

GSDD: Generative Space Dataset Distillation for Image Super-resolution

Haiyu Zhang¹, Shaolin Su¹, Yu Zhu^{1*}, Jinqiu Sun², Yanning Zhang¹

¹School of Computer Science, Northwestern Polytechnical University

²School of Astronautics, Northwestern Polytechnical University
zhanghaiyu@mail.nwpu.edu.cn

Abstract

Single image super-resolution (SISR), especially in the real world, usually builds a large amount of LR-HR image pairs to learn representations that contain rich textural and structural information. However, relying on massive data for model training not only reduces training efficiency, but also causes heavy data storage burdens. In this paper, we attempt a pioneering study on dataset distillation (DD) for SISR problems to explore how data could be slimmed and compressed for the task. Unlike previous coreset selection methods which select a few typical examples directly from the original data, we remove the limitation that the selected data cannot be further edited, and propose to synthesize and optimize samples to preserve more task-useful representations. Concretely, by utilizing pre-trained GANs as a suitable approximation of realistic data distribution, we propose GSDD, which distills data in a latent generative space based on GAN-inversion techniques. By optimizing them to match with the practical data distribution in an informative feature space, the distilled data could then be synthesized. Experimental results demonstrate that when trained with our distilled data, GSDD can achieve comparable performance to the state-of-the-art (SOTA) SISR algorithms, while a nearly $\times 8$ increase in training efficiency and a saving of almost 93.2% data storage space can be realized. Further experiments on challenging real-world data also demonstrate the promising generalization ability of GSDD.

Introduction

Single image super-resolution (SISR) refers to recovering the high-resolution (HR) image from its low-resolution (LR) counterpart. With the increasing pursuit and preference for high-definition images, a rising requirement for the performance improvement of current SISR algorithms is urgently demanded. To this end, people attempt to build a massive amount of LR-HR image pairs to train SR models that cover rich textural and structural information existing in the real scenarios. However, one of the most prominent problems emerges accordingly is that its success was mainly derived at the cost of the massive data, which consumes huge computational and storage resources for model implementation. A feasible way to reduce the cost is to compact the data while preserving sufficient information for deep models. Thus, it

*Corresponding author.

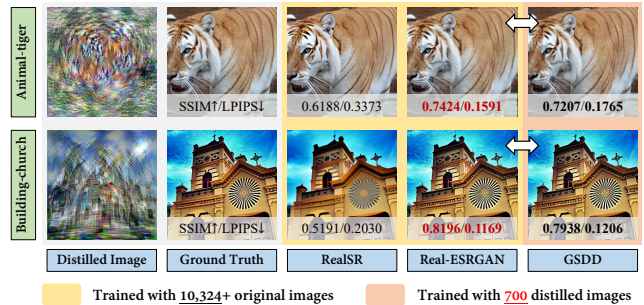


Figure 1: A demonstration of the corresponding distilled images and SR results of two example categories. Two advanced SISR models trained on more than 10,324 original images (yellow part) are compared. The proposed GSDD (based on Real-ESRGAN (Wang et al. 2021) generator) was only trained on 700 distilled images (red part), but achieved approximate results with the SOTA. The arrow here indicates comparable performance. Please zoom in for best view.

is natural to think of adopting the coreset selection (Phillips 2017) methods. By selecting a subset of data from the whole training data, models are expected to perform competitively with those trained on the whole population of data. Despite the straightforward methodology, it has two drawbacks: **1)** most coreset selections are developed as upstream tasks, they do not guarantee an optimal solution for downstream tasks (e.g., image restoration); **2)** the method simply selects samples from dataset, while the most informative data representation cannot be reached (Zhao and Bilén 2021).

To overcome above limitations, dataset distillation¹ (DD) was proposed to synthesize a bunch of compact data that has better informative property than the coreset (Wang et al. 2018a). This process can also be interpreted as learning a small set of representative images from a large amount of training data, while achieving similar generalization abilities but higher training efficiencies than models trained on the original data. Unlike coreset selection, the synthetic data are directly optimized for downstream tasks and thus leading to better testing performance (Zhao, Mopuri, and Bilén 2020).

¹Dataset distillation is also referred to as dataset condensation in some literatures (Zhao and Bilén 2021; Zhao, Mopuri, and Bilén 2020; Kim et al. 2022). The two are essentially the same.

Therefore, in this paper, to alleviate the massive data dependency for current SR models, we implement a pioneering study on DD for solving SISR problems by optimizing a compact image set in a latent generative space that fits best for training SISR models. Concretely, as pre-trained GANs (Goodfellow et al. 2020) can be interpreted as an approximation of complicated image distribution, we utilize this preferable property for realizing the distillation operations. By distilling data in a latent generative space based on GAN-inversion (Xia et al. 2023) techniques, we obtain a small set of compact but informative data, and models trained on them achieve comparable generalization ability with the ones trained on the original large-scale dataset. We also propose a regularization term \mathcal{R} on the basis of distribution matching loss to strengthen the informative representation of the latent vectors during the distillation process.

From Figure 1, it can be observed that GSDD obtains similar performances both quantitatively and qualitatively to Real-ESRGAN (Wang et al. 2021) (trained on more than 10,324 images) by only training on the distilled images, which is less than 7% of the original data amount. In summary, our contributions of this paper are as follows:

- ★ We propose to optimize and distill current SR dataset to form a compact and informative representation. To the best of our knowledge, it is the first attempt to explore the possibility of applying DD techniques to SISR research.
- ★ We establish both a GAN-inversion based scheme and a regularization term \mathcal{R} for data optimization and distillation operations. Our approach ensures the distilled data match the realistic data distribution while preserving representative information for fulfilling SISR tasks.
- ★ Extensive experimental results validate our effectiveness. We achieve comparable performance to SOTA SISR algorithms while obtain a nearly $\times 8$ increase in training efficiency and a saving of almost 93.2% data storage space. Furthermore, GSDD can generalize well even to the challenging real-world scenes images.

Related Works

Dataset Distillation (DD)

Compared to common knowledge distillation (also known as model distillation (Hinton, Vinyals, and Dean 2015)), DD (Wang et al. 2018a) distills the dataset rather than the model. Specifically, it synthesizes a small portion of data so that a model trained on these data maintains the performance of the model trained on the full dataset (Cazenavette et al. 2022). Over the past few years, it has drawn increasing attention in the field of machine learning (ML) (Nguyen, Chen, and Lee 2020; Nguyen et al. 2021; Zhao and Bilen 2021; Zhao, Mopuri, and Bilen 2020) including various applications, such as continual learning (CL) (Rebuffi et al. 2017; Deng and Russakovsky 2022), neural architecture search (NAS) (Such et al. 2020; Cui et al. 2022), federated learning (FL) (Liu, Yu, and Zhou 2022; Hu et al. 2022) and privacy-preserving ML (Dong, Zhao, and Lyu 2022), etc. Different from traditional data compression, DD aims to generate a small-scale

synthetic dataset that preserves adequate task-useful information so that the model trained on it can generalize well to other unseen data.

Single Image Super-Resolution (SISR)

In recent years, intensive research on deep neural networks (DNNs) has dramatically boosted the performance of many SISR models, as well as achieving SOTA results on various benchmarks. More specifically, from early approaches based on convolutional neural networks (CNNs) (e.g., SRCNN (Dong et al. 2014, 2016), VDSR (Kim, Lee, and Lee 2016), EDSR (Lim et al. 2017), etc.) to more recent promising methods using generative adversarial networks (GANs) (Goodfellow et al. 2020) (e.g., SRGAN (Ledig et al. 2017), ESRGAN (Wang et al. 2018c), RankSRGAN (Zhang et al. 2019), SFTGAN (Wang et al. 2018b), etc.), various deep learning strategies have been applied to this field. Meanwhile, datasets dedicated to SR models also have been proliferated. For example, DIV2K (800) (Agustsson and Timofte 2017) and Flickr2K (2,650) (Timofte et al. 2017) are often utilized for model training. In addition, merging multiple datasets for SR training process is also a popular trend, e.g., combining DIV2K and Flickr2K into DF2K (3,450) (Lim et al. 2017; Haris, Shakhnarovich, and Ukita 2021), combining DF2K and OST (Wang et al. 2018b) into DF2K_OST (13,744) (Wang et al. 2021) and so on.

Remarks

In order to learn representations that contain sufficiently detailed information, most current SISR models often depend on datasets containing a large amount of data samples. As a result, the burden of training efficiency and data storage become a non-negligible issue. Although there are studies on DD and its applications on high-level computer vision areas, few attention has been paid to low-level image restoration tasks, especially for image SR community. To promote the training efficiency as well as maintain the generalization capability of current SISR models, we propose GSDD, which explores DD techniques based on GAN-inversion operations for solving SISR problems. Our underlying hypothesis is that since existing GANs can well fit the large-scale image space to provide rich data information, conducting DD in the latent space of pre-trained GAN models can be regarded as an efficient and convenient solution. Since we are the first to apply the DD ideology to SISR tasks, we expect this to be a pioneering study that potentially facilitates the practical applications under model and data constraints.

Proposed Method

Figure 2 illustrates our framework, which consists of a data distillation phase and a model retraining & inference phase. Specifically, phase one distills and optimizes the latent vector set \mathcal{Z} by minimizing a distribution matching loss between the training set \mathcal{T} and synthetic set \mathcal{S} . Wherein, a pre-trained SISR generator combined with GAN-inversion operations is utilized to synthesize the small-scale set \mathcal{S} , serving as the distilled dataset. In phase two, the distilled image pairs are used to retrain existing SISR models from scratch and then test their restoration and generalization performances.

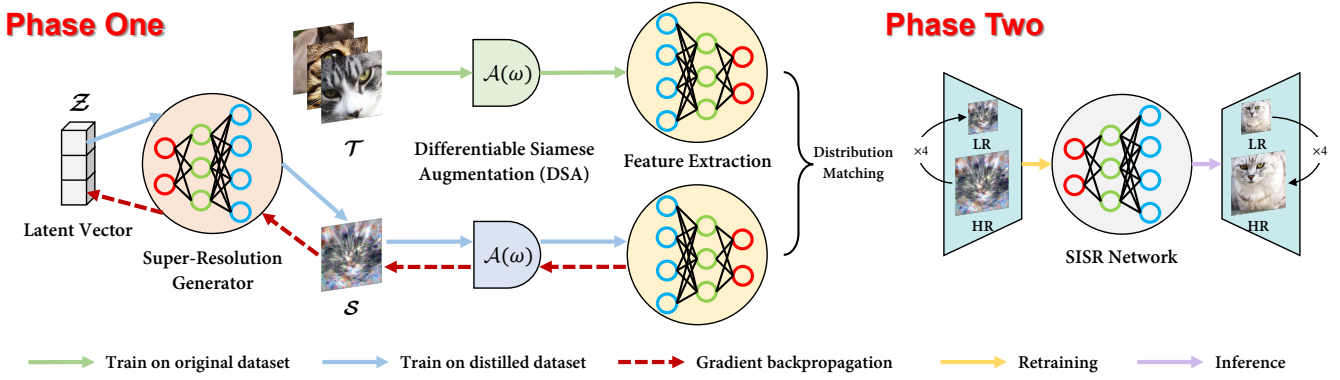


Figure 2: The flow diagram of the proposed GSDD model.

Problem Formulation

Given a large training dataset $\mathcal{T} = \{x_i, y_i\}_{i=1}^m$, DD aims at extracting the knowledge of \mathcal{T} into a small synthetic dataset $\mathcal{S} = \{s_j, y_j\}_{j=1}^n$ ($n \ll m$) so that the model $\theta^{\mathcal{S}}$ trained on \mathcal{S} can achieve comparable performance to the model $\theta^{\mathcal{T}}$ trained on \mathcal{T} (Lei and Tao 2024). This process can be formulated as Equation 1.

$$\mathbb{E}_{x \sim \mathcal{P}_{RDD}}[\ell(\phi_{\theta^{\mathcal{T}}}(x), y)] \simeq \mathbb{E}_{x \sim \mathcal{P}_{RDD}}[\ell(\phi_{\theta^{\mathcal{S}}}(x), y)] \quad (1)$$

Where x is the input data, y is the label, and \mathcal{P}_{RDD} represents the real data distribution. $\phi_{\theta^{\mathcal{T}}}$ and $\phi_{\theta^{\mathcal{S}}}$ denote DNNs parameterized by $\theta^{\mathcal{T}}$ and $\theta^{\mathcal{S}}$, respectively.

GAN-Inversion

Given a target image, GAN-inversion aims to map it back into the latent space of a pre-trained GAN model, so that the target image can be faithfully reconstructed from the feed-forward inverted code (Xia et al. 2023). In our practical approach, we employ the optimization-based (Creswell and Bharath 2019; Abdal, Qin, and Wonka 2019, 2020) inversion method, because it learns the latent vectors of each input image separately. For a pre-trained generator G , the latent vector for a real image is optimized by Equation 2.

$$\hat{z} = \arg \min_z \ell(G(z) - x) \quad (2)$$

Where ℓ calculates the loss in feature or pixel space, z denotes the latent vector, x represents the target image. Since GAN-inversion focuses on the manipulation of visual effects for a specific image, the learned synthetic images are not guaranteed to be informative enough for training a well-behaved DNN (Xia et al. 2023). Therefore, it is necessary to impose some constraints on the inversion process especially on the latent vector z . As shown in Equation 3, received the inspiration from latent space optimization (LSO) (Bojanowski et al. 2018), we use the latent space embedding (Abdal, Qin, and Wonka 2019), a simple but effective GAN-inversion method, to initialize and optimize the latent vectors via minimizing the feature and pixel distances between the synthetic images and training samples.

$$z' = \arg \min_z \mathcal{D}_f + \mathcal{D}_p \quad (3)$$

$$\mathcal{D}_f = \frac{1}{d_f} \|\psi_{\vartheta}(G(z)) - \psi_{\vartheta}(x)\|^2 \quad (4)$$

$$\mathcal{D}_p = \frac{1}{d_I} \|G(z) - x\|^2 \quad (5)$$

Where ψ_{ϑ} is a pre-trained feature extractor, d_f and d_I are the dimensions of feature and image, respectively.

Pre-trained SISR Generator G

Since the pre-trained generator can serve as an approximation of complicated data distribution (Chan et al. 2021), optimizing its latent space yields to more informative representations. In our proposed GSDD framework, the pre-trained generator can be taken from arbitrary existing GAN-based SISR models, such as ESRGAN (Wang et al. 2018c), SFTGAN (Wang et al. 2018b), DGAN (Li et al. 2022), FSSR (Fritsche, Gu, and Timofte 2019), RealSR (Ji et al. 2020), and Real-ESRGAN (Wang et al. 2021), etc. For better generalizability and robustness, we employ pre-trained generator from the SOTA **Real-ESRGAN** model covering a wide range of practical degradations. The generator serves two purposes here: **1)** optimizing data in a more compact and informative latent coding space by back-propagation; **2)** synthesizing distilled images by feed-forward processes.

Distribution Matching Optimization

As compared to the more commonly used gradient-matching (Zhao, Mopuri, and Bilen 2020) and trajectory-matching (Cazenavette et al. 2022) strategies for DD optimization, we employ the distribution-matching (Zhao and Bilen 2023) to avoid the intrinsic bi-level optimization resulting in an expensive consumption on time and memory resources (Sachdeva and McAuley 2023). Specifically, we match the distributions of data in training set \mathcal{T} and synthetic set \mathcal{S} . The underlying assumption is that two datasets which are similar according to a particular distribution divergence metric, also lead to similarly trained models (Zhao and Bilen 2023). In practice, we employ a pre-trained feature extractor ψ_{ϑ} with parameter ϑ to achieve a mapping from input space to feature space. The synthetic data is optimized by Equation 6, where $c \in N^*$ denotes the image category.

Image Per Class (IPC)		1	5	10	20	40	60	90	100 [‡]	500 [†]	FULL
Ratio		0.07%	0.34%	0.68%	1.36%	2.71%	4.07%	6.10%	6.78%	33.90%	100%
*ESRGAN	SSIM \uparrow	0.3390	0.4026	0.4632	0.4834	0.5795	0.6165	0.6875	0.7257	0.7598	0.7661
	LPIPS \downarrow	0.6810	0.6334	0.5985	0.5101	0.4612	0.4185	0.3332	0.2940	0.2105	0.2012
*SFTGAN	SSIM \uparrow	0.4528	0.4547	0.4995	0.4958	0.5831	0.6459	0.7477	0.7754	0.8391	0.8422
	LPIPS \downarrow	0.7055	0.6720	0.6398	0.5549	0.4836	0.4511	0.3490	0.3266	0.2847	0.2738
*DGAN	SSIM \uparrow	0.4137	0.4840	0.4911	0.4832	0.5904	0.6938	0.7236	0.7541	0.8445	0.8487
	LPIPS \downarrow	0.6316	0.6003	0.5862	0.5028	0.4274	0.4021	0.2844	0.2589	0.2001	0.1966
*FSSR	SSIM \uparrow	0.4489	0.4259	0.4970	0.4856	0.5794	0.6480	0.7110	0.7705	0.8286	0.8340
	LPIPS \downarrow	0.6681	0.6511	0.5913	0.5154	0.4502	0.4077	0.2933	0.2702	0.2332	0.2217
*RealSR	SSIM \uparrow	0.4173	0.4323	0.4288	0.4042	0.5124	0.6205	0.7154	0.7822	0.8219	0.8254
	LPIPS \downarrow	0.5874	0.5667	0.5595	0.4723	0.4089	0.3805	0.2166	0.1610	0.1210	0.1123
*Real-ESRGAN	SSIM \uparrow	0.4805	0.4201	0.4992	0.4964	0.5140	0.6273	0.7618	0.7986	0.8331	0.8367
	LPIPS \downarrow	0.5102	0.5029	0.4807	0.4162	0.3841	0.3512	0.1979	0.1351	0.0924	0.0903

Table 1: The SSIM (Wang et al. 2004) and LPIPS (Zhang et al. 2018) results using different pre-trained generators for distillation w.r.t. different distilled data numbers on the OST (Wang et al. 2018b) dataset. The number of categories here is 7. IPC denotes distilled images per class. * indicates that the model was trained on the distilled dataset optimized from the corresponding generator G . FULL means the model was trained on the full-scale original OST dataset. The underline represents the best SSIM and bold represents the best LPIPS. \dagger indicates the closest result to that on the full data, and \ddagger indicates a second position.

$$\mathcal{L}(S) = \sum_{c=1}^C \|\psi_{\vartheta}(\mathcal{S}_c) - \psi_{\vartheta}(\mathcal{T}_c)\|^2 \quad (6)$$

Training Data Distillation

Inspired by the distribution matching idea discussed above, we distill dataset knowledge from real training images into synthetic ones that are produced by generator. In practice, we minimize the distillation loss \mathcal{L}_{dis} to optimize latent vectors. Instead of directly optimizing the image pixels in (Zhao and Bilen 2023), we optimize the latent vectors from a pre-trained GAN. The advantage of our approach is that it follows a generic framework, thus any pre-trained GAN can be integrated in our process. Furthermore, since we can generate arbitrary number of latent vectors to synthesize training images, the amount is more flexible and controllable compared with existing SR datasets which only have a fixed number of samples. It is also worth noting that it costs less to increase training samples by synthesizing data, compared with augmenting HR images for current limited SR datasets.

To conduct a meaningful distillation optimization (i.e., simple and fast), as mentioned before, we make use of the distribution-matching based distillation method, while getting rid of the bi-level optimization and the second-order derivation. Specifically, in a feature extractor space ψ_{ϑ} (i.e., randomly sampled embedding space), the synthetic samples are expected to have a distribution similar to the real training samples, which can be formulated as Equation 7.

$$\mathcal{L}_{dis} = \mathbb{E}_{\vartheta \sim P_{\vartheta}, \omega \sim \Omega} \|\mathfrak{A} - \mathfrak{B}\|^2 \quad (7)$$

Where \mathfrak{A} denotes $\frac{1}{|\mathcal{T}|} \sum_{i=1}^{|\mathcal{T}|} \psi_{\vartheta}(\mathcal{A}(x_i, \omega))$ and \mathfrak{B} represents $\frac{1}{|\mathcal{Z}|} \sum_{j=1}^{|\mathcal{Z}|} \psi_{\vartheta}(\mathcal{A}(G(z_j), \omega))$. Differentiable siamese augmentation (DSA) operation $\mathcal{A}(\omega)$ parameterized with

$\omega \sim \Omega$ enables the effective use of data augmentation strategies in the image synthesis process (Zhao and Bilen 2021).

Regularization Term \mathcal{R}

Because of the imbalance between the latent vector set $|\mathcal{Z}|$ and training sample set $|\mathcal{T}|$, training the model with a same distillation loss \mathcal{L}_{dis} can easily result in homogenization of different latent vectors (Elingaard et al. 2022). This reduces the data informativeness when training models, thus impairing the performance. Therefore, in accordance with common practice in the DD domain (Yu, Liu, and Wang 2024), we further introduce a regularization term \mathcal{R} to prevent the risk of over-fitting, which can be formulated as Equation 8.

$$\mathcal{R} = \mathbb{E}_{\vartheta \sim P_{\vartheta}, \omega \sim \Omega} \|\mathfrak{C} - \mathfrak{D}\|^2 \quad (8)$$

Where $\mathfrak{C} = \psi_{\vartheta}(\mathcal{A}(x_k, \omega))$ and $\mathfrak{D} = \psi_{\vartheta}(\mathcal{A}(G(z_k), \omega))$. Unlike Equation 7 where the number of x_i and z_j is unbalanced, here x_k and z_k are in pairs to retain the original information. Finally, the overall training loss of GSDD is Equation 9, where λ serves as the regularization coefficient.

$$\mathcal{L}_{overall} = (1 - \lambda) * \mathcal{L}_{dis} + \lambda * \mathcal{R} \quad (9)$$

Experiments

Dataset & Evaluation

Following the paradigm of common DD protocols, the original data are required to have their explicit categories for better distillation. To satisfy the requirement, we select the SR dataset OST (Wang et al. 2018b) for our main experiments, in which the images all have their labeled classes. It consists of data totalling over 10,324 images in 7 categories (sky, water, building, grass, plant, animal, and mountain). Since it includes two subsets, we use OutdoorSceneTraining

Algorithm	Volume	NIQE↓	NRQM↑	PI↓
BICUBIC	—	7.9556	3.1220	7.3963
ESRGAN	13,774	5.1441	4.7773	5.6033
SFTGAN	10,324	5.1622	4.5932	5.5227
DGAN	3,450	6.9110	4.0772	6.4135
FSSR	3,450	7.3122	5.1077	6.0089
RealSR	3,450	4.9824	6.1021	4.7405
Real-ESRGAN	13,774	4.5896	5.9741	4.2089
GSDD	700	4.7251	5.8498	4.4013
Difference		0.1355	0.2523	0.1924

Table 2: The comparison on the DPED (Ignatov et al. 2017) dataset with SOTA SISR models which trained on datasets from their official papers. Where bold represents the best and underline represents the second. The difference is between the proposed GSDD model and the best result.

for training and OutdoorSceneTest300 for testing. We further employ a real-world dataset DPED (Ignatov et al. 2017) to verify the generalization capability of GSDD under realistic degradation. Similar to most SR comparisons, we adopt SSIM (Wang et al. 2004), LPIPS (Zhang et al. 2018), NIQE (Mittal, Soundararajan, and Bovik 2013), NRQM (Ma et al. 2017), and PI (Wang et al. 2018c) to evaluate the recovery accuracy of various SR models trained by the distilled data.

Training Details

In our practical experiments, we use the ResNet18 (He et al. 2016) for feature extraction, due to its similarity to most existing SISR networks. The scale factor for SISR task is $\times 4$. To obtain distilled images in pairs, we adopt the classical bicubic down-sampling to HR distilled images and form LR counterparts, which facilitates implementation while ensuring the integrity of the image content and edge structure. We use Adam optimizer (Kingma and Ba 2014) with $\beta_1 = 0.9$, $\beta_2 = 0.999$, and learning rate $\eta = 0.001$ for training. The training iterations for optimizing latent vectors is 500K and regularization coefficient is set to $\lambda = 0.1$.

Comparison to the State-of-the-art

As we are the pioneer in exploring DD to SISR tasks, we first select various GAN-based SISR models to verify the effectiveness and versatility of our proposed DD pipeline, including ESRGAN (Wang et al. 2018c), SFTGAN (Wang et al. 2018b), DGAN (Li et al. 2022), FSSR (Fritsche, Gu, and Timofte 2019), RealSR (Ji et al. 2020), and Real-ESRGAN (Wang et al. 2021). To be more specific, we perform DD based on different generators from these SISR models, and show models performances when trained on small-scale distilled data. We further compare performances when trained on the full-scale original dataset for better demonstration.

Quantitative Aspect The experiments include two parts, one exploring the recovery accuracy of SOTA GAN-based SISR models w.r.t different distilled image numbers (see Table 1), the second comparing to the reported performance of generalization capability of advanced methods (see Table 2).

From Table 1, we make three observations: **1)** as the number of distilled images grows, the recovery accuracy of each

SISR model also increases (when $IPC = 500$, the performance is almost the same as the original data), due to the introduction of more informative features for model training; **2)** even if trained on only 100 distilled samples per class (reducing almost 93.2% amount of data), models still achieve comparable performances in terms of SSIM and LPIPS to those trained on entire dataset, this fully validates our effectiveness; **3)** when using Real-ESRGAN’s generator (i.e., the setting of our proposed GSDD model), it obtained the best LPIPS results among other models. *Based on the above findings, we choose to use $IPC = 100$ for subsequent experiments considering the balance between data storage (or training efficiency) and model effectiveness.*

To verify the generalization capability of GSDD on more complex degradations, we further compare it with advanced SISR models on a challenging real-world dataset DPED (Ignatov et al. 2017). From Table 2, it can be observed that GSDD maintains good performance when tested on images distributed out of training samples. Notably, GSDD outperforms some SOTA solutions (e.g., DGAN (Li et al. 2022) and FSSR (Fritsche, Gu, and Timofte 2019)) on their reported performance, even trained on less samples, reflecting the better generalization capability of our model. Nevertheless, the ideal performance achieved of GSDD also relies on advanced SISR algorithms, as can be derived from the Real-ESRGAN results (see Table 1).

Qualitative Aspect Here we show some visualization results. Firstly, in Figure 3, we select and exhibit one sample of the distilled images from each image category in the OST (Wang et al. 2018b) dataset. It is interesting to see that the distilled images from each category seem abstract and cannot be easily identified by human eyes, however, in the eyes of DNNs, they are good and meaningful image samples. If we take a closer look, we can find some peculiarities in these distilled images. For example, the distilled image for animal category has some furry textures, the image in plant category contains some fruit or leaf shapes, and the image in water category consists of ripple-like grains. This stimulates us to think further about how DNNs interpret good features and how we can construct those good training samples for DNNs in future related studies.

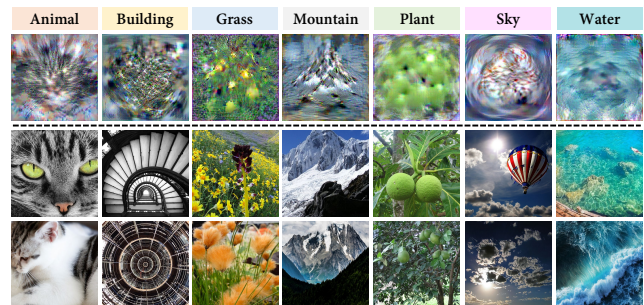


Figure 3: A visual demonstration of distilled image samples generated from the OST (Wang et al. 2018b) dataset. The first row is the distilled image and the other two rows are the original images of the corresponding categories.

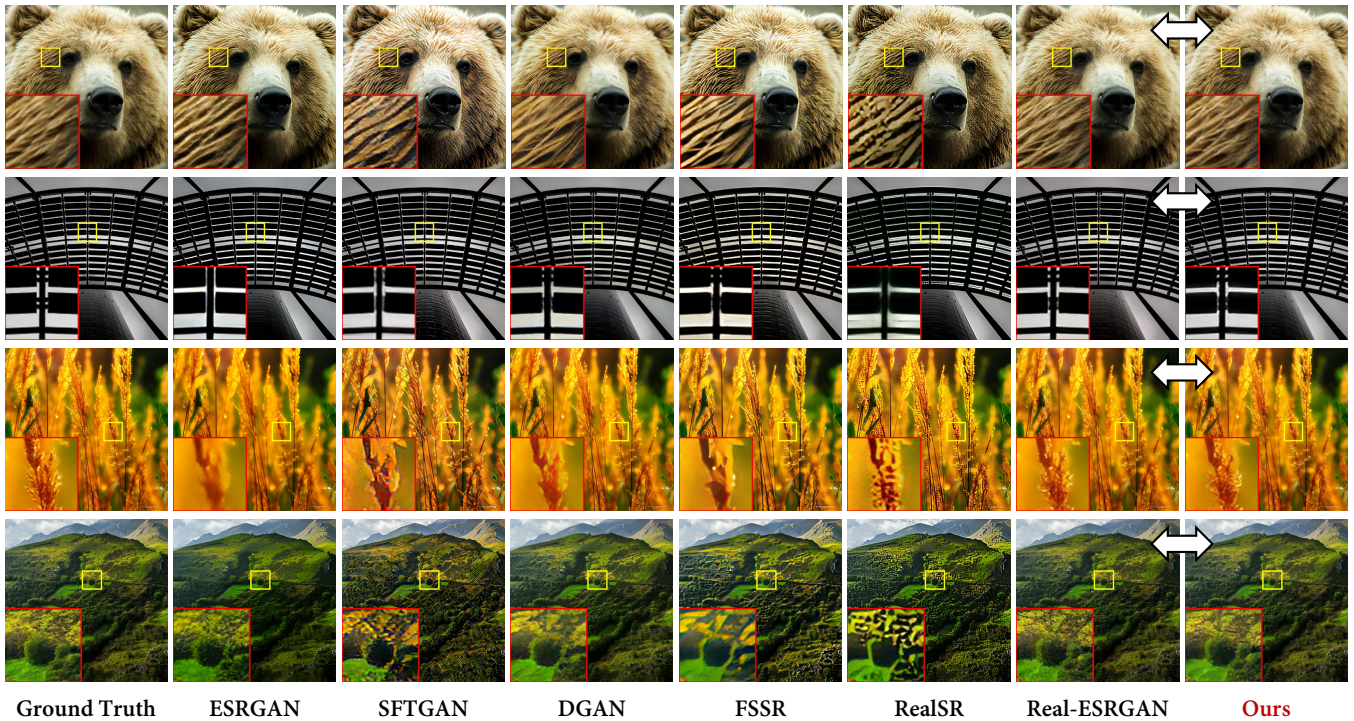


Figure 4: Visual comparison of SR results generated by GSDD (trained on 700 distilled images) and SOTA SISR models (trained on 10,324 original images) on OST. The arrow indicates comparable visual effects. Please zoom in for best view.

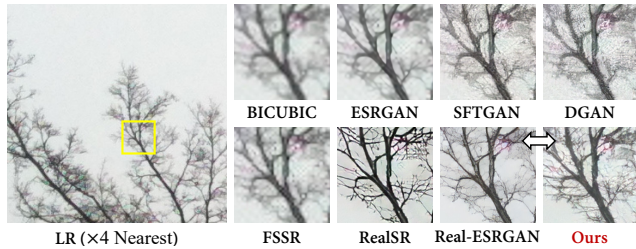


Figure 5: Visual comparison on the real-world dataset DPED (Ignatov et al. 2017). Notably, we only trained on small-scale distilled data, while others were trained on full-scale training data. The arrow indicates comparable visual effects. Please zoom in for best view.

We then provide restoration results derived from other advanced SISR models trained on the whole dataset and SR images reconstructed by the model trained on our distilled data (when IPC = 100). From Figure 4, it can be found that we obtained a fairly good perceptual result compared with other solutions, despite trained on only 700 distilled images, which is $\frac{1}{15}$ of the size of the original dataset. Particularly, we are able to faithfully restore detailed textures in the original image, which seems even not possible for RealSR (Ji et al. 2020) model. Furthermore, we successfully achieve the closest visual effects to our baseline model Real-ESRGAN on small-scale synthetic data. These results fully validate the effectiveness of our approach. Next, in Figure 5, we display the SR results of different models on the real-world dataset

Model	SSIM \uparrow	LPIPS \downarrow
GSDD	0.7986	0.1351
*ESRGAN	0.5968	0.4237
*SFTGAN	0.6221	0.4418
*DGAN	0.6577	0.3623
*FSSR	0.6996	0.3558
*RealSR	0.6831	0.2807
*SRResNet	0.6434	0.3470
*SwinIR	0.6670	0.2983

Table 3: The cross-architecture validation. The distilled data generated by proposed GSDD method are evaluated on other SISR architectures. * means that the current SISR network has been retrained on the specified distilled images.

DPED (Ignatov et al. 2017). As can be seen, even for the unknown test data in real scenes, we obtained relatively good perceptual results (a bit lower than Real-ESRGAN but obviously better than the other competitors), demonstrating the excellent generalizability achieved by our method.

We also investigate how different distilled image numbers affect the visual effects of restored images. We set four settings including IPC = 50, 100, 200, and 500 to demonstrate this influence respectively. According to Figure 6, two trends can be observed: **1**) as the number of IPC increases, it seems the textural features in distilled images are spreading out, causing that each distilled image is more identifiable with others; **2**) the distilled data contains increasing details and finer textures, and gradually reaches the visual quality of the reference image (especially for the setting of IPC = 500).

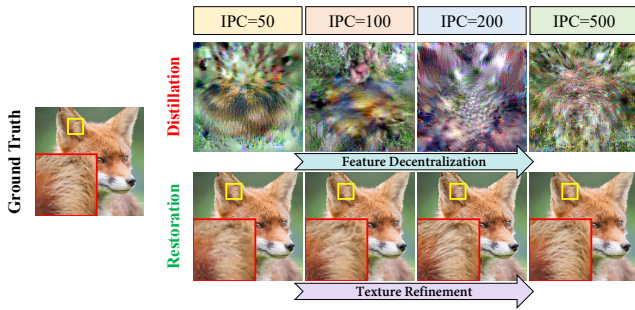


Figure 6: Visual comparison of distilled and restored results on the OST dataset when GSDD trained with different distilled image numbers. Please zoom in for best view.

Cross-architecture Validation To further verify the generality of our distilled data, we performed validation experiments. Concretely, we first learn latent vectors and generate distilled images from GSDD. Then we retrain other SISR models (we additionally add representative CNN-based (SR-ResNet (Ledig et al. 2017)) and Transformer-based (SwinIR (Liang et al. 2021)) models to verify the applicability of our approach.) on these data. At last, we test models on OutdoorSceneTest300 to observe the corresponding recovery accuracy. The results are listed in Table 3, we find that when the current synthetic training samples are applied to other SISR frameworks, acceptable restoration performance can still be achieved. This demonstrates that the distilled data do include some general features to different SISR models. In other words, the proposed distillation process is able to extract and retain some common representations that are informative for various deep learning based SISR models.

Ablation Study

The Effectiveness of Data Compression We compare with three representative SISR models (ESRGAN, RealSR, and Real-ESRGAN) on different training data. Concretely, they are coreset selection² (CS), the proposed dataset distillation method (GSDD), and total dataset (TD). We evaluate their performance using single iteration training time (s) at a fixed number (300K) of iterations, and image restoration precision (LPIPS). According to Figure 7, it is obvious that we achieve comparable generalization performance to that using the full-size of training data (TD) but consuming much less time (e.g., a nearly $\times 8$ improvement on training efficiency in Real-ESRGAN model). Furthermore, we outperform CS compression both in time consumption and recovery accuracy, which fully demonstrates our superiority.

The Effectiveness of \mathcal{R} , λ , and IPC We conduct the ablation experiment to validate \mathcal{R} and show the corresponding results in Figure 8. Concretely, when the coefficient $\lambda = 0$, training unbalanced numbers of samples by identical \mathcal{L}_{dis} would lead to the homogenization problem of latent vectors and reduce the informativeness of the distilled samples.

²For fair comparison, the number of typical samples from coreset selection (CS) is identical to the number of latent vectors from the proposed dataset distillation method (GSDD).

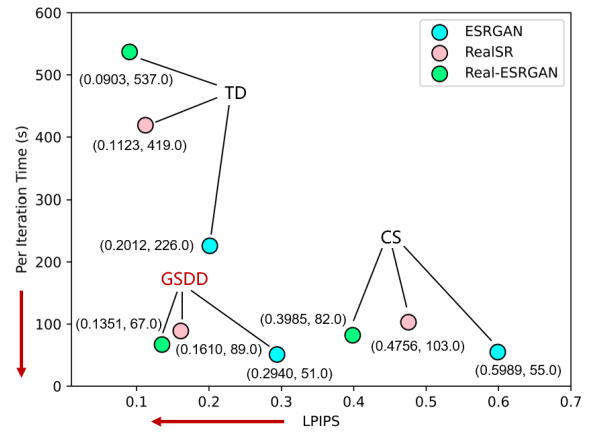


Figure 7: The ablation studies of data compression.

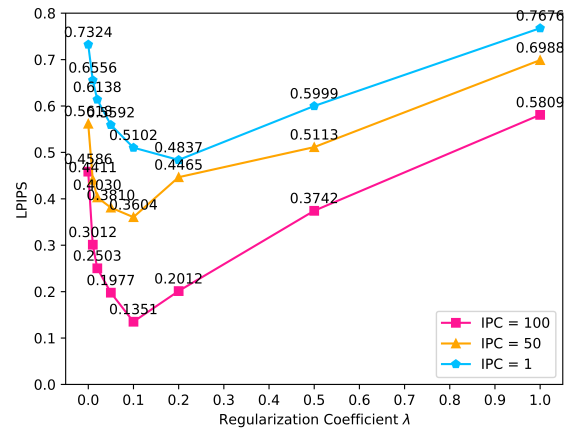


Figure 8: The ablation studies of \mathcal{R} , λ , and IPC.

It can be seen that the recovery accuracy (LPIPS) would drop from 0.1351 to 0.4586 for IPC = 100. Furthermore, we investigate the influence on recovery accuracy (LPIPS) in terms of varying λ (from 0~1). Overall, according to Figure 8, we find that the recovery accuracy is not always on an increasing trend when λ increases. Specifically, for IPC = 50 and IPC = 100, LPIPS reaches its highest value when $\lambda = 0.1$, while for IPC = 1, this value is obtained at $\lambda = 0.2$.

Conclusion

This paper focuses on exploring the possibility of applying DD to benefit low-level CV tasks (especially for SISR). Overall, we propose a GSDD framework that optimizes and synthesizes training samples by the GAN-inversion manipulation in a latent generative space. Our optimization process is based on a distribution-matching data condensation strategy, so that the SISR network can have comparable performance to models trained on the original dataset. We further improve the approach by proposing a distillation loss with the regularization term \mathcal{R} . Finally, we demonstrate its effectiveness via extensive experiments. We achieve a competitive performance compared to SOTA SISR solutions under the premise of a nearly $\times 8$ increase in training efficiency and a saving of almost 93.2% data storage space.

Acknowledgments

This work was supported by National Science Foundation of China under Grant No.U19B2037 and No.61901384, Natural Science Basic Research Program of Shaanxi Province (Program No.2021JCW-03).

References

- Abdal, R.; Qin, Y.; and Wonka, P. 2019. Image2StyleGAN: How to Embed Images Into the StyleGAN Latent Space? In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.
- Abdal, R.; Qin, Y.; and Wonka, P. 2020. Image2StyleGAN++: How to Edit the Embedded Images? In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 8293–8302.
- Agustsson, E.; and Timofte, R. 2017. NTIRE 2017 Challenge on Single Image Super-Resolution: Dataset and Study. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 1122–1131.
- Bojanowski, P.; Joulin, A.; Lopez-Pas, D.; and Szlam, A. 2018. Optimizing the Latent Space of Generative Networks. In Dy, J.; and Krause, A., eds., *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, 600–609. PMLR.
- Cazenavette, G.; Wang, T.; Torralba, A.; Efros, A. A.; and Zhu, J.-Y. 2022. Dataset Distillation by Matching Training Trajectories. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 10708–10717.
- Chan, K. C.; Wang, X.; Xu, X.; Gu, J.; and Loy, C. C. 2021. GLEAN: Generative Latent Bank for Large-Factor Image Super-Resolution. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 14240–14249.
- Creswell, A.; and Bharath, A. A. 2019. Inverting the Generator of a Generative Adversarial Network. *IEEE Transactions on Neural Networks and Learning Systems*, 30(7): 1967–1974.
- Cui, J.; Wang, R.; Si, S.; and Hsieh, C.-J. 2022. DC-BENCH: Dataset Condensation Benchmark. *arXiv preprint arXiv:2207.09639*.
- Deng, Z.; and Russakovsky, O. 2022. Remember the Past: Distilling Datasets into Addressable Memories for Neural Networks. In *Neural Information Processing Systems (NeurIPS)*.
- Dong, C.; Loy, C. C.; He, K.; and Tang, X. 2014. Learning a Deep Convolutional Network for Image Super-Resolution. In Fleet, D.; Pajdla, T.; Schiele, B.; and Tuytelaars, T., eds., *Computer Vision – ECCV 2014*, 184–199. Cham: Springer International Publishing. ISBN 978-3-319-10593-2.
- Dong, C.; Loy, C. C.; He, K.; and Tang, X. 2016. Image Super-Resolution Using Deep Convolutional Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(2): 295–307.
- Dong, T.; Zhao, B.; and Lyu, L. 2022. Privacy for free: How does dataset condensation help privacy? In *International Conference on Machine Learning*, 5378–5396. PMLR.
- Elingaard, M. O.; Aage, N.; Bærentzen, J. A.; and Sigmund, O. 2022. De-homogenization using convolutional neural networks. *Computer Methods in Applied Mechanics and Engineering*, 388: 114197.
- Fritsche, M.; Gu, S.; and Timofte, R. 2019. Frequency separation for real-world super-resolution. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, 3599–3608. IEEE.
- Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2020. Generative Adversarial Networks. *Commun. ACM*, 63(11): 139–144.
- Haris, M.; Shakhnarovich, G.; and Ukita, N. 2021. Deep Back-Project Networks for Single Image Super-Resolution. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(12): 4323–4337.
- He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep Residual Learning for Image Recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Hinton, G.; Vinyals, O.; and Dean, J. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.
- Hu, S.; Goetz, J.; Malik, K.; Zhan, H.; Liu, Z.; and Liu, Y. 2022. Fedsynth: Gradient compression via synthetic data in federated learning. *arXiv preprint arXiv:2204.01273*.
- Ignatov, A.; Kobyshev, N.; Timofte, R.; Vanhoey, K.; and Van Gool, L. 2017. DSLR-Quality Photos on Mobile Devices With Deep Convolutional Networks. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.
- Ji, X.; Cao, Y.; Tai, Y.; Wang, C.; Li, J.; and Huang, F. 2020. Real-world super-resolution via kernel estimation and noise injection. In *proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, 466–467.
- Kim, J.; Lee, J. K.; and Lee, K. M. 2016. Accurate Image Super-Resolution Using Very Deep Convolutional Networks. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1646–1654.
- Kim, J.-H.; Kim, J.; Oh, S. J.; Yun, S.; Song, H.; Jeong, J.; Ha, J.-W.; and Song, H. O. 2022. Dataset Condensation via Efficient Synthetic-Data Parameterization. In Chaudhuri, K.; Jegelka, S.; Song, L.; Szepesvari, C.; Niu, G.; and Sabato, S., eds., *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, 11102–11118. PMLR.
- Kingma, D. P.; and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.
- Ledig, C.; Theis, L.; Huszár, F.; Caballero, J.; Cunningham, A.; Acosta, A.; Aitken, A.; Tejani, A.; Totz, J.; Wang, Z.; and Shi, W. 2017. Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 105–114.

- Lei, S.; and Tao, D. 2024. A Comprehensive Survey of Dataset Distillation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(1): 17–32.
- Li, W.; Zhou, K.; Qi, L.; Lu, L.; and Lu, J. 2022. Best-Buddy GANs for Highly Detailed Image Super-resolution. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(2): 1412–1420.
- Liang, J.; Cao, J.; Sun, G.; Zhang, K.; Van Gool, L.; and Timofte, R. 2021. SwinIR: Image Restoration Using Swin Transformer. In *2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, 1833–1844.
- Lim, B.; Son, S.; Kim, H.; Nah, S.; and Lee, K. M. 2017. Enhanced Deep Residual Networks for Single Image Super-Resolution. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 1132–1140.
- Liu, P.; Yu, X.; and Zhou, J. T. 2022. Meta knowledge condensation for federated learning. *arXiv preprint arXiv:2209.14851*.
- Ma, C.; Yang, C.-Y.; Yang, X.; and Yang, M.-H. 2017. Learning a no-reference quality metric for single-image super-resolution. *Computer Vision and Image Understanding*, 158: 1–16.
- Mittal, A.; Soundararajan, R.; and Bovik, A. C. 2013. Making a “Completely Blind” Image Quality Analyzer. *IEEE Signal Processing Letters*, 20(3): 209–212.
- Nguyen, T.; Chen, Z.; and Lee, J. 2020. Dataset meta-learning from kernel ridge-regression. In *2020 International Conference on Learning Representations (ICLR)*.
- Nguyen, T.; Novak, R.; Xiao, L.; and Lee, J. 2021. Dataset distillation with infinitely wide convolutional networks. In *2021 Conference and Workshop on Neural Information Processing Systems (NeurIPS)*.
- Phillips, J. M. 2017. Coresets and sketches. In *Handbook of discrete and computational geometry*, 1269–1288. Chapman and Hall/CRC.
- Rebuffi, S.-A.; Kolesnikov, A.; Sperl, G.; and Lampert, C. H. 2017. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2001–2010.
- Sachdeva, N.; and McAuley, J. 2023. Data Distillation: A Survey. *arXiv preprint arXiv:2301.04272*.
- Such, F. P.; Rawal, A.; Lehman, J.; Stanley, K.; and Clune, J. 2020. Generative teaching networks: Accelerating neural architecture search by learning to generate synthetic training data. In *International Conference on Machine Learning*, 9206–9216. PMLR.
- Timofte, R.; Agustsson, E.; Gool, L. V.; Yang, M.-H.; Zhang, L.; Lim, B.; Son, S.; Kim, H.; Nah, S.; Lee, K. M.; Wang, X.; Tian, Y.; Yu, K.; Zhang, Y.; Wu, S.; Dong, C.; Lin, L.; Qiao, Y.; Loy, C. C.; Bae, W.; Yoo, J.; Han, Y.; Ye, J. C.; Choi, J.-S.; Kim, M.; Fan, Y.; Yu, J.; Han, W.; Liu, D.; Yu, H.; Wang, Z.; Shi, H.; Wang, X.; Huang, T. S.; Chen, Y.; Zhang, K.; Zuo, W.; Tang, Z.; Luo, L.; Li, S.; Fu, M.; Cao, L.; Heng, W.; Bui, G.; Le, T.; Duan, Y.; Tao, D.; Wang, R.; Lin, X.; Pang, J.; Xu, J.; Zhao, Y.; Xu, X.; Pan, J.; Sun, D.; Zhang, Y.; Song, X.; Dai, Y.; Qin, X.; Huynh, X.-P.; Guo, T.; Mousavi, H. S.; Vu, T. H.; Monga, V.; Cruz, C.; Egiazarian, K.; Katkovnik, V.; Mehta, R.; Jain, A. K.; Aggarwala, A.; Praveen, C. V. S.; Zhou, R.; Wen, H.; Zhu, C.; Xia, Z.; Wang, Z.; and Guo, Q. 2017. NTIRE 2017 Challenge on Single Image Super-Resolution: Methods and Results. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 1110–1121.
- Wang, T.; Zhu, J.-Y.; Torralba, A.; and Efros, A. A. 2018a. Dataset distillation. *arXiv preprint arXiv:1811.10959*.
- Wang, X.; Xie, L.; Dong, C.; and Shan, Y. 2021. Real-ESRGAN: Training Real-World Blind Super-Resolution with Pure Synthetic Data. In *2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, 1905–1914.
- Wang, X.; Yu, K.; Dong, C.; and Change Loy, C. 2018b. Recovering Realistic Texture in Image Super-Resolution by Deep Spatial Feature Transform. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 606–615.
- Wang, X.; Yu, K.; Wu, S.; Gu, J.; Liu, Y.; Dong, C.; Qiao, Y.; and Change Loy, C. 2018c. ESRGAN: Enhanced Super-Resolution Generative Adversarial Networks. In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*.
- Wang, Z.; Bovik, A.; Sheikh, H.; and Simoncelli, E. 2004. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4): 600–612.
- Xia, W.; Zhang, Y.; Yang, Y.; Xue, J.-H.; Zhou, B.; and Yang, M.-H. 2023. GAN Inversion: A Survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(3): 3121–3138.
- Yu, R.; Liu, S.; and Wang, X. 2024. Dataset Distillation: A Comprehensive Review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(1): 150–170.
- Zhang, R.; Isola, P.; Efros, A. A.; Shechtman, E.; and Wang, O. 2018. The Unreasonable Effectiveness of Deep Features as a Perceptual Metric. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 586–595.
- Zhang, W.; Liu, Y.; Dong, C.; and Qiao, Y. 2019. RankSRGAN: Generative Adversarial Networks With Ranker for Image Super-Resolution. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 3096–3105.
- Zhao, B.; and Bilen, H. 2021. Dataset condensation with differentiable siamese augmentation. In *2021 International Conference on Machine Learning (ICML)*, 12674–12685.
- Zhao, B.; and Bilen, H. 2023. Dataset Condensation with Distribution Matching. In *2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 6503–6512.
- Zhao, B.; Mopuri, K. R.; and Bilen, H. 2020. Dataset condensation with gradient matching. In *2020 International Conference on Learning Representations (ICLR)*, 12674–12685.