the idea of noise2void (Krull et al., 2019) by using masks and dropout to allow the model to learn enough information on a single noisy image. ZSSR (Shocher et al., 2018) can complete super-resolution tasks without any dataset. Most of these methods focus on a single task or a few tasks. However, deep image prior (DIP) (Ulyanov et al., 2018) shows that the convolutional generative model can be applied as a good image prior for most image restoration tasks without any additional training on data. DeepRED (Mataev et al., 2019) has improved the DIP method based on RED (Romano et al., 2017). Our research also focuses on this zero-shot case. All of the previously mentioned methods are based on convolutional kernels. They also rely on some special structures like skip-connections. We show that, without these structures, *LINR* also performs well in these tasks.

3 INR IN LOW-LEVEL IMAGE RESTORATION

Our work is based on implicit neural representation (INR). The aim of INR is to parameterise complex continuous signals by a neural network. In the case of a 2D colour image, INR is a mapping from the 2D coordinates system to the RGB value space: $f : \mathbb{R}^2 \rightarrow \mathbb{R}^3, f(x, y) = (r, g, b)$.

As opposed to conventional training strategies that rely on a large dataset, INR is able to accomplish image restoration tasks by parameterising image signals in only one image. For instance, when we train our *LINR* to represent noisy images, the noise will be inhibited in the early training iterations.

As shown in fig. 2, the blue curve is the PSNR between the model output and the clean image. The orange curve is the PSNR between the output and the noisy input image. We can see that this training process suppresses the learning of noise. Although we only use the noisy image for training, the output approaches the clean image in the early iterations. Continued training will result in noisier results, but due to suppression, it is hard for INR to learn the exact same input noisy image without enough capacity.

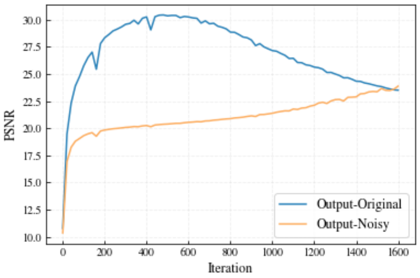


Figure 2: During training, the PSNR between the model output and the ground-truth clean image (the blue curve), and the PSNR between the output and the original noisy input (orange curve).

In addition, we have found that the process of training INR to represent images can also complement the missing information in corrupted images, such as low-resolution images. We attribute all the above findings to the assumption that when training on the image, INR applies implicit restrictions on the output, including local continuity, global consistency, smoothness tendency in early iteration etc. These properties also exist in natural clean images, e.g. neighbouring pixels often change smoothly. As a result, the output that follows these restrictions is close to high-quality clean images. These restrictions are embedded in the structure and training process of INR. For example, local continuity comes from continuous representation, part of global consistency comes from the sine activation function, and smoothness tendency comes from the training tendency of parameterised methods as these methods often tend to achieve smooth training at the beginning.

Based on this assumption, we propose the *LINR* and train on the corrupted image to achieve a high-quality clean target, which is achieved by the parameterising process. The implicit restriction hidden in this process will make sure the representation learned by *LINR* has the properties of a high-quality image while containing the information in the corrupted image. For different image restoration tasks, the only difference is the method of obtaining information from the corrupted image, i.e. the setting of the loss:

$$\min_{\theta} \left\| f_\theta(c) - x_0 \right\|^2 . \tag{1}$$

For the denoising task, as shown in eq. (1), the *LINR* is simply trained on the corrupted image with the L2 loss, and then the output is the clean image in an early iteration. For the super-resolution task, we assume that the INR represents a high-resolution image and has the loss function:

$$\min_{\theta} \left\| D(f_\theta(c)) - x_0 \right\|^2 , \tag{2}$$

where $D(\cdot)$ is the downsampling method to get the low-resolution result from the *LINR* output. Ideally, this low-resolution result should be similar to the original low-resolution image we actually have. In this case, the implicit restrictions of INR help with the reconstruction of details in the high-resolution image.

For the inpainting task, we assume that the *LINR* represents a high-quality image and has the loss function:

$$\min_{\theta} \left\| M(f_\theta(c)) - M(x_0) \right\|^2 , \tag{3}$$

where $M(\cdot)$ is the masking function. As shown in eq. (3), we just need to ensure that the unobscured part of the original image is similar to the INR result, i.e. only the pixels without masks will be used for training. As a result, the *LINR* training process will restore the obscured position.

The non-blind deblur task is similar to super-resolution and we have the loss function:

$$\min_{\theta} \|B(f_{\theta}(c)) - x_0\|^2 , \tag{4}$$

where $B(\cdot)$ is the blur kernel. The output of *LINR* is blurred using the blur kernel, and then the blurred image is forced to be close to the original corrupted image as defined in the loss eq. (4).

Overall, *LINR* is designed to represent and reconstruct high-quality clean images directly. Following that, its output is compared to the original corrupted image as a loss function to train the model.

4 Experiment

In this section, we evaluate and investigate the effectiveness of the proposed *LINR* over four low-level vision tasks, including image denoising, super-resolution, inpainting and deblurring. We also explore the performance of *LINR* when facing more different corruptions.

The implementation of *LINR* is based on SIREN (Sitzmann et al., 2020), which is a recent INR-based work replacing the activation function of the MLP with the sine function. This gives INR the ability to model information contained in higher-order derivatives, and the periodic feature of sine also provides SIREN with a degree of shift-invariance, which fits well with the properties of the image, as images often have repetitive information and feature. The width of the MLP model for all experiments is set to 256 and the depth to 6. The default learning rate is set to 10^{-4}. The experiment is conducted on an NVIDIA GPU A100.

The parameters are initialised following SIREN. By default, the training iteration is the same (500 unless otherwise specified) for all the deep-learning methods and images in each task, for a more stable and fair comparison and to show the performance of image restoration with limited resources. PSNR and SSIM are used for the evaluation metrics.

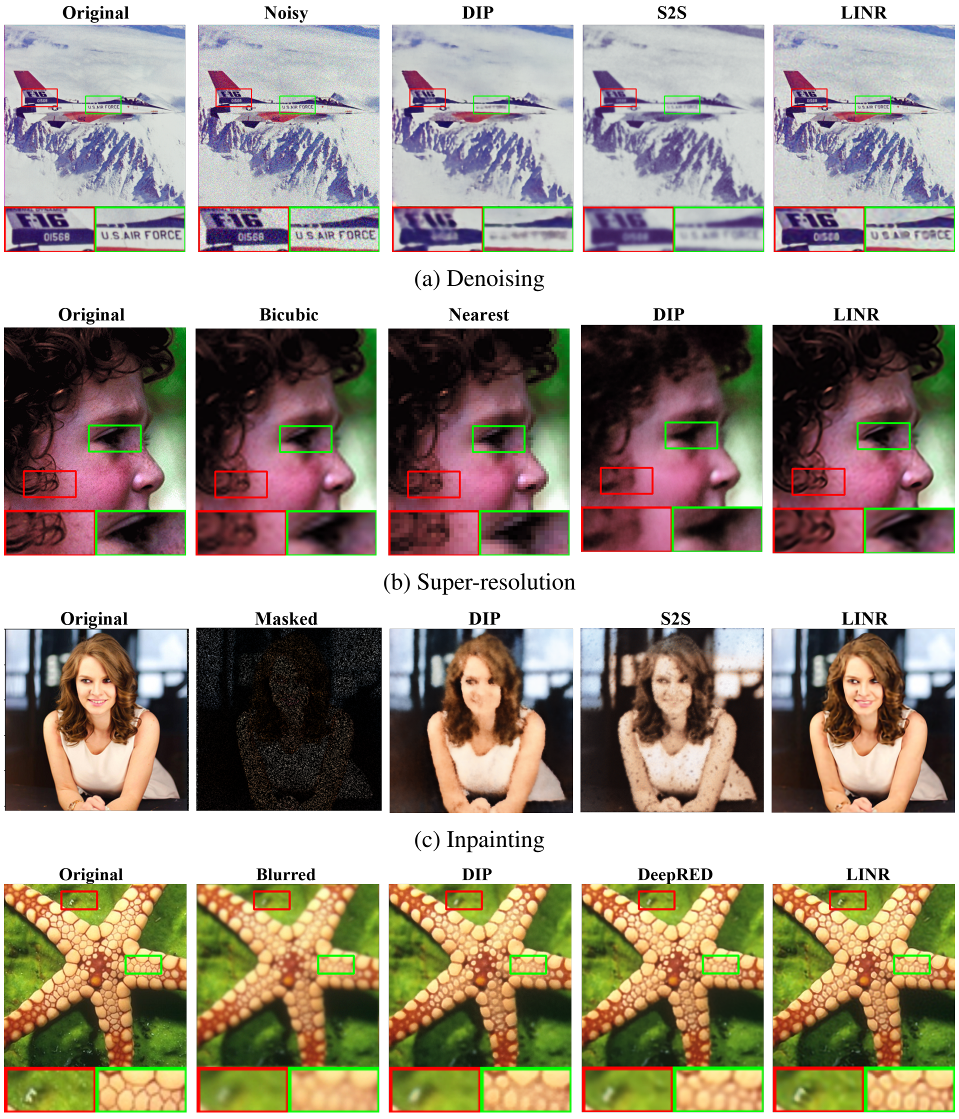
Table 1: Denoising performance (PSNR/SSIM) on Set9.

Images	DIP (Ulyanov et al., 2018)	S2S (Quan et al., 2020)	*LINR*(Ours)
Baboon	19.68/0.502	20.29/0.581	**23.67/0.794**
F16	25.27/0.871	24.14/0.880	**30.36/0.916**
House	**28.34/0.887**	24.31/0.851	23.10/0.580
Lena	26.72/0.851	25.31/0.861	**30.27/0.896**
Peppers	24.78/0.813	24.34/0.823	**29.46/0.860**
K01	22.19/0.624	22.40/0.711	**25.99/0.818**
K02	27.83/0.834	26.21/0.829	**29.66/0.866**
K03	27.63/0.869	24.77/0.860	**30.40/0.903**
K12	27.47/0.858	24.93/0.864	**30.34/0.892**
Average	25.55/0.790	24.08/0.807	**28.14/0.836**

4.1 Denoising

First, we show the performance of *LINR* on image denoising. We use the Set9 dataset as in (Ulyanov et al., 2018) and add Gaussian noise with $\sigma = 25$. The result is shown in table 1 and fig. 3a. It can be seen that the average results of *LINR* are significantly better than the other methods for both SSIM and PSNR. As mentioned before, *LINR* needs to stop early in the training period for denoising, and the best stopping position is different for different images. This is especially the case if the size of the image is different. However, the same stopping point is chosen for all images and methods for a fair comparison. So *LINR*'s results are poor on smaller-size images like the house (256*256). INR is so efficient that it learns to represent the clean version very quickly on these small images and then moves on to learn more noise information. The training time for LINR on a 256*256 image is

8.3s, and 14.3s for DIP as a comparison. For higher-resolution images, LINR will take more time as it is trained on all pixels with MLP.



(a) Denoising

(b) Super-resolution

(c) Inpainting

(d) Deblurring

Figure 3: Qualitative results of different tasks with comparison to state-of-the-arts.

4.2 SUPER-RESOLUTION

We then conduct experiments for image super-resolution on Set5 (Bevilacqua et al., 2012) with a sampling factor of 4. The downsampling method used in the loss function is lanczos2 (Turkowski, 1990). The results are shown in table 2 and fig. 3b, from which we can see that the *LINR* performs significantly better than the other methods both quantitatively and qualitatively.

4.3 INPAINTING

For the inpainting experiments, the mask is generated randomly. We show the results on sparsity 0.1. This means only ten percent of pixels remain after the masking. Again, *LINR* performs significantly better than the other methods, as shown in table 3 and fig. 3c.

Table 2: Super-resolution results on Set5 with factor = 4.

Images	Bicubic	Nearest	DIP (Ulyanov et al., 2018)	*LINR*
Baby	30.73/0.916	27.76/0.875	26.40/0.827	**31.20/0.921**
Bird	28.44/0.917	25.22/0.851	25.57/0.797	**30.48/0.944**
Butterfly	21.18/0.797	18.85/0.725	22.68/0.857	**24.29/0.898**
Head	29.08/0.841	27.79/0.811	26.58/0.768	**29.58/0.854**
Woman	25.39/0.885	22.90/0.827	24.28/0.837	**27.44/0.924**
Average	26.96/0.871	24.50/0.818	25.10/0.817	**28.60/0.908**

Table 3: Inpainting results on images with sparsity of 0.1 Bernoulli random masks.

Images	DIP (Ulyanov et al., 2018)	S2S (Quan et al., 2020)	*LINR*
Kate	25.87/0.893	22.60/0.841	**31.78/0.956**
Baby	25.99/0.826	24.36/0.802	**29.27/0.903**
Bird	24.88/0.800	21.78/0.736	**28.21/0.922**
Butterfly	20.59/0.827	18.04/0.706	**21.05/0.853**
Head	**26.53**/0.770	22.94/0.743	26.39/**0.775**
Woman	23.53/0.840	20.35/0.747	**24.41/0.892**
Average	24.57/0.826	21.68/0.763	**26.85/0.884**

4.4 Deblurring

For the deblurring task, we show the results for the Gaussian blur in table 4 and fig. 3d. A Gaussian kernel of width 25 and sigma 1.6 was chosen as the blur kernel. We can see that *LINR* performs better than the other methods. In this task, the iteration was set to 4000, as a lower number of iterations results in unreasonable results.

Table 4: Deblurring results (PSNR) on Gaussian blurred images ($\sigma = 1.6$, width = 25).

Images	DIP (Ulyanov et al., 2018)	DeepRED (Mataev et al., 2019)	*LINR*
Leaves	28.15	29.85	**30.10**
Parrot	30.03	31.30	**32.84**
Starfish	28.99	30.08	**31.78**
Butterfly	28.96	**30.15**	30.07
Average	29.03	30.35	**31.20**

4.5 Joint training

Here we explore the performance of *LINR* under multiple types of different corruption. The models are jointly trained with two corrupted images, which are derived from the same original image but with different corruption types. We found that *LINR* performs very well on joint training tasks. It can combine information from different corrupted images while not being misled by the differences between the corruptions. We tested joint training on super-resolution, denoising, and inpainting. The average PSNR performance of joint training with denoising and super-resolution tasks with limited resources is 30.98, outperforming DIP by a large margin (25.06). More importantly, it is significantly higher than training on a single corruption (Denoising: 27.63, SR: 28.60). Due to the space limit, detailed comparison results of other joint-training tasks are presented in the appendix A.

The core idea of image restoration based on INR is to guarantee the high quality of the represented image through implicit restrictions rather than learning the direct mapping between corrupted images and original images. This core process is independent of the corruption type. *LINR* is therefore not limited to one image restoration task and will not be misled by the differences between corruptions. More corrupted images will provide richer information to help *LINR* express high-quality images. Our finding shows the feasibility of learning under multiple different corruptions with INR.

5 Conclusion

In this paper, we revisited INR and explored its applications in several image restoration tasks. Under the same setting, it has been shown that the proposed *LINR* significantly outperforms other alternative solutions across all the evaluated tasks, suggesting the effectiveness of such simple representations in low-level tasks. Furthermore, we showcased the strong performance of *LINR* when given more realistic multiple corruptions, outperforming its single-corruption counterparts as well as other competitive solutions. We believe the findings derived from this study will potentially attract research interest in this new direction.

Similar to DIP (Ulyanov et al., 2018), one possible limitation of *LINR* is that the denoising process is easy to overfit. Therefore, the study of the stop criterion (e.g. (Jo et al., 2021)) could be a future direction worth investigating.

References

Ivan Anokhin, Kirill Demochkin, Taras Khakhulin, Gleb Sterkin, Victor Lempitsky, and Denis Korzhenkov. Image generators with conditionally-independent pixel synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14278–14287, 2021.

Joshua Batson and Loic Royer. Noise2self: Blind denoising by self-supervision. In *International Conference on Machine Learning*, pp. 524–533. PMLR, 2019.

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

Yinbo Chen, Sifei Liu, and Xiaolong Wang. Learning continuous image representation with local implicit image function. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8628–8638, 2021.

Emilien Dupont, Yee Whye Teh, and Arnaud Doucet. Generative models as distributions of functions. *arXiv preprint arXiv:2102.04776*, 2021.

Ruohan Gao and Kristen Grauman. On-demand learning for deep image restoration. In *Proceedings of the IEEE international conference on computer vision*, pp. 1086–1095, 2017.

Yeonsik Jo, Se Young Chun, and Jonghyun Choi. Rethinking deep image prior for denoising. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5087–5096, 2021.

Alexander Krull, Tim-Oliver Buchholz, and Florian Jug. Noise2void-learning denoising from single noisy images. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2129–2137, 2019.

Wei-Sheng Lai, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. Deep laplacian pyramid networks for fast and accurate super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 624–632, 2017.

Jaakko Lehtinen, Jacob Munkberg, Jon Hasselgren, Samuli Laine, Tero Karras, Miika Aittala, and Timo Aila. Noise2noise: Learning image restoration without clean data. *arXiv preprint arXiv:1803.04189*, 2018.

Gary Mataev, Peyman Milanfar, and Michael Elad. Deepred: Deep image prior powered by red. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pp. 0–0, 2019.

Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4460–4470, 2019.

Mateusz Michalkiewicz, Jhony K Pontes, Dominic Jack, Mahsa Baktashmotlagh, and Anders Eriksson. Implicit surface representations as layers in neural networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4743–4752, 2019.

Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021.

Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *Icml*, 2010.

Michael Oechsle, Lars Mescheder, Michael Niemeyer, Thilo Strauss, and Andreas Geiger. Texture fields: Learning texture representations in function space. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4531–4540, 2019.

Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 165–174, 2019.

Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2536–2544, 2016.

Yuhui Quan, Mingqin Chen, Tongyao Pang, and Hui Ji. Self2self with dropout: Learning self-supervised denoising from single image. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 1890–1898, 2020.

Yaniv Romano, Michael Elad, and Peyman Milanfar. The little engine that could: Regularization by denoising (red). *SIAM Journal on Imaging Sciences*, 10(4):1804–1844, 2017.

Shunsuke Saito, Zeng Huang, Ryota Natsume, Shigeo Morishima, Angjoo Kanazawa, and Hao Li. Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2304–2314, 2019.

Tamar Rott Shaham, Michaël Gharbi, Richard Zhang, Eli Shechtman, and Tomer Michaeli. Spatially-adaptive pixelwise networks for fast image translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14882–14891, 2021.

Assaf Shocher, Nadav Cohen, and Michal Irani. “zero-shot” super-resolution using deep internal learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3118–3126, 2018.

Vincent Sitzmann, Michael Zollhöfer, and Gordon Wetzstein. Scene representation networks: Continuous 3d-structure-aware neural scene representations. *Advances in Neural Information Processing Systems*, 32, 2019.

Vincent Sitzmann, Julien Martel, Alexander Bergman, David Lindell, and Gordon Wetzstein. Implicit neural representations with periodic activation functions. *Advances in Neural Information Processing Systems*, 33:7462–7473, 2020.

Ivan Skorokhodov, Savva Ignatyev, and Mohamed Elhoseiny. Adversarial generation of continuous images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10753–10764, 2021.

Ken Turkowski. Filters for common resampling tasks. *Graphics gems*, pp. 147–165, 1990.

Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Deep image prior. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 9446–9454, 2018.

Jun Xu, Hui Li, Zhetong Liang, David Zhang, and Lei Zhang. Real-world noisy image denoising: A new benchmark. *arXiv preprint arXiv:1804.02603*, 2018.

Xingqian Xu, Zhangyang Wang, and Humphrey Shi. Ultrasr: Spatial encoding is a missing key for implicit image function-based arbitrary-scale super-resolution. *arXiv preprint arXiv:2103.12716*, 2021.

Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. Generative image inpainting with contextual attention. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5505–5514, 2018.

Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE transactions on image processing*, 26 (7):3142–3155, 2017.

Kai Zhang, Wangmeng Zuo, and Lei Zhang. Ffdnet: Toward a fast and flexible solution for cnn-based image denoising. *IEEE Transactions on Image Processing*, 27(9):4608–4622, 2018.

A APPENDIX

A.1 TABLE OF JOINT-TRAINING

Here, We show the results of the combination of super-resolution and denoising in table 5, denoising and inpainting in table 6, and super-resolution and inpainting in table 7. In these settings, the performance of *LINR* is significantly better than its counterpart.

Furthermore, in table 8, table 9 and table 10, we compare joint training and training on only one corruption. It can be seen that joint training achieves significantly better performance than training on a single corruption across all three tasks, suggesting the effectiveness of the *LINR*.

For images containing noise in all joint-training experiments, the training weight is set to 0.1. This aims to reduce the misleading effect of this type of corruption. For all other images, the weight is 1.

Table 5: The restoring results of joint training on noisy and low-resolution images.

Images	DIP (Ulyanov et al., 2018)	*LINR*
Baby	25.96/0.823	**31.80/0.927**
Bird	25.74/0.804	**33.28/0.962**
Butterfly	23.32/0.873	**28.89/0.950**
Head	26.63/0.772	**30.07/0.864**
Woman	23.63/0.833	**30.86/0.955**
Average	25.06/0.821	**30.98/0.932**

Table 6: The restoring results of joint training on noisy and random masked (inpainting) images.

Images	DIP (Ulyanov et al., 2018)	*LINR*
Kate	25.93/0.893	**33.41/0.964**
F16	25.52/0.871	**31.08/0.942**
Peppers	25.01/0.825	**30.62/0.891**
House	28.29/**0.890**	**29.00**/0.843
Baboon	19.47/0.502	**24.05/0.813**
Lena	26.78/0.852	**31.49/0.919**
k01	22.03/0.624	**25.71/0.807**
k02	28.10/0.838	**29.66/0.871**
k03	27.48/0.869	**30.98/0.914**
k12	27.55/0.857	**30.55/0.896**
Average	25.62/0.802	**29.65/0.886**

Table 7: The restoring results of joint training on low-resolution and random masked (inpainting) images.

Images	DIP (Ulyanov et al., 2018)	*LINR*
Baby	25.83/0.821	**31.57/0.924**
Bird	25.02/0.797	**32.58/0.962**
Butterfly	23.14/0.869	**26.36/0.934**
Head	26.30/0.766	**29.71/0.861**
Woman	24.33/0.847	**29.47/0.948**
Average	24.92/0.820	**29.94/0.926**

Table 8: The *LINR* joint results where both noisy and low-resolution images are used as input and the *LINR* restoring results where only one of them is used.

Images	Denoising	SR	Together
Baby	30.58/0.915	31.20/0.921	**31.80/0.927**
Parrot	27.14/0.796	30.48/0.944	**33.28/0.962**
Butterfly	24.82/0.821	24.29/0.898	**28.89/0.950**
Head	27.36/0.765	29.58/0.854	**30.07/0.864**
Women	28.24/0.868	27.44/0.924	**30.86/0.955**
Average	27.63/0.833	28.60/0.908	**30.98/0.932**

Table 9: The *LINR* joint training results where both noisy and random masked (inpainting) images are used as input and the *LINR* restoring results where only one of them is used.

Images	Denoising	Inpainting	Together
Kate	32.00/0.951	31.78/0.956	**33.41/0.964**
F16	30.36/0.916	26.00/0.901	**31.08/0.942**
Pepper	29.46/0.860	27.30/0.879	**30.62/0.891**
House	23.10/0.580	26.29/0.862	**29.00/0.843**
Baboon	23.67/0.794	17.43/0.508	**24.05/0.813**
Lena	30.27/0.896	27.84/0.887	**31.49/0.919**
k01	**25.99/0.818**	19.91/0.607	25.71/0.807
k02	29.66/0.866	28.31/0.857	**29.66/0.871**
k03	30.40/0.903	29.02/0.902	**30.98/0.914**
k12	30.34/0.892	28.34/0.880	**30.55/0.896**
Average	28.53/0.848	26.22/0.824	**29.65/0.886**

Table 10: The *LINR* joint results where both low-resolution and random masked (inpainting) images are used as input and the *LINR* restoring results where only one of them is used.

Images	Inpainting	SR	Together
Baby	29.27/0.903	31.20/0.921	**31.57/0.924**
Parrot	28.21/0.922	30.48/0.944	**32.58/0.962**
Butterfly	21.05/0.853	24.29/0.898	**26.36/0.934**
Head	26.39/0.775	29.58/0.854	**29.71/0.861**
Women	24.41/0.892	27.44/0.924	**29.47/0.948**
Average	25.87/0.869	28.60/0.908	**29.94/0.926**

A.2 An ablation study with different activation functions

Our work is based on SIREN (Sitzmann et al., 2020), which uses sine as the activation function. We also replace it with other activation functions for super-resolution tasks as an ablation study here. We have tried Tanh, Sigmoid, ReLU, etc. We have also tried using positional encoding with ReLU, like what is used in NeRF (Mildenhall et al., 2021). The result is shown in table 11. With other activation functions, INR performed poorly on SR with the same number of training iterations(500). SIREN can represent high-quality images in a very short time, whereas the other activation functions can not. If given more training iterations, the performance of ReLU-PE can increase significantly (with an average PSNR of 26.23 on Set5 after 2000 iterations), but is still much worse than the result of SIREN (because ReLU cannot model higher-order derivatives and also lack the ability to extract enough global information (Sitzmann et al., 2020).) However, it is significantly better than the result of Nearest (24.50), which shows that with other activation functions, INR also has the capacity to restore high-quality images.

Table 11: Super-resolution results of INR on Set5 with different activation functions.

	Sigmoid	Tanh	ReLU	SeLU	ReLU-PE	Sine (SIREN/*LINR*)
Set5	14.48/0.520	15.29/0.520	19.94/0.643	18.18/0.590	24.29/0.742	28.60/0.908

A.3 Super-resolution results with different factors

Table 12: Super-resolution results (PSNR) on Set5 with factor = 8.

	Bicubic	Nearest	DIP (Ulyanov et al., 2018)	*LINR*
Baby	26.14	24.03	24.53	**27.32**
Bird	23.30	21.48	22.30	**24.07**
Butterfly	16.85	15.67	18.16	**18.66**
Head	26.99	25.50	25.18	**27.48**
Woman	21.25	19.60	21.09	**22.43**
Average	22.91	21.26	22.25	**23.99**

Table 13: Super-resolution results (PSNR) on Set5 with factor = 2.

	Bicubic	Nearest	DIP (Ulyanov et al., 2018)	*LINR*
Baby	**36.02**	32.64	26.63	32.87
Bird	35.56	30.53	26.15	**36.86**
Butterfly	26.59	23.43	24.63	**28.82**
Head	**31.67**	30.54	27.18	31.08
Woman	31.34	27.75	25.56	**32.75**
Average	32.24	28.98	26.03	**32.48**

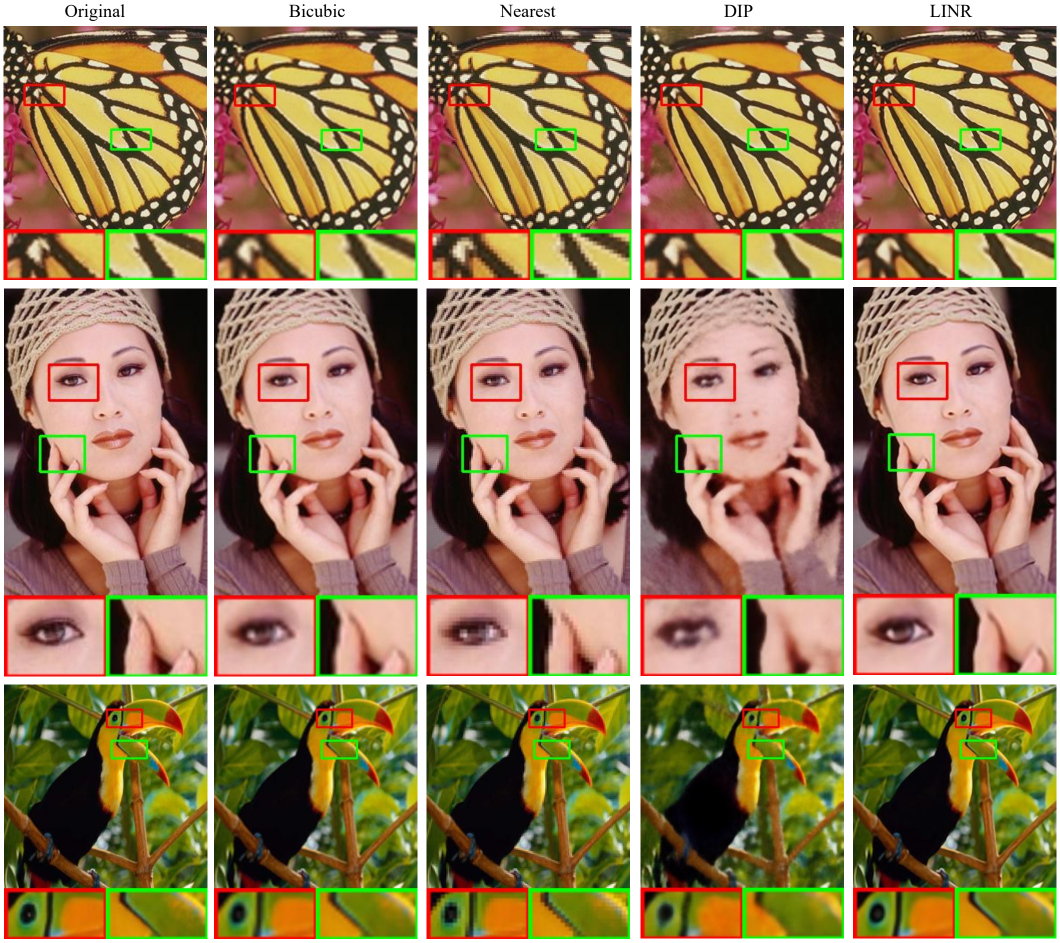


Figure 4: Super-resolution results on Set5 with factor = 2.

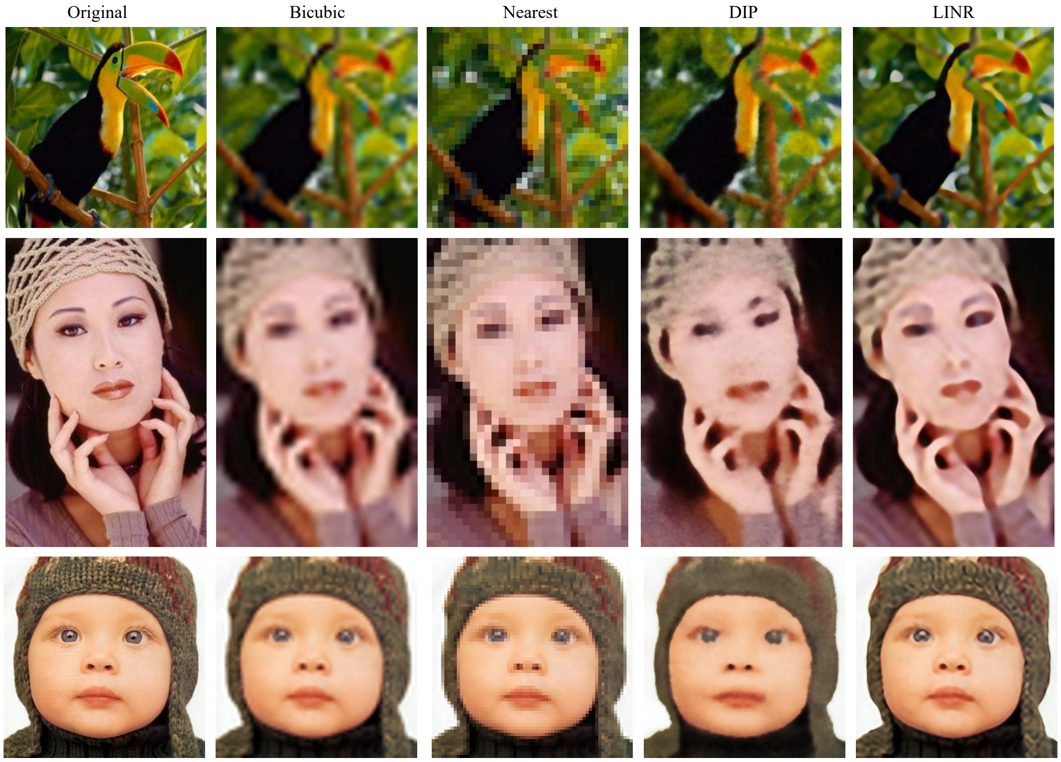


Figure 5: Super-resolution results on Set5 with factor = 8.

A.4 Qualitative results on joint training

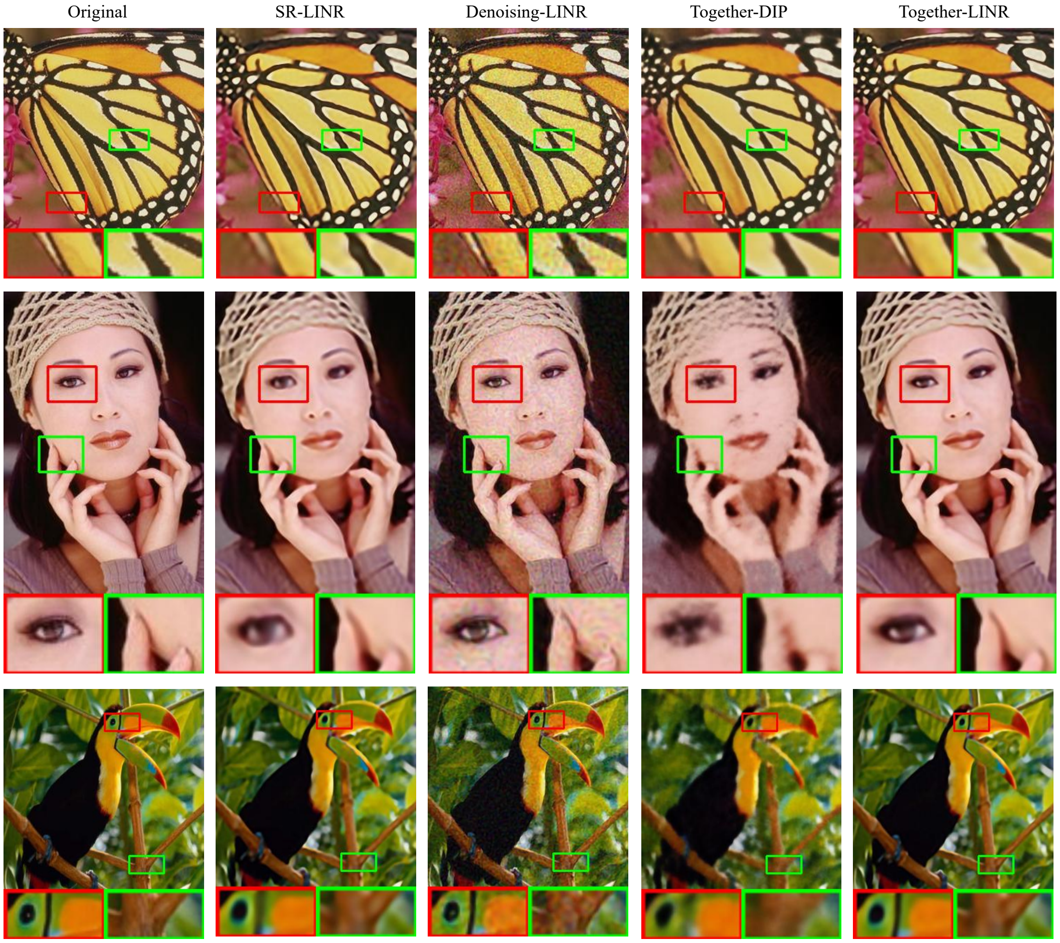


Figure 6: The joint results on *LINR* and DIP where both noisy and low-resolution images are used as input and the *LINR* restoring results where only one of them is used.

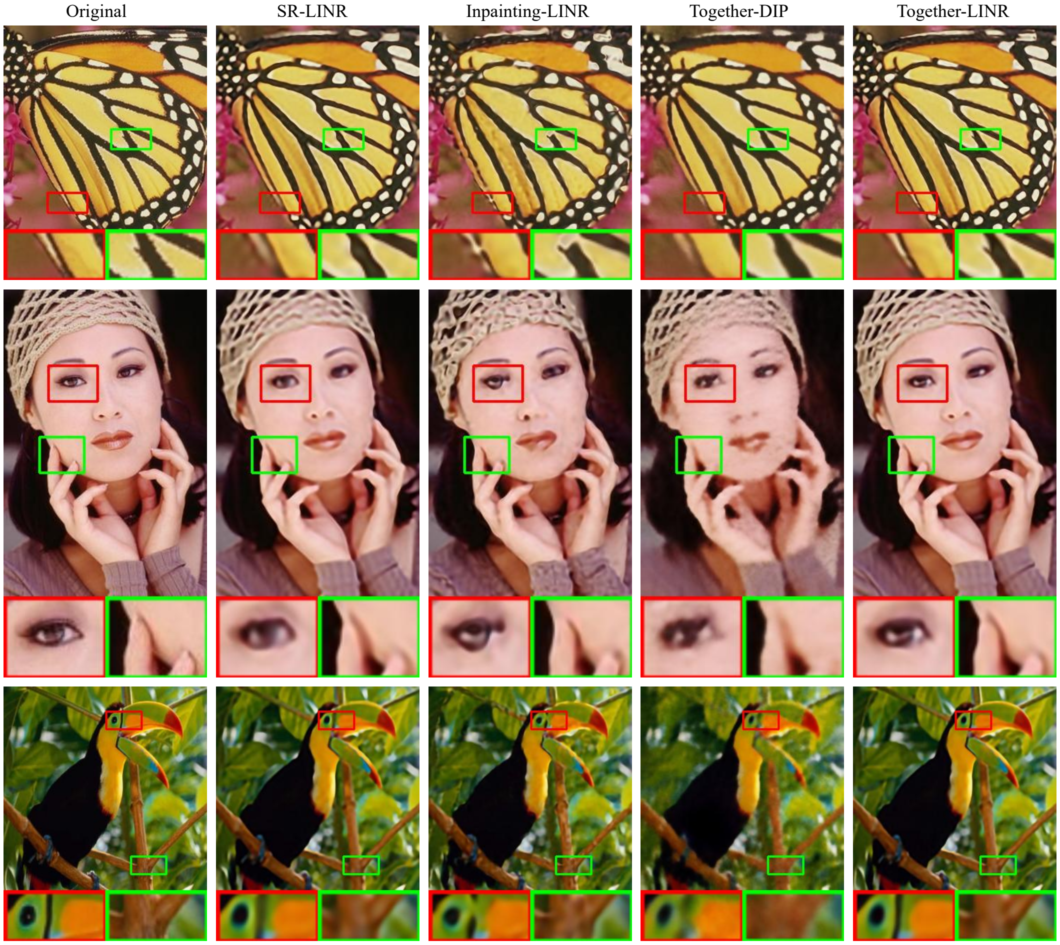


Figure 7: The joint results on *LINR* and DIP where both low-resolution and random masked (inpainting) images are used and the *LINR* restoring results where only one of them is used.

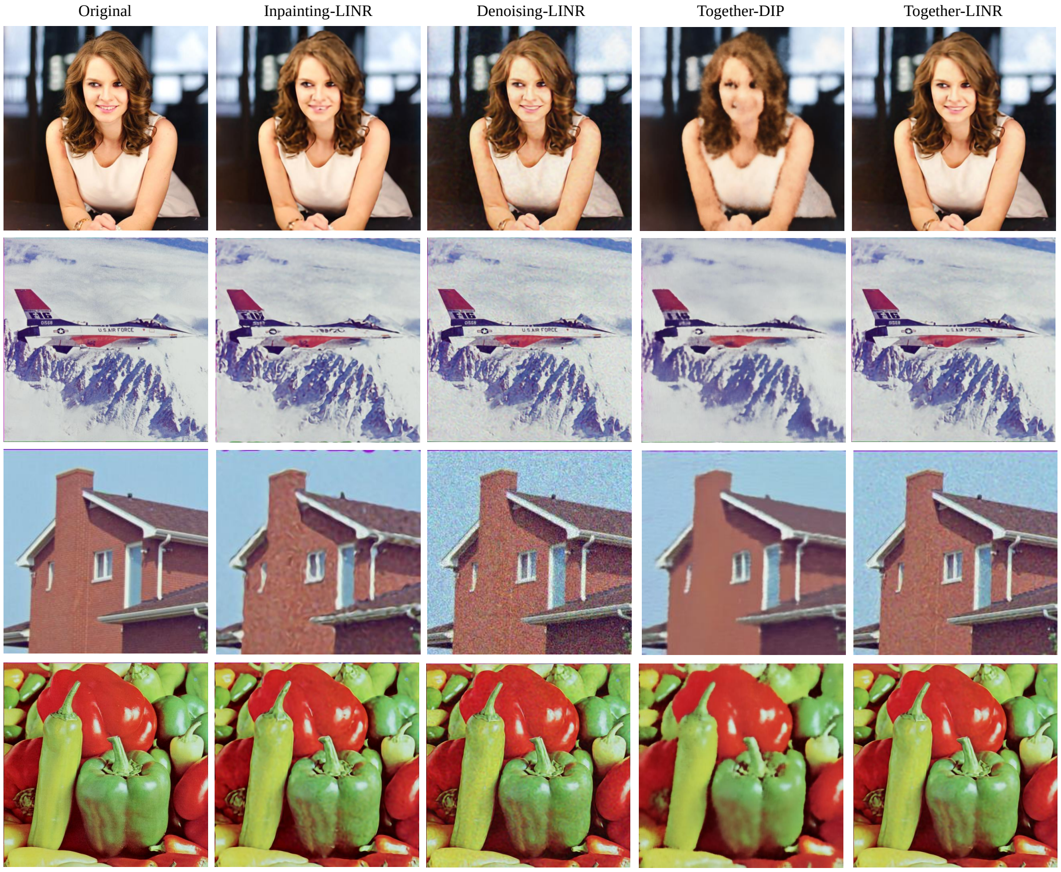


Figure 8: The joint results on *LINR* and DIP where both noisy and random masked (inpainting) images are used as input and the *LINR* restoring results where only one of them is used.

A.5 Qualitative results on inpainting

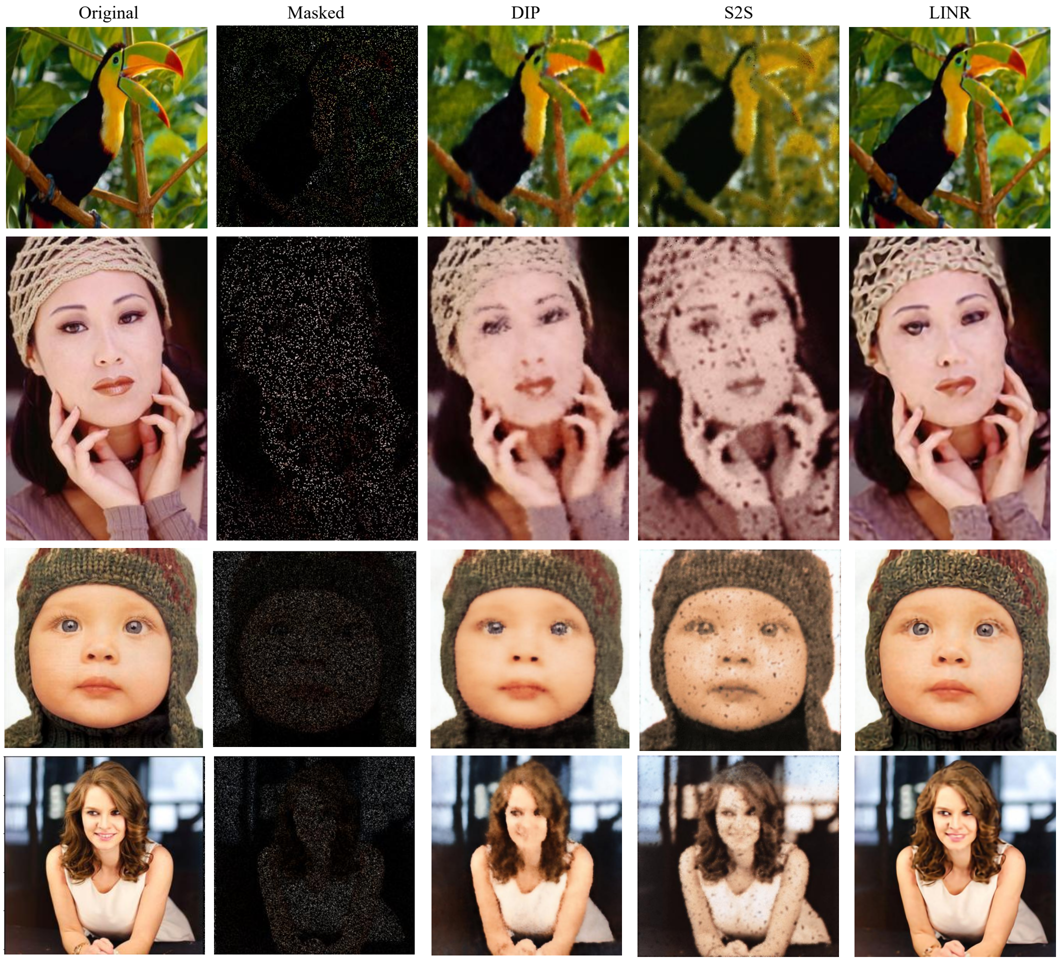


Figure 9: Visual results on random masked inpainting with sparsity = 0.1. Even if the mask obscures most of the pixel, *LINR* can still produce very high-quality results.

A.6 Qualitative results on super-resolution

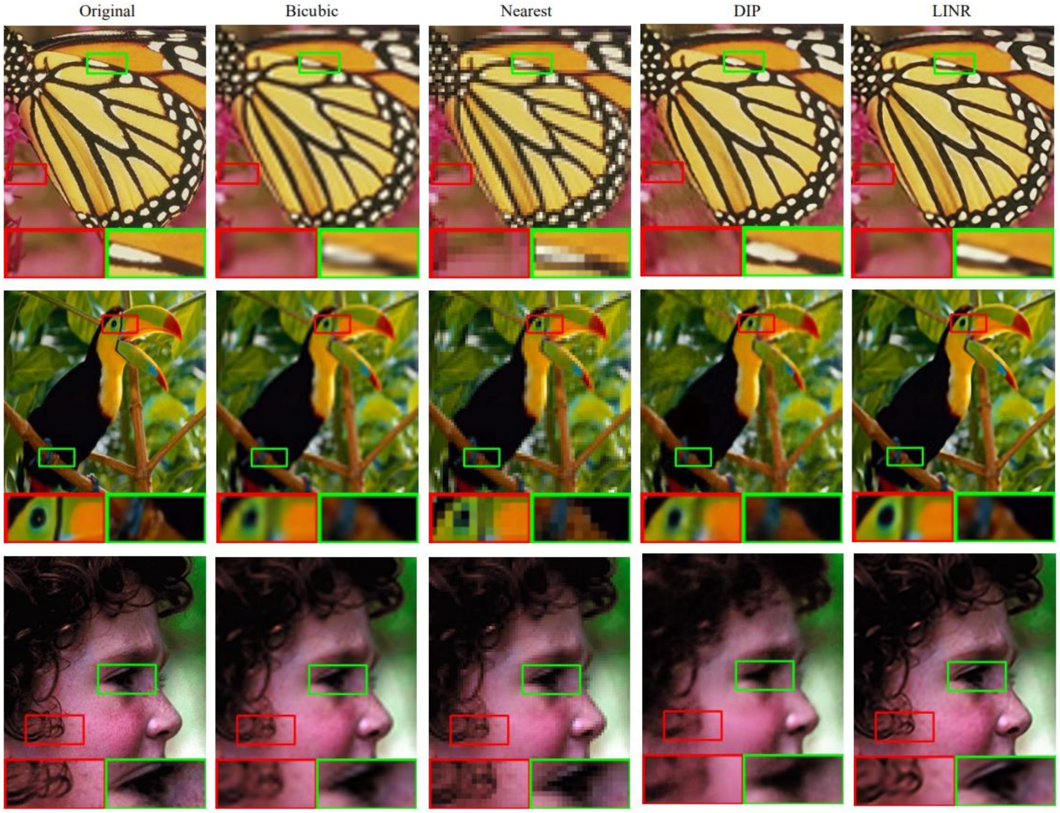


Figure 10: Qualitative results on SR with factor = 4. Compared to other methods, *LINR* results are clearer and contain more details with limited training resources.

A.7 QUALITATIVE RESULTS ON DEBLURRING

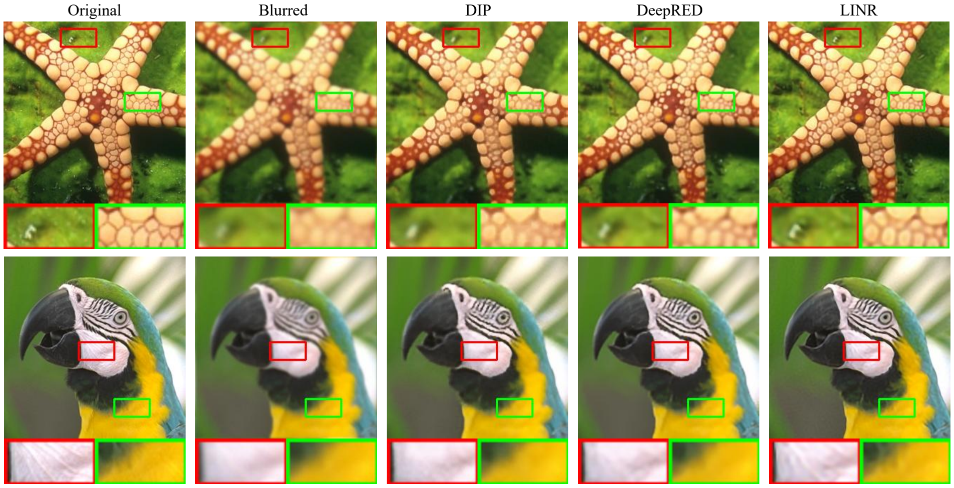


Figure 11: Qualitative results on Gaussian blurred images. We can see that the edges of the *LINR* are more significant and better recovered.

A.8 ADDITIONAL QUALITATIVE RESULTS AND REAL NOISY IMAGE DENOISING

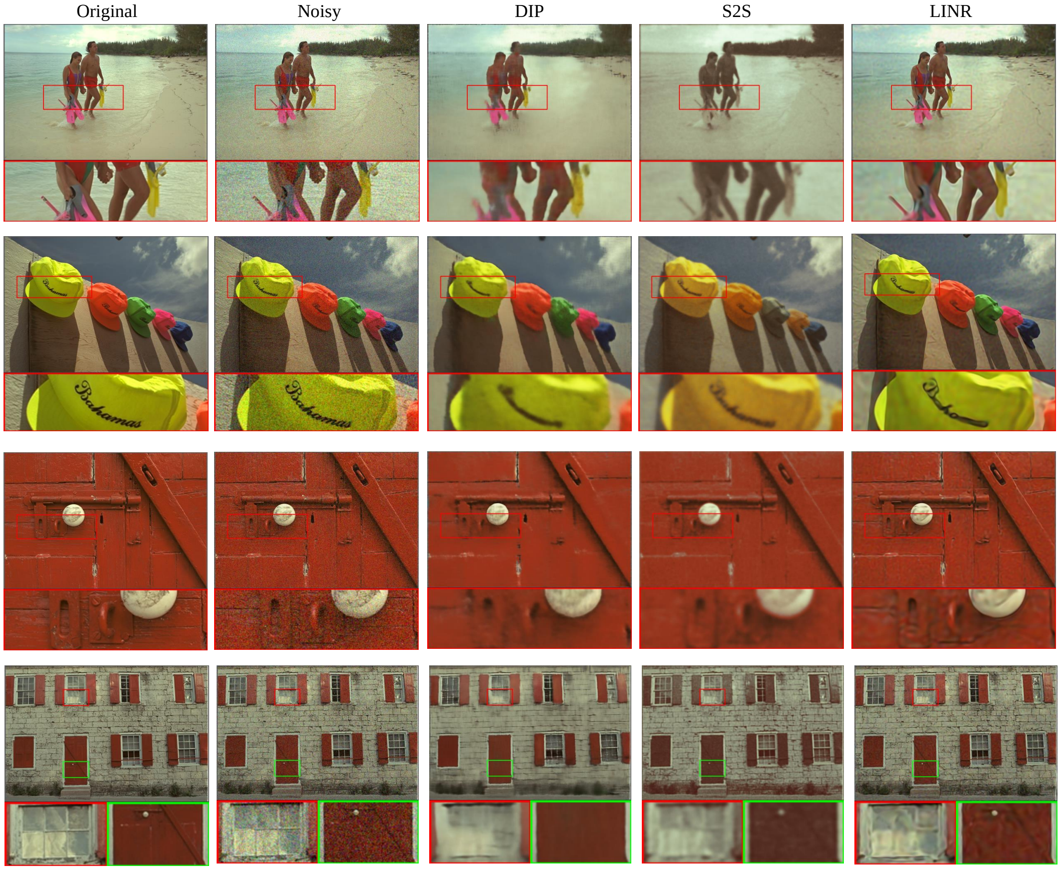


Figure 12: Qualitative results on denoising with $\sigma = 25$. *LINR*'s denoise results contain more detailed information than other methods.

LINR also works well on real-world noise. We tested the *LINR* on PolyU (Xu et al., 2018) real noisy images dataset. Due to the weak noise, we increased the training iteration for all methods to 1000. The result is shown in table 14 and fig. 13, from which we can see that the proposed *LINR* performs better than other methods. In the experiment, we additionally used the exponential sliding window to average the result on DIP, as mentioned in their original paper (Ulyanov et al., 2018), because, without this, DIP denoised images even perform worse (lower PSNR) than the original noisy images.

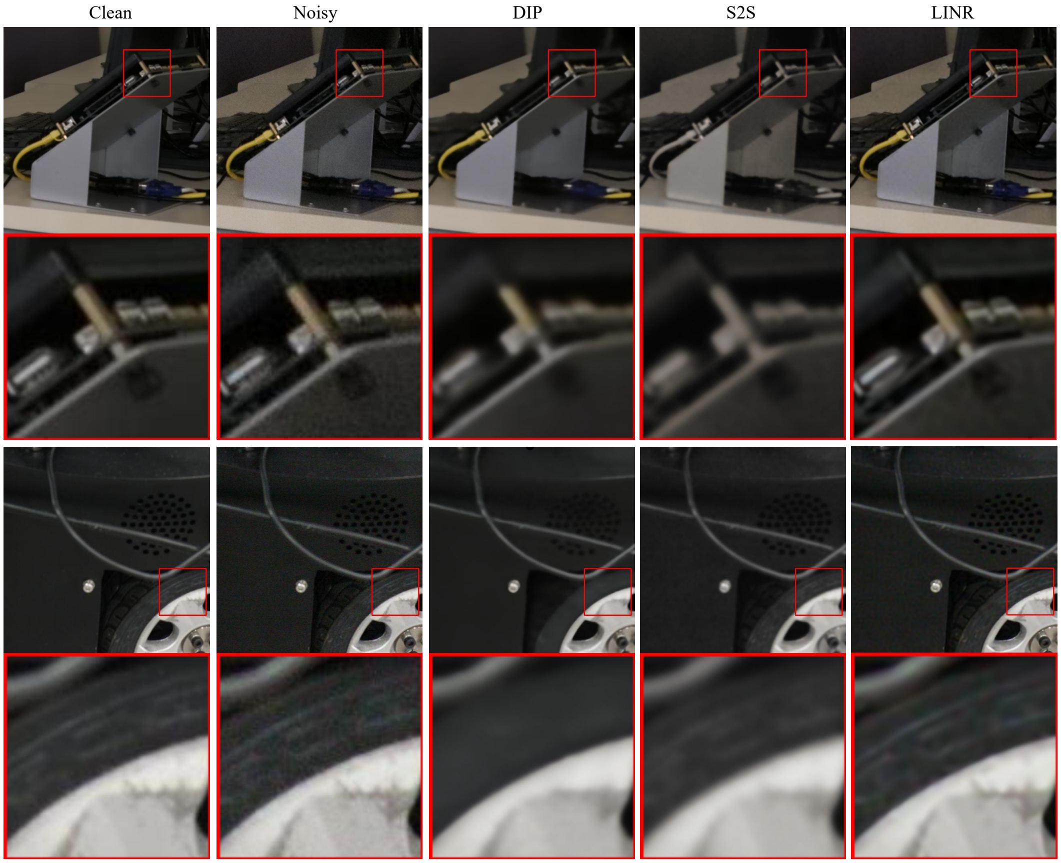


Figure 13: Qualitative results on PolyU real-world noise dataset.

Table 14: Average denoising performance (PSNR/SSIM) on real-world noisy images (PolyU).

Images	DIP (Ulyanov et al., 2018)	S2S (Quan et al., 2020)	*LINR*
PolyU	36.07/0.968	32.42/0.954	**37.12/0.979**