

Knowledge-Based Systems

Freezing Partial Source Representations Matters for Image Inpainting under Limited Data

--Manuscript Draft--


Manuscript Number:	KNOSYS-D-23-04001
Article Type:	Full Length Article
Keywords:	Image inpainting; Transfer learning; Limited data
Abstract:	<p>Recent years have seen significant advances in image inpainting, for any shape of missing regions. However, the performance of existing methods degrades drastically when insufficient data is given (e.g., 100), which has drawn limited attention in the community. This work provides an appropriate solution for image inpainting on the challenging limited data regime. Specifically, we first make an in-depth comparison on fine-tuning and training from scratch and find that, although the former advances the performance than the latter, the overall structural consistency and fine details are still unsatisfactory. Consequently, we propose a two-stage method based on transfer learning, namely T2inpaint. In stage one, only domain-specific weights are updated, enabling the model to capture the preference structure of the target domain. In stage two, we freeze the network obtained from the first stage and adapt the model to the target domain with additional parameters. Such that, the reusable knowledge of the source domain could better guide the optimization process and avoid ambiguous contents. Extensive experiments on various datasets demonstrate that our T2inpaint produces plausible images and achieves state-of-the-art performance. Moreover, an empirical study on the source domains, data regimes, and various data augmentation is conducted, facilitating potential interesting works.</p>

Dear Editor,

We the undersigned declare that this manuscript entitled “Freezing Partial Source Representations Matters for Image Inpainting under Limited Data” is original, has not been published before and is not currently being considered for publication elsewhere.

We would like to draw the attention of the Editor to the following publications of one or more of us that refer to aspects of the manuscript presently being submitted. Where relevant copies of such publications are attached.

We confirm that the manuscript has been read and approved by all named authors and that there are no other persons who satisfied the criteria for authorship but are not listed. We further confirm that the order of authors listed in the manuscript has been approved by all of us. We understand that the Corresponding Author is the sole contact for the Editorial process. He is responsible for communicating with the other authors about progress, submissions of revisions and final approval of proofs.

Signed by all authors as follows: Yanbing Zhang, Mengping Yang, Ting Xiao, , Ziqiu Chi

Freezing Partial Source Representations Matters for Image Inpainting under Limited Data

Yanbing Zhang^{a,b}, Mengping Yang^{a,b}, Ting Xiao^{a,b,*}, Zhe Wang^{a,b,*}, Ziqiu Chi^{a,b}

^a*Key Laboratory of Smart Manufacturing in Energy Chemical Process,
Ministry of Education, East China University of Science and Technology,
Shanghai 200237, China*

^b*Department of Computer Science and Engineering,
East China University of Science and Technology, Shanghai 200237, China*

Abstract

Recent years have seen significant advances in image inpainting, for any shape of missing regions. However, the performance of existing methods degrades drastically when insufficient data is given (*e.g.*, 100), which has drawn limited attention in the community. This work provides an appropriate solution for image inpainting on the challenging limited data regime. Specifically, we first make an in-depth comparison on fine-tuning and training from scratch and find that, although the former advances the performance than the latter, the overall structural consistency and fine details are still unsatisfactory. Consequently, we propose a two-stage method based on transfer learning, namely $T^2inpaint$. In stage one, only domain-specific weights are updated, enabling the model to capture the preference structure of the target domain. In stage two, we freeze the network obtained from the first stage and adapt the model to the target domain with additional parameters. Such that, the reusable knowledge of the source domain could better guide the optimization process and avoid ambiguous contents. Extensive experiments on various datasets demonstrate that our $T^2inpaint$ produces plausible images and achieves state-of-the-art performance. Moreover, an empirical study on the source domains, data regimes, and various

*Corresponding author

Email addresses: xiaoting@ecust.edu.cn (Ting Xiao), wangzhe@ecust.edu.cn (Zhe Wang)

data augmentation is conducted, facilitating potential interesting works.

Keywords: Image inpainting, Transfer learning, Limited data

1. Introduction

Image inpainting aims to reconstruct missing regions, which can be either naturally occurring or artificially edited, to obtain a more realistic image. It has many practical applications such as old photo restoration [1, 2], unwanted
5 object removal [3], face editing [4, 5], and video inpainting [6]. Thanks to deep-based inpainting methods [7, 8, 9, 10, 11, 12, 13], numerous inpainted images are indistinguishable from natural images, whatever shapes of missing regions.

The success of image inpainting is partly due to the use of large datasets, such as the widely used CelebA-HQ [14] (30,000 images) and Places-Challenge [15]
10 (8,000,000 images) datasets. However, in many real-world applications, only a limited number of training samples are available, which results in poor quality of inpainting (as shown in Figure 1). Therefore, we take a first step toward seeking satisfactory image inpainting from limited data.

Intuitively, one may reuse knowledge pre-trained on large-scale datasets via
15 transfer learning for limited data learning [16], *e.g.*, fine-tuning [17]. Despite many advances have been made in discriminative tasks, such as classification and segmentation, there is no attempts of employing transfer-based methods within the context of inpainting community. Accordingly, we make abundant comparisons on fine-tuning and training from scratch under multiple datasets
20 and settings. Based on our experimental results, we found that the fine-tuning yields a significant improvement compared to training from scratch, but still suffers from structural inconsistency and poor texture detail, as shown in Figure 1. For example, fine-tuning is incapable of generating geometrically reasonable objects for the CLEVR [18] dataset, nor can it produce realistically detailed eyes
25 for the AFHQ [19] dataset. There are many works devoted to achieving consistent global structures [7, 10] and fine details [9, 20, 21], but they are all based on methods that rely on large-scale data and are obviously unsuitable for scenarios

with limited data. Such that, we aim to enhance inpainted performance from the perspective of developing better transfer.

30 Recently, there have been many studies that investigate how to improve transfer. Some methods argue that tuning all parameters during transfer is not optimal [22, 23, 24, 25, 26, 27], and instead, different ways are used to select weights that need to be changed. Updating only these weights learns more domain-specific semantic aspects, including the global structure of data. 35 Similarly, we adjust specific weights during transfer to resolve the defect of the unrealistic structure obtained from fine-tuning. Besides, some methods point out that tuning parameters except for domain-specific weights will boost performance [27, 28]. Therefore the process after the domain-specific transfer, the second stage, is necessary. The second stage in [28] involves a simple fine-tuning process, but this can easily lead to overfitting in cases of limited data. 40 Here motivated by several one-stage methods [29, 30], we believe that attaching new parameters while freezing the backbone can generate more detailed results. The methods used in each stage can alleviate overfitting arising from limited training data, as updating only a few parameters in stage one and the frozen 45 backbone in stage two can be viewed as a regularization term.

Based on the above observations, this paper proposes a transfer-based two-stage inpainting ($T^2\text{inpaint}$) method under limited data. To address the issue of structural inconsistency, we implement domain-specific transfer in stage one. In this process, we only update the weights that are inclined to global patterns, 50 allowing the network to focus on learning structural features of the target domain. To address the problem of lack of detail, in the second stage, we freeze the network obtained from stage one and add additional parameters to better adapt to the target domain. This approach can maintain the low-level statistics features learned from the source domain as a guide to learn the details of the 55 target domain.

Our main contributions can be summarized in four parts:

- We evaluate a variety of experimental settings, showing that using pre-

trained inpainting models significantly improves performance over training from scratch under limited target data.

- 60 • We propose a transfer-based two-stage inpainting method that can fully exploit useful features of pre-trained models to generate high-fidelity restoration results that are structurally consistent and fine-detailed under limited data.
- 65 • Extensive experiments on various public datasets with limited data show that our method has state-of-the-art performance under various data sizes. Ablation studies demonstrate the effectiveness of each component and the best ways in each stage.
- 70 • We find that during the transfer, the more abundant data the source domain possesses, the more improvement is observed; and traditional data augmentation helps when the target data size is extremely small (≤ 100).

2. Related Work

2.1. Image Inpainting

Image inpainting, which has important practical significance, has been developed for decades. Traditional image inpainting methods use low-level statistical information [31, 32] within the image or search for similar patches [3, 33] 75 from known regions to fill in unknown regions, resulting in blurry generated content and unable to handle large missing regions. Pathak et al. [34] firstly proposed the use of an encoder-decoder architecture for achieving semantically-plausible inpainting, which spurred the rapid development of GAN-based [35] deep generative methods. Iizuka et al. [36] proposed using a global and local 80 discriminator for better global and local consistency, and used dilated convolution [37] to expand the receptive field. Subsequently, methods for generating coarse intermediate results, such as semantic segmentation maps [38] and object structures [7, 39, 40], also made significant progress. Adding attention

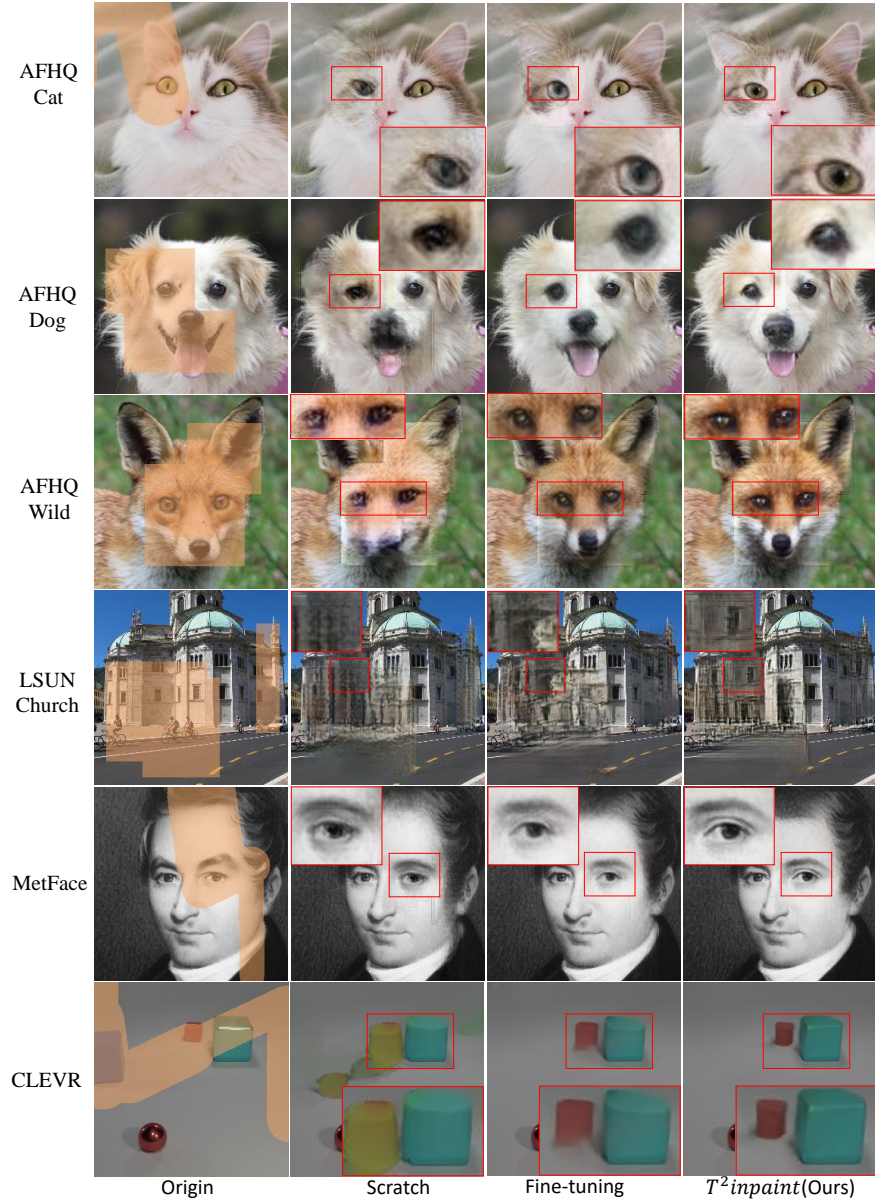


Figure 1: Qualitative evaluation on six 256×256 datasets with only 100 samples. The images are from the test sets of six different datasets. Except for the Scratch, which trains from scratch, the other methods are initialized with a pre-trained checkpoint from the CelebA-HQ [14] dataset. Fine-tuning is able to initially utilize the features learned from the source domain to mitigate the issue of generating blurry images through Scratch, but it still struggles to produce clearer object boundaries and details. Our method generates better results for both structure and texture through domain-specific transfer and better feature reuse.

mechanisms [41], using partial convolutions [42], layer-wise pyramidal convolutions [43], and gated convolutions [8] are also effective approaches. LaMa [10] proposed using Fast-Fourier Convolutions [44] to obtain image-wide receptive field and achieved remarkable results. There have been many improvements based on this, such as adding frequency loss [45], iterative refinement for detail [20], combining LaMa [10] with CoModGAN [9] and combining LaMa [10] with transformer [11]. At the same time, diffusion probabilistic models [46] have also achieved great success [12, 13], but their training and inference costs are very expensive. All of the above methods do not consider the scenario of limited data scale which we focus on.

2.2. Transfer Learning

Transfer learning refers to reusing knowledge from a source domain when training in a target domain [16]. It is often used to compensate for the lack of features in the target domain [47, 48]. In classification tasks, the widely used methods are linear probing (updating only the last classification head and freezing the other layers) and fine-tuning (updating all weights during training). However, these two methods are not always the optimal choice [29]. Some methods attempt to select weights that are more suitable for transfer, such as using a policy network [22], recurrent neural network [49] gate [23], group lasso [24], and evolutionary algorithms [25]. In addition to these, there are also methods such as using a two-stage approach [28], using an adapter while freezing the backbone [29], and combining multiple pre-trained models [50, 51, 52] for transfer. There are also many transfer-based works in generative tasks based on GAN architecture. Wang et al. [53] were the first to explore the application of fine-tuning techniques in image generation. [54] only updated the scale and shift parameters of the batch normalization statistics within the generator, thereby allowing the model to select filters similar to the target domain. [26, 27] discussed which layers in the network should be frozen when transferring. Wang et al. [55] proposed fixing the whole pre-trained network and training an additional miner network. In addition to designing trainable parameters, [56] and [57] re-

115 spectively design elastic weight consolidation loss [58] and cross-domain distance
 consistency loss to enhance the quality and diversity of generated images. Xiao
 et al. further optimized the training scheme in [55] and the loss from [57] to
 resolve overfitting. However, these methods do not design a transfer method for
 the image inpainting task.

120 3. Method

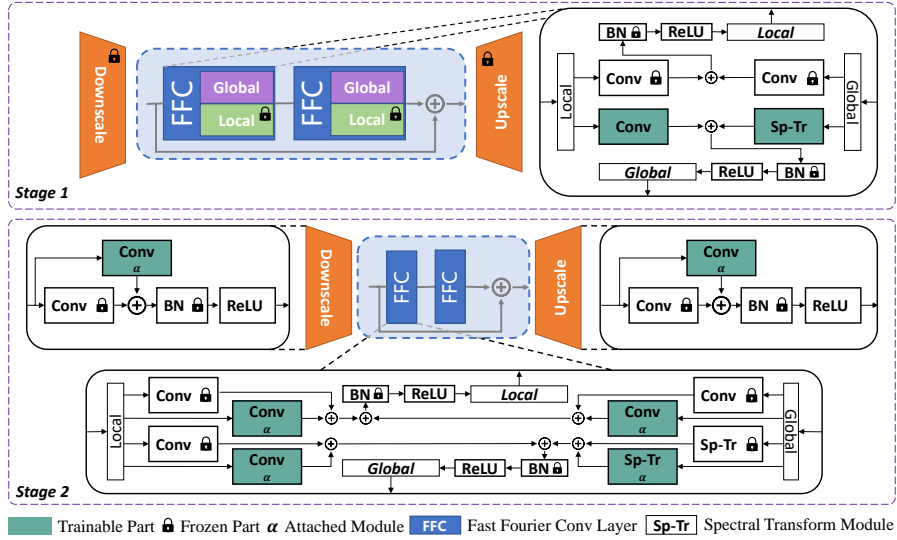


Figure 2: The overall framework of our proposed transfer-based two-stage inpainting ($T^2inpaint$) method. During training in Stage 1, We only adjust the parameters of the global branch, shown as green blocks. After that, we freeze the network obtained from Stage 1 and train new attached modules, shown as green blocks with α in Stage 2. The first stage focuses on domain-specific transfer for the model to learn global patterns, while the second stage attaches new modules for better feature reuse.

We propose the transfer-based two-stage inpainting ($T^2inpaint$) method under limited data, and the overall framework is shown in Figure 2. In this section, we first introduce the preliminary of image inpainting. Second, we explain why transfer-based methods should be used when training data is limited. After that, we describe how to perform the domain-specific transfer in stage one. Fi-

nally, we introduce how to reuse features to further refine local textures in stage two.

3.1. Preliminary

Our goal is to explore image inpainting that only can be trained on a small dataset. The inpainting task aims to complete an RGB image X_m with missing region m , where $X_m = X_{org} \odot m$. During training, (X_{org}, m) pairs are obtained from real images and randomly generated masks. The input is a four-channel tensor $X' = \text{concat}(X_m, m)$ that concatenates the masked image X_m and the mask m . X' is fed into the inpainting network $f_\theta(\cdot)$ to output an inpainted RGB image $\hat{X} = f_\theta(X')$.

3.2. Fine-tuning on Image Inpainting

A simple solution to deal with the issue of limited data is to use fine-tuning technology, specifically using a pre-trained model from a large-scale source domain as the initial checkpoint, and then training on a small-scale target domain with updating all weights. There is no work to discuss fine-tuning for image inpainting yet, so we first time try to assess whether image inpainting trained on small datasets can benefit from a pre-trained model.

We set only 100 training images in each target dataset to ensure the scarcity of data; at the same time, to check the generalization of fine-tuning, we select six small-scale target datasets (see Sec. 4.1 for details); and the source domain dataset is CelebA-HQ [14]. The results are shown in Table 1. Compared with training from scratch, fine-tuning achieves better performance on all target datasets and improves significantly on datasets whose distribution is similar to the source domain. As also shown in Figure 1, fine-tuning can obtain more realistic images than training from scratch. See Sec. 4.2 for more discussion on fine-tuning.

We explicitly demonstrate that fine-tuning can alleviate the problem of insufficient features under limited data in image inpainting through quantitative

and qualitative results. This is due to the **transferable and effective representations** learned by pre-trained models on the source domain with large data. However, fine-tuning all parameters will distort pre-trained features[28], resulting in inconsistent structure and blurred content, such as the third picture in the first row of Figure 1. These findings motivate our next work on how to leverage pre-trained features for more reasonable transfer.

3.3. Domain-specific Transfer

We believe that domain-specific transfer is crucial, as demonstrated in [26, 27, 59]. In classification tasks, the domain-specific transfer is typically accomplished by updating only the last linear layer (also known as linear probing); in generative tasks, [26] fixes the lower layers of the discriminator and fine-tunes only the upper layers, while [27] further fixes the lower layers of the generator. However, to date, there has been no work on domain-specific transfer for encoder-decoder structures such as image inpainting.

Most image inpainting models downscale the image, use intermediate “bottleneck” layers that capture global information to perform the restoration process, and then upscale to the original image dimensionality. Due to its access to global information, the features of intermediate *bottleneck* layers are domain-specific. Here, we further narrow down the scope of domain-specific features of the inpainting network using the fast Fourier Conv (FFC) [10] structure, rather than training all layers in the *bottleneck*. As pointed out by [21], the Spectral Transform module in the FFC is capable of learning global repeating patterns, which can be regarded as domain-specific features. However, adapting to the target domain through only the Spectral Transform module will severely lack the local features of the image. It is necessary to release the layers that can learn local features in the global branch, as shown in Stage 1 of Figure 2. The ablation studies, shown in Table 5, further verifies that such a combination is optimal (detailed in Sec. 4.3). Therefore, the results of the local and global

branches in the FFC layer can be obtained as follows:

$$\begin{aligned} X'_{Local} &= \text{ReLU}(\text{BN}(\text{Conv}(X_{Local}) + \text{Conv}(X_{Global}))) \\ X'_{Global} &= \text{ReLU}(\text{BN}(\mathbf{Conv}(X_{Local}) + \mathbf{Sp-Tr}(X_{Global}))) \end{aligned} \quad (1)$$

where X_{Local} and X_{Global} are the results of the local and global branches in the previous FFC layer, the **Conv** and **Sp-Tr** modules in the global branch are trainable, and the remaining modules Conv, BN are frozen. All Conv in the two branches are 3×3 convolution, and **Sp-Tr** is the Spectral Transform module. **Sp-Tr** uses real fast Fourier transform to obtain global context, with details in [10]. This way of transfer forces the network to learn domain-specific features, which makes it easier to learn global information about the target domain from limited data.

3.4. Better Feature-reuse

The domain-specific transfer allows the model to focus on learning global context, but the model does not learn detailed features such as fine textures well. As [28] suggests that fine-tuning all parameters after adjusting the last linear layer is effective for classification tasks, we believe that the adjustment of other parameters in the stage two is necessary following domain-specific transfer to address this issue. However, if all parameters are fine-tuned in the stage two, it will destroy the good features learned from the source domain, so we freeze the inpainting network $f_{\theta}(\cdot)$ and add a trainable adapter to each module. For classification tasks, [29] parallel connects a convolution layer with 1×1 kernels as an adapter to each convolution layer in the ResNet backbone. Inspired by this, and taking into account that image inpainting is a dense prediction task, we adopt more complex layers as adapters. Therefore, we parallel connect a new 3×3 convolution layer to each convolution layer, and parallel connect a new Sp - Tr module to each Sp - Tr module, as shown in Stage 2 of Figure 2. Specifically, we attach adapters r_{α} to the *Downscale*, *bottleneck*, and *Upscale* in the inpainting network as follows:

$$f_{\theta_l, \alpha}(X) = r_{\alpha}(X) + f_{\theta_l}(X) \quad (2)$$

where $X \in \mathbf{R}^{W \times H \times C}$ is the input tensor, $f_{\theta_l}(\cdot)$ is the l -th Conv or Sp – Tr in the network obtained from Stage 1, and its parameters are frozen. r_α is a newly added trainable Conv or Sp – Tr. We parallel connect newly Sp – Tr to frozen Sp – Tr instead of newly Conv, because Conv is not as efficient as Sp – Tr in learning global context (detailed in Sec.4.3). Since we freeze the backbone network, we can maintain the low-level statistics features learned from the source domain. At the same time, the frozen features can serve as a strong regularization term to correct the model’s optimization process [30] and mitigate overfitting due to limited data (see Sec.4.4).

4. Experiments

In this section, we demonstrate the effectiveness of the method from both quantitative and qualitative results and uncover the impact of several aspects on the pre-trained model’s transferability. We then conduct ablation experiments, including the importance of each component, the different settings of each method, whether the simple data augmentation should be used, and integrating our method to new models. Finally, we further show the feasibility of the method through feature visualization and convergence analysis.

4.1. Experimental Settings

1) *Datasets and Metrics:* We use six datasets, AFHQ-Cat,-Dog,-Wild [19], LSUN-Church [60], MetFace [61], and CLEVR [18], representing four different target domain. The AnimalFace HQ (AFHQ) dataset consists of faces of three classes of animals, each of which has approximately 5k training images and 500 test images. The LSUN-Church dataset is a collection of 126k images of complex outdoor churches. The MetFace is a dataset of human faces extracted from works of art, with a total of 1,336 images. The CLEVR is a dataset of structured geometric objects, with a total of 70k images.

For each dataset, we first explored the case where the training data only has 100 images to correspond to the scene of scarce data, which is analogous

to other generative tasks [26, 62]. Since the 100 randomly selected images may not cover the distribution of the training data well, we use three different random seeds to randomly select three sets of training sets with 100 images, i.e., training the model three times and finally taking the average of the three
240 test results. We then explore the cases where the training data has 500, 1000, 2000, and 5000 images. In each training strategy, we randomly selected 500 images as the validation set. For AFHQ, we use its original 500 test images; for LSUN-Church and CLEVR, we randomly select 2000 images as the test set; for MetFace, as it only has 1336 samples, we randomly select 500 images as the
245 test set. All images are resized to 256×256 , and high-resolution images are either scaled down proportionally or cropped from the center [63]. We use the Learned Perceptual Image Patch Similarity (LPIPS) [64] and Fréchet inception distance (FID) [65], which are commonly used in recent image inpainting works, as metrics for validating performance.

250 2) *Baselines*: We select the milestone work LaMa [10] in image inpainting as the baseline. To clearly understand the impact of different source domains on model transferability, we choose LaMa-Fourier with the source domain being CelebA-HQ [14] (containing 30k high-quality human facial images) and Big LaMa-Fourier with the source domain being Places-Challenge [15] (containing 8
255 million real-world images). Note that the number of parameters for Big LaMa-Fourier is almost twice that of LaMa-Fourier. Therefore, the comparison methods are (i) Scratch, which trains the model from scratch using the data from the target domain; and (ii) Fine-tuning, which adjusts all parameters of the pre-trained model from different source domains on the target data.

260 3) *Implementation Details*: The loss function, random mask generation strategy, optimizer, and learning rate used in the training process are the same as those in LaMa [10]. In all experiments, the batch size is 15. For training data sizes of 100 and 500, the models are trained for 20,000 iterations; for training data sizes of 1000, 2000, and 5000, the number of iterations is 40,000, 60,000,
265 and 60,000, respectively. During testing, we use medium and thick mask generation strategies [10] in ways that contain random strokes and rectangular boxes,

Table 1: Quantitative evaluation on six 256×256 datasets with only 100 samples. We report the FID and LPIPS metrics, with lower values indicating better performance. Except for the Scratch, which trains from scratch, the other methods are initialized with a pre-trained checkpoint from the CelebA-HQ [14] dataset. Run with three different random seeds and take the average of the test results. **Bold** text indicates the best performance, and blue text indicates the second best.

CelebA-HQ \rightarrow	AFHQ Cat(256×256)				AFHQ Dog(256×256)				AFHQ Wild(256×256)			
	Medium masks		Thick masks		Medium masks		Thick masks		Medium masks		Thick masks	
	FID \downarrow	LPIPS \downarrow	FID \downarrow	LPIPS \downarrow	FID \downarrow	LPIPS \downarrow	FID \downarrow	LPIPS \downarrow	FID \downarrow	LPIPS \downarrow	FID \downarrow	LPIPS \downarrow
Scratch	9.95	0.139	12.28	0.156	15.89	0.149	22.64	0.170	5.72	0.149	7.24	0.156
Fine-tuning	7.76	0.118	9.47	0.136	11.90	0.126	17.04	0.148	4.28	0.130	5.22	0.138
Only DS	8.11	0.120	9.87	0.137	11.80	0.126	16.85	0.146	4.46	0.132	5.13	0.139
$T^2inpaint$ (ours)	7.32	0.113	9.07	0.131	11.51	0.125	16.14	0.145	3.87	0.126	4.54	0.134
CelebA-HQ \rightarrow	Church(256×256)				MetFace(256×256)				CLEVR(256×256)			
	Medium masks		Thick masks		Medium masks		Thick masks		Medium masks		Thick masks	
	FID \downarrow	LPIPS \downarrow	FID \downarrow	LPIPS \downarrow	FID \downarrow	LPIPS \downarrow	FID \downarrow	LPIPS \downarrow	FID \downarrow	LPIPS \downarrow	FID \downarrow	LPIPS \downarrow
Scratch	10.46	0.163	12.72	0.176	24.31	0.141	24.33	0.148	10.81	0.130	11.19	0.138
Fine-tuning	9.31	0.155	11.78	0.170	18.75	0.119	20.06	0.131	5.39	0.078	6.51	0.098
Only DS	8.83	0.155	10.79	0.168	19.86	0.124	20.73	0.134	13.89	0.144	13.23	0.147
$T^2inpaint$ (ours)	8.09	0.150	9.82	0.165	18.63	0.122	19.65	0.133	4.76	0.080	5.68	0.099

with mask rates of 10% to 50%, and the result is the average of all mask rates.

4.2. Comparison Results

1) *Quantitative comparison on datasets of 100 training images:* We compare the Only DS (domain-specific transfer) and entirety of our proposed method $T^2inpaint$ with the baselines and report the results on six datasets with only 100 training samples in Table 1 and Table 2. $T^2inpaint$ achieves superior performance on the FID metric for different source and target domains. On the LPIPS metric, $T^2inpaint$ also performs well on most datasets and is comparable to Fine-tuning on the MetFace and CLEVR datasets. In some cases, the performance of the Only DS surpasses Fine-tuning, which requires updating all parameters, demonstrating the effectiveness of adjusting domain-specific parameters.

From the results on all datasets, it is clear that leveraging knowledge learned from the source domain is greatly beneficial for training models from limited target data. For the CLEVR dataset with a strong geometric structure, model reuse can bring more than a twofold performance advantage. It is worth not-

Table 2: Quantitative evaluation on six 256×256 datasets with only 100 samples. We report the FID and LPIPS metrics, with lower values indicating better performance. Except for the Scratch, which trains from scratch, the other methods are initialized with a pre-trained checkpoint from the Places-Challenge [15] dataset. Run with three different random seeds and take the average of the test results. **Bold** text indicates the best performance, and blue text indicates the second best.

Places-Challenge →	AFHQ Cat(256×256)				AFHQ Dog(256×256)				AFHQ Wild(256×256)			
	Medium masks		Thick masks		Medium masks		Thick masks		Medium masks		Thick masks	
Method	FID↓	LPIPS↓	FID↓	LPIPS↓	FID↓	LPIPS↓	FID↓	LPIPS↓	FID↓	LPIPS↓	FID↓	LPIPS↓
Scratch	9.63	0.138	13.09	0.175	14.62	0.144	21.09	0.163	6.04	0.153	7.74	0.160
Fine-tuning	6.39	0.103	7.96	0.122	9.33	0.110	13.74	0.134	3.45	0.115	4.18	0.126
Only DS	6.28	0.103	7.97	0.123	9.31	0.111	13.98	0.135	3.64	0.115	4.28	0.127
T^2 inpaint(ours)	6.20	0.102	7.73	0.122	9.17	0.110	13.36	0.134	3.35	0.113	3.97	0.125
Places-Challenge →	Church(256×256)				MetFace(256×256)				CLEVR(256×256)			
	Medium masks		Thick masks		Medium masks		Thick masks		Medium masks		Thick masks	
Method	FID↓	LPIPS↓	FID↓	LPIPS↓	FID↓	LPIPS↓	FID↓	LPIPS↓	FID↓	LPIPS↓	FID↓	LPIPS↓
Scratch	10.36	0.162	12.69	0.177	25.77	0.147	25.80	0.151	8.34	0.105	9.07	0.119
Fine-tuning	5.00	0.118	6.26	0.139	15.57	0.111	17.34	0.124	3.65	0.068	4.59	0.088
Only DS	4.89	0.118	6.15	0.139	15.55	0.112	17.60	0.126	4.99	0.088	5.54	0.107
T^2 inpaint(ours)	4.89	0.117	6.14	0.138	15.24	0.111	17.02	0.124	3.50	0.067	4.44	0.089

ing that when the source domain is Places-Challenge and the target domain is LSUN-Church, as they have similar distributions, transfer achieves more than a twofold performance improvement. Comparing the results of transferring from different source domains to different target domains, i.e., Tables 1 and 2, we find that pre-training on a source domain with more rich data is more helpful. However, due to a large amount of data in the Places-Challenge (8 million samples) and the fact that the Big LaMa model trained on it has twice as many parameters as LaMa, it is more difficult to optimize the pre-trained model on limited target data. Therefore, our method achieves greater improvement when the source domain is CelebA-HQ compared to when the source domain is Places-Challenge.

2) *Qualitative Comparison*: The qualitative results of our proposed method T^2 inpaint compared to the baselines are shown in Figure 1. Each row corresponds to one dataset; each column from left to right represents the original image with mask, and the results inpainted by the Scratch, Fine-tuning, and T^2 inpaint, respectively. Scratch produces blurred and ghosted results, which demonstrates the catastrophic consequences of training a model from scratch

when there are very few target data, such as 100 training samples in this setting. When initialized with a model trained on large source data, Fine-tuning produces more reasonable inpainting results. However, they still have many issues, such as the defects in the cat’s eye, the dark dog’s and fox’s eyes, the messed up building, the dissonant faces, and the inconsistent geometric structure as shown in Figure 1. Our $T^2inpaint$ method generates high-quality inpainted results on the test sets of all six datasets, including the sharp cat’s ear, the more normal-looking cat’s eye, the brighter and more realistic dog’s and fox’s eyes, the windows with building characteristics, the more artistic facial features, and the consistent geometric structure. This demonstrates further improvement in both semantic alignment and texture details of our method.

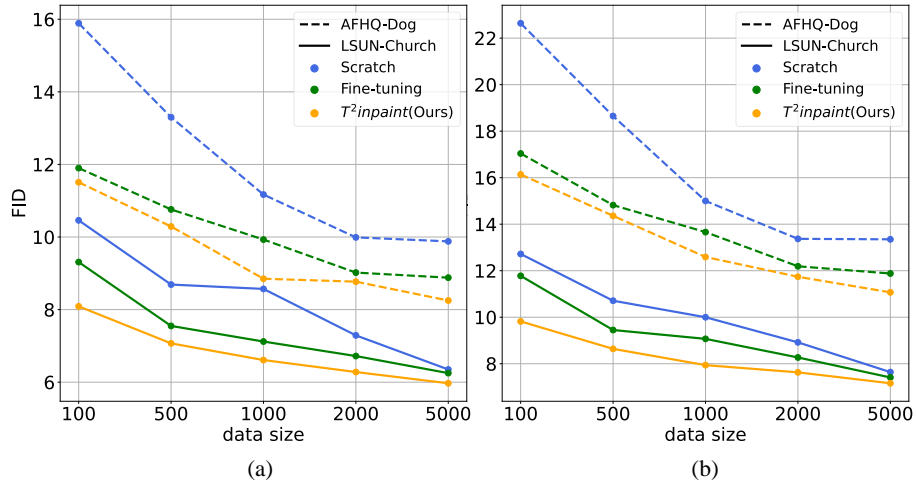


Figure 3: Quantitative evaluation of different data scales on two 256×256 datasets. We report the FID score of the images generated on the test set using (a) the medium mask generation and (b) the thick mask generation, with lower values indicating better performance. The dashed lines represent the results tested on the AFHQ-Dog [19] dataset, while the solid lines represent the results tested on the LSUN-Church [60] dataset; the blue, green, and orange lines represent the results of the Scratch, Fine-tuning, and our $T^2inpaint$ methods, respectively, on different data scales. Except for Scratch, the rest are initialized with the pre-trained models on CelebA-HQ [14].

3) *Comparison under different data sizes:* To understand the influence of

Table 3: Comparison with the method BSA [54] for image generation on samll data. The source domain is CelebA-HQ [14] and the target domain is AFHQ-Dog [19], with only 100 training samples.

Method	Medium masks		Thick masks	
	FID↓	LPIPS↓	FID↓	LPIPS↓
BSA [54]	14.57	0.141	21.34	0.161
Ours	<u>11.51</u>	<u>0.125</u>	<u>16.14</u>	<u>0.145</u>

different data scales on transferability in image inpainting, we conducted further experiments with training sets of 100, 500, 1000, 2000, and 5000 samples, as illustrated in Figure 3. It shows the trend of the FID metric on the test sets of two datasets for different methods under different data sizes. For the target domain AFHQ-Dog, which has facial features similar to the source domain CelebA-HQ, patterns from the source domain are more helpful when the data size in the target domain is smaller; for the completely unrelated target domain LSUN-Church, the improvement brought by pre-training is not so significant, but at this point, our method can fully exploit the potential effective features in the pre-trained model to obtain better performance. As the size of the target data increases, the benefits of transfer learning decrease, and the transfer to domains with a larger gap in data distribution is particularly evident. However, under any experimental setting, $T^2_{inpaint}$ achieves the best performance, which demonstrates the robustness of $T^2_{inpaint}$ for different domains and data scales.

4) *Comparison with the method that fine-tuning partial parameters:* To further demonstrate the effectiveness of our transfer manner, we compare against the method that fine-tuning partial parameters for image generation. BSA [54] discover that different object categories correspond to different scale and shift parameters within the batch statistics, hence it only updated the weights of the batch normalization layers during transfer. As shown in Table 3, our method substantially outperforms BSA in terms of FID and LPIPS metrics, demonstrating that our two-stage fashion is more suitable for the image inpainting task under limited data.

Table 4: Ablation studies on each component of our $T^2\text{inpaint}$. The source domain is CelebA-HQ [14] and the target domain is AFHQ-Cat [19], with only 100 training samples. Fine-tuning and Only DS are one-stage methods, while DS then Fine-tuning and $T^2\text{inpaint}$ are two-stage methods. DS stands for domain-specific transfer.

Method	Medium masks		Thick masks	
	FID↓	LPIPS↓	FID↓	LPIPS↓
Fine-tuning	7.76	0.118	9.47	0.136
Only DS	8.11	0.120	9.87	0.137
DS then Fine-tuning	7.50	0.114	9.25	0.132
$T^2\text{inpaint}$ (ours)	7.32	0.113	9.07	0.131

4.3. Ablation Studies

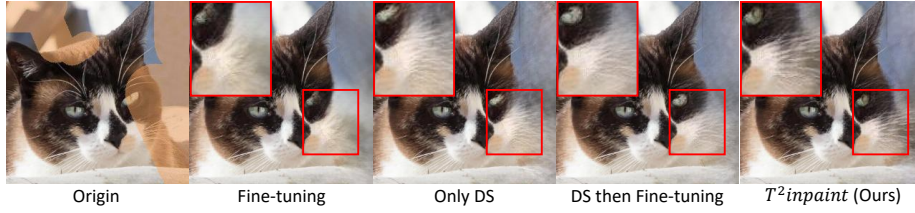


Figure 4: Qualitative results of ablation for each component. The source domain is CelebA-HQ [14] and the target domain is AFHQ-Cat [19], with only 100 training samples. Fine-tuning and Only DS are one-stage methods, while DS then Fine-tuning and $T^2\text{inpaint}$ are two-stage methods. DS stands for domain-specific transfer.

1) *Ablation studies on each component of $T^2\text{inpaint}$:* To reveal the contribution of each component to our method, we conduct experiments on four approaches. (a) **Fine-tuning**: using a pre-trained model and fine-tuning all parameters on the target data as the baseline; (b) **Only DS**: adjusting only the domain-specific parameters, demonstrating the effect of learning global features; (c) **DS then Fine-tuning**: based on (b), adjusting all parameters on the target data, showing the necessity of performing DS before Fine-tuning; (d) **$T^2\text{inpaint}$** : revealing the contribution of using the adapter in the Stage 2. As shown in Figure 4, Only DS can obtain clearer whiskers of the cat than Fine-tuning, indicating that Fine-tuning has difficulty in learning domain-specific

Table 5: Ablation studies on domain-specific blocks. In this experimental setting, we select the most similar distribution of target domains for different source domains. The results show that using the Global branch as the domain-specific part is the most suitable.

CelebA-HQ \rightarrow	MetFace(256 \times 256)			
	Medium masks		Thick masks	
	FID↓	LPIPS↓	FID↓	LPIPS↓
Trainable Part				
Sp-Tr	20.65	0.125	21.55	0.135
Global	19.86	0.124	20.73	0.134
Local&Global	20.06	0.123	21.08	0.133
Places-Challenge \rightarrow	Church(256 \times 256)			
	Medium masks		Thick masks	
	FID↓	LPIPS↓	FID↓	LPIPS↓
Trainable Part				
Sp-Tr	4.93	0.118	6.26	0.138
Global	4.89	0.118	6.15	0.139
Local&Global	4.90	0.118	6.18	0.139

features; DS then Fine-tuning shows better semantics of the cat’s eyes than both Fine-tuning and Only DS, and it also outperforms Fine-tuning in the quantitative metric (as shown in Table 4), which is the benefit of the two-stage approach; $T^2inpaint$ further improves on details by training proposed parallel-connected adapters, as it can generate more realistic cat eyes and more distinct whiskers in Figure 4, and achieve the best performance in Table 4.

2) *Which are domain-specific blocks?* We select multiple combinations of blocks, where only the weights of these blocks will be changed during transfer, to identify domain-specific blocks. Intuitively, for source and target domains with highly similar data distribution, it is sufficient to change only the domain-specific part to adapt to the target domain. Therefore, as shown in Table 5, when the source domain is CelebA-HQ [14], we select the most facial-style MetFace [61] as the target domain; when the source domain is Places-Challenge [15], we select the most architectural-style LSUN-Church [60] as the target domain. For the trainable part, we select three combinations of blocks: the Sp – Tr module, which is the most critical module in the inpainting network (see Sec. 3.3 for detailed analysis), the Global branch containing the Sp – Tr and Conv modules

Table 6: Ablation studies on the structure of the adapter. We try three adapter structures and attempt to reduce parameters using group convolution operations. The results show that our designed Conv – Sp – Tr structure is the best. Where groups equal 1 means no grouping.

Adapter	groups	Medium masks		Thick masks	
		FID↓	LPIPS↓	FID↓	LPIPS↓
Matrix	1	11.00	0.134	14.72	0.160
Conv	1	6.51	0.105	8.04	0.125
Conv-Sp-Tr	1	6.16	0.102	7.69	0.122
Conv-Sp-Tr	2	6.34	0.103	7.96	0.124
Conv-Sp-Tr	4	6.48	0.104	8.21	0.125
Conv-Sp-Tr	8	6.44	0.104	8.33	0.126

(which we use in the Stage 1), and the entire FFC layer containing the Local and Global branches (whose clear structure can be seen in Figure 2). The results in Table 5 show that the best FID score is obtained when the trainable part is Global, and the LPIPS score is almost the same as the best score. When the trainable part is extended from Sp – Tr to Global, the FID score improves significantly; while it is extended from Global to Local&Global, the FID score deteriorates, and the LPIPS score is comparable. Therefore, the Global branch is the domain-specific part, which proves the correctness of our selection and demonstrates that blindly increasing the number of trainable parameters is not optimal during transfer.

3) *What is the best structure of the adapter?* Our method introduces the parallel connection of adapters in Stage 2, and there can be many variations in the specific structure of the adapters. We consider three structures: Matrix, a learnable matrix tensor from [29]; Conv, a 3x3 convolutional layer; Conv – Sp – Tr, parallel connection of Conv to the existing convolutional module and parallel connection of Sp – Tr to the same existing Sp – Tr module (which we used, as shown clearly in Figure 2). The Conv – Sp – Tr structure we adopted achieves the best performance, as shown in Table 6. Changing the adapter structure from Matrix to Conv brings about a significant improvement, indicating that the simple Matrix structure is not feasible for dense prediction tasks such as

generating images. The further improvement from Conv to Conv – Sp – Tr demonstrates the crucial role of the Sp – Tr module in inpainting network. Based on the Conv – Sp – Tr structure, we attempt to use a groupwise convolution operation with a group size k as in [66] to reduce added parameters. The results in Table 6 show that the quality of inpainting decreases as the number of groups k increases, making this approach infeasible.

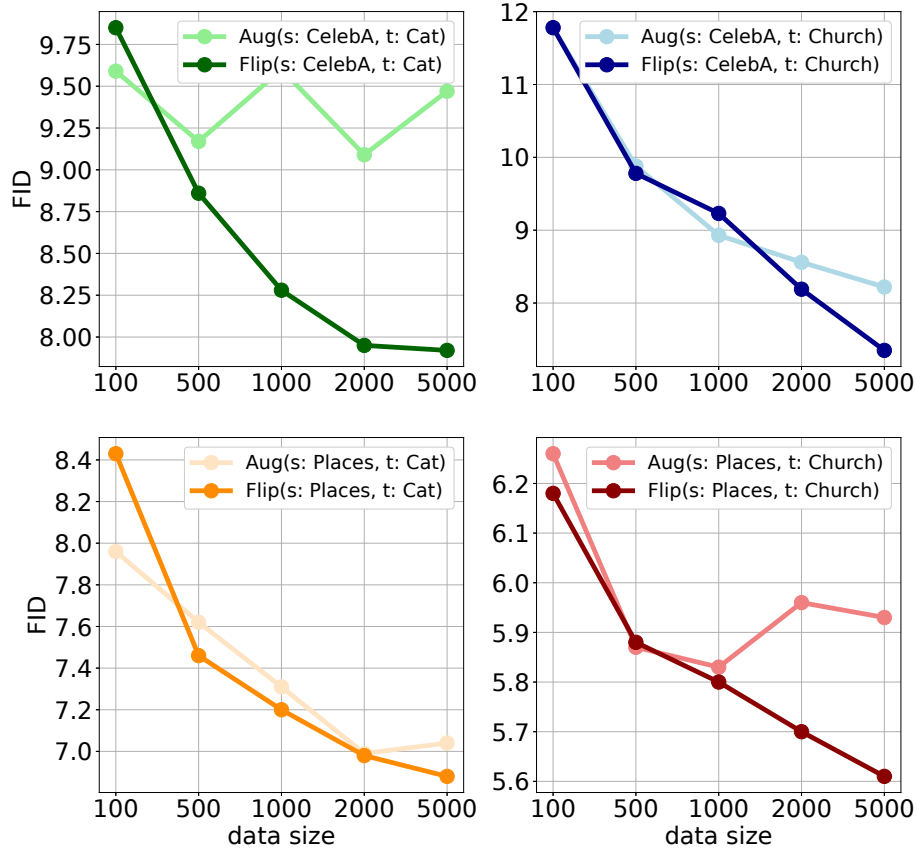


Figure 5: Ablation studies on data augmentation at different data scales. Aug refers to the results obtained using data augmentation, while Flip refers to the results obtained using random horizontal flipping. s denotes the source domain, while t denotes the target domain. The fine-tuning method is used here.

4) *When is it appropriate to use data augmentation?* We consider whether simple data augmentation techniques would be beneficial for image inpainting

Table 7: Results of using different transfer methods on the image inpainting model FcF [21]. The source domain is CelebA-HQ [14] while the target domains are AFHQ-Dog [19] and LSUN Church [60], with only 100 training samples.

CelebA-HQ \rightarrow	AFHQ Dog		Church	
Method	FID↓	LPIPS↓	FID↓	LPIPS↓
FcF [21] + Fine-tuning	13.82	0.129	24.77	0.152
FcF [21] + Ours	13.74	0.128	24.19	0.152

under limited data, and conduct ablation studies on different source and target domains under different data scales, as shown in Figure 5. Aug refers to augmentation techniques, including perspective transformation, affine transformation, contrast limited adaptive histogram equalization, brightness transformation, hue and saturation transformation; Flip represents random horizontal flipping of the input image. From Figure 5, it can be observed that when the data is very limited (i.e., only 100 training samples), the use of data augmentation does not cause the optimization direction to become incorrect (the worst being when the source domain is Places-Challenge and the target domain is LSUN-Church, with an increase of 0.08 in FID value when using data augmentation), and sometimes brings a noticeable gain (the best being when the source domain is Places-Challenge and the target domain is AFHQ-Cat, with a reduction of 0.47 in FID value when using data augmentation). As the amount of data increases, the use of augmentation techniques generally leads to a decrease in performance, and even to a situation where the results are worse when the data scale increases (such as the upper left of Figure 5). This is because when data is very scarce, data augmentation can act as a kind of regularization, but it can also disrupt the semantics of the image [67], so its performance deteriorates as the data increases. Therefore, for image inpainting tasks, we recommend using data augmentation when the data scale is 100 or below, and not using it when the data scale is above 100.

5) *Compatibility of our method.* We integrate our proposed techniques into different baselines to investigate the compatibility. Concretely, many works

adopt the FFC structure in LaMa [10] for image inpainting, among which FcF
 415 [21] combines LaMa and CoModGAN [9] to achieve superior inpainted results.
 We attempt to apply our transfer manner to FcF [21]. In the first stage, we only
 train the generator’s layers that process the spatial sizes from 4×4 to 64×64
 (here starting from 4×4 because the encoder in FcF’s generator encodes the
 image to this spatial size), and for the FFC Block, we only release its global
 420 branch. FcF’s FFC Blocks span resolutions from 32×32 to 256×256 , thus
 these blocks can generate both coarse structure and fine details. For simplicity,
 in the second stage, we only attach new layers with the same configuration to
 all Conv and Sp – Tr in FFC Blocks, and only train the newly added layers
 while freezing the encoder and mapping network in FcF’s generator. As Table
 425 7 shows, our method performs better than Fine-tuning, which indicates the
 flexibility of $T^2inpaint$ and its applicability to future emerging large models.

4.4. Why it works?

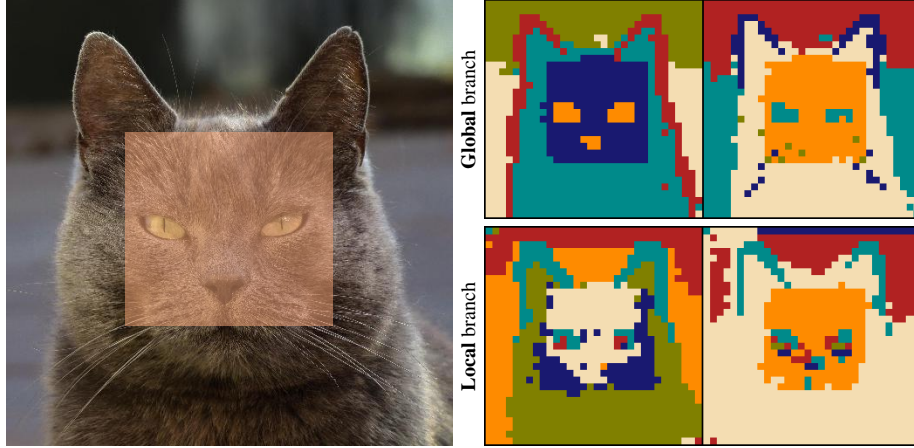


Figure 6: Visualization of the global and local branch feature maps using k -means ($k = 6$) clustering. The left side shows the input image with mask, while the right side displays the result of clustering the feature maps of the two branches. This visualized feature map reveals that the global consistency of the global branch and more detailed information in the local branch.

1) *Two clear-cut branches.* We use visualization to intuitively understand

the characteristics of the global branch and the local branch in the intermediate
 430 *bottleneck* layer of the model, as shown in Figure 6. We apply k -means cluster-
 ing [68] to the outputs of the last two blocks of the trained model’s *bottleneck*
 layer, with the number of categories for both branches set to $k = 6$. From Fig-
 ure 6, it can be observed that the feature maps of the global branch will have
 clearer semantic information than the local branch. For example, the pixels in
 435 the eye region of the feature maps of the global branch are all clustered into
 one, while there are more categories in the local branch. Therefore, the global
 branch is responsible for handling the global structure and the local branch is
 responsible for handling details, which is an intuitive reason for the success of
 Stage 1 in our method.

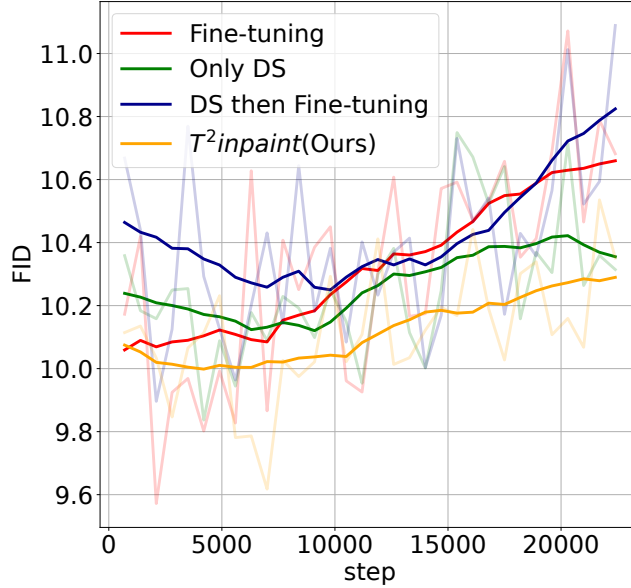


Figure 7: Visualization of the training process for different methods. The source domain is Places-Challenge [15] and the target domain is LSUN-Church [60]. We report the FID values tested on the validation set during the training process, and use a Savitzky-Golay filter [69] to better observe the trend. Our proposed method alleviates overfitting and achieves better performance during the training process.

440 2) *Mitigating overfitting.* To illustrate the effectiveness of maintaining pre-

trained features during transfer, we record the FID values on the validation set during training for different methods, as shown in Figure 7. When transferring from the source domain Places-Challenge to the target domain LSUN-Church, fine-tuning all parameters can destroy good pre-trained features due to the small data scale, as the two domains have similar data distributions. Fine-tuning and DS then Fine-tuning in Figure 7 both exhibit severe overfitting. Our proposed approaches, domain-specific transfer in Stage 1 and better feature-reuse in Stage 2, both alleviate overfitting (shown as the green and orange lines in Figure 7). In addition, our method $T^2inpaint$ mostly achieves the lowest FID values throughout the training process.

5. Conclusions

This paper presents the first attempt to explore improving the inpainted quality from limited data. Fine-tuning, a commonly used method in transfer learning, is first attempted and shows that using a pre-trained model can significantly improve performance when data size is small compared to training from scratch. Based on our experimental results, we find that fine-tuning is not perfect and there is room for improvement. Therefore, this paper proposes the transfer-based two-stage inpainting method. Our method can learn the global structure and fine details through domain-specific transfer in the first stage and better feature-reuse in the second stage, which fine-tuning fail to achieve. We conduct extensive experiments and demonstrate that $T^2inpaint$ is successful in both qualitative and quantitative results. Our research focuses on image inpainting but we hope the method can be extended to other tasks that use encoder-decoder structures, such as image super-resolution from limited training data.

Acknowledgment

This work is supported by Shanghai Science and Technology Program “Federated based cross-domain and cross-task incremental learning” under Grant No.

21511100800, Shanghai Science and Technology Program “Distributed and generative few-shot algorithm and theory research” under Grant No. 20511100600,
 470 Natural Science Foundation of China under Grant No. 62076094, Chinese Defense Program of Science and Technology under Grant No.2021-JCJQ-JJ-0041, and China Aerospace Science and Technology Corporation Industry-University-Research Cooperation Foundation of the Eighth Research Institute under Grant
 475 No.SAST2021-007.

References

References

- [1] Z. Wan, B. Zhang, D. Chen, P. Zhang, D. Chen, J. Liao, F. Wen, Bringing old photos back to life, in: proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020, pp. 2747–2757. 1
 480
- [2] Z. Wan, B. Zhang, D. Chen, P. Zhang, D. Chen, F. Wen, J. Liao, Old photo restoration via deep latent space translation, IEEE Transactions on Pattern Analysis and Machine Intelligence. 1
- [3] C. Barnes, E. Shechtman, A. Finkelstein, D. B. Goldman, Patchmatch: A randomized correspondence algorithm for structural image editing, ACM Trans. Graph. 28 (3) (2009) 24. 1, 2.1
 485
- [4] Y. Li, Y. Li, J. Lu, E. Shechtman, Y. J. Lee, K. K. Singh, Contrastive learning for diverse disentangled foreground generation, in: European Conference on Computer Vision, Springer, 2022, pp. 334–351. 1
- [5] H. Li, W. Wang, C. Yu, S. Zhang, Swapinpaint: Identity-specific face inpainting with identity swapping, IEEE Transactions on Circuits and Systems for Video Technology 32 (7) (2022) 4271–4281. 1
 490
- [6] D. Kim, S. Woo, J.-Y. Lee, I. S. Kweon, Deep video inpainting, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 5792–5801. 1
 495

- [7] K. Nazeri, E. Ng, T. Joseph, F. Z. Qureshi, M. Ebrahimi, Edgeconnect: Generative image inpainting with adversarial edge learning, arXiv preprint arXiv:1901.00212. 1, 2.1
- [8] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, T. S. Huang, Free-form image inpainting with gated convolution, in: Proceedings of the IEEE/CVF international conference on computer vision, 2019, pp. 4471–4480. 1, 2.1
- [9] S. Zhao, J. Cui, Y. Sheng, Y. Dong, X. Liang, E. I. Chang, Y. Xu, Large scale image completion via co-modulated generative adversarial networks, arXiv preprint arXiv:2103.10428. 1, 2.1, 4.3
- [10] R. Suvorov, E. Logacheva, A. Mashikhin, A. Remizova, A. Ashukha, A. Silvestrov, N. Kong, H. Goka, K. Park, V. Lempitsky, Resolution-robust large mask inpainting with fourier convolutions, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2022, pp. 2149–2159. 1, 2.1, 3.3, 3.3, 4.1, 4.3
- [11] Q. Dong, C. Cao, Y. Fu, Incremental transformer structure enhanced image inpainting with masking positional encoding, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 11358–11368. 1, 2.1
- [12] A. Lugmayr, M. Danelljan, A. Romero, F. Yu, R. Timofte, L. Van Gool, Repaint: Inpainting using denoising diffusion probabilistic models, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 11461–11471. 1, 2.1
- [13] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, B. Ommer, High-resolution image synthesis with latent diffusion models, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 10684–10695. 1, 2.1
- [14] T. Karras, T. Aila, S. Laine, J. Lehtinen, Progressive growing of gans for

improved quality, stability, and variation, arXiv preprint arXiv:1710.10196.

1, 1, 3.2, 4.1, 1, 3, 3, 4, 4, 4.3, 7

- 525 [15] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, A. Torralba, Places: A 10 million image database for scene recognition, *IEEE transactions on pattern analysis and machine intelligence* 40 (6) (2017) 1452–1464. 1, 4.1, 2, 4.3, 7
- [16] F. Zhuang, Z. Qi, K. Duan, D. Xi, Y. Zhu, H. Zhu, H. Xiong, Q. He, A comprehensive survey on transfer learning, *Proceedings of the IEEE* 109 (1) 530 (2020) 43–76. 1, 2.2
- [17] M. Oquab, L. Bottou, I. Laptev, J. Sivic, Learning and transferring mid-level image representations using convolutional neural networks, in: *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 1717–1724. 1
- 535 [18] J. Johnson, B. Hariharan, L. Van Der Maaten, L. Fei-Fei, C. Lawrence Zitnick, R. Girshick, Clevr: A diagnostic dataset for compositional language and elementary visual reasoning, in: *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2901–2910. 1, 4.1
- [19] Y. Choi, Y. Uh, J. Yoo, J.-W. Ha, Stargan v2: Diverse image synthesis 540 for multiple domains, in: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 8188–8197. 1, 4.1, 3, 3, 4, 4, 7
- [20] P. Kulshreshtha, B. Pugh, S. Jiddi, Feature refinement to improve high resolution image inpainting, arXiv preprint arXiv:2206.13644. 1, 2.1
- 545 [21] J. Jain, Y. Zhou, N. Yu, H. Shi, Keys to better image inpainting: Structure and texture go hand in hand, in: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2023, pp. 208–217. 1, 3.3, 7, 4.3
- [22] Y. Guo, H. Shi, A. Kumar, K. Grauman, T. Rosing, R. Feris, Spot-tune: transfer learning through adaptive fine-tuning, in: *Proceedings of* 550

the IEEE/CVF conference on computer vision and pattern recognition, 2019, pp. 4805–4814. 1, 2.2

- 555 [23] Y. Guo, Y. Li, L. Wang, T. Rosing, Adafilter: Adaptive filter fine-tuning for deep transfer learning, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 34, 2020, pp. 4060–4066. 1, 2.2
- [24] U. Evci, V. Dumoulin, H. Larochelle, M. C. Mozer, Head2toe: Utilizing intermediate representations for better transfer learning, in: International Conference on Machine Learning, PMLR, 2022, pp. 6009–6033. 1, 2.2
- 560 [25] Z. Shen, Z. Liu, J. Qin, M. Savvides, K.-T. Cheng, Partial is better than all: Revisiting fine-tuning strategy for few-shot learning, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 35, 2021, pp. 9594–9602. 1, 2.2
- [26] S. Mo, M. Cho, J. Shin, Freeze the discriminator: a simple baseline for fine-tuning gans, in: CVPR AI for Content Creation Workshop, 2020. 1, 565 2.2, 3.3, 4.1
- [27] M. Zhao, Y. Cong, L. Carin, On leveraging pretrained gans for generation with limited data, in: Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event, Vol. 119 of Proceedings of Machine Learning Research, PMLR, 2020, pp. 11340–11351. 570 1, 2.2, 3.3
- [28] A. Kumar, A. Raghunathan, R. M. Jones, T. Ma, P. Liang, Fine-tuning can distort pretrained features and underperform out-of-distribution, in: The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022, 2022. 1, 2.2, 3.2, 3.4
- 575 [29] W.-H. Li, X. Liu, H. Bilen, Cross-domain few-shot learning with task-specific adapters, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 7161–7170. 1, 2.2, 3.4, 4.3

- [30] C.-B. Zhang, J.-W. Xiao, X. Liu, Y.-C. Chen, M.-M. Cheng, Representation compensation networks for continual semantic segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 7053–7064. 1, 3.4
- [31] C. Ballester, M. Bertalmio, V. Caselles, G. Sapiro, J. Verdera, Filling-in by joint interpolation of vector fields and gray levels, IEEE transactions on image processing 10 (8) (2001) 1200–1211. 2.1
- [32] M. Bertalmio, L. Vese, G. Sapiro, S. Osher, Simultaneous structure and texture image inpainting, IEEE transactions on image processing 12 (8) (2003) 882–889. 2.1
- [33] Z. Xu, J. Sun, Image inpainting by patch propagation using patch sparsity, IEEE transactions on image processing 19 (5) (2010) 1153–1165. 2.1
- [34] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, A. A. Efros, Context encoders: Feature learning by inpainting, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 2536–2544. 2.1
- [35] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. C. Courville, Y. Bengio, Generative adversarial nets, in: NIPS, 2014. 2.1
- [36] S. Iizuka, E. Simo-Serra, H. Ishikawa, Globally and locally consistent image completion, ACM Transactions on Graphics (ToG) 36 (4) (2017) 1–14. 2.1
- [37] F. Yu, V. Koltun, Multi-scale context aggregation by dilated convolutions, arXiv preprint arXiv:1511.07122. 2.1
- [38] Y. Song, C. Yang, Y. Shen, P. Wang, Q. Huang, C.-C. J. Kuo, Spg-net: Segmentation prediction and guidance network for image inpainting, arXiv preprint arXiv:1805.03356. 2.1

- [39] S. Xu, D. Liu, Z. Xiong, E2i: Generative inpainting from edge to image, IEEE Transactions on Circuits and Systems for Video Technology 31 (4) (2020) 1308–1322. 2.1
- [40] C. Wang, X. Chen, S. Min, J. Wang, Z.-J. Zha, Structure-guided deep video inpainting, IEEE Transactions on Circuits and Systems for Video Technology 31 (8) (2021) 2953–2965. 2.1
- [41] H. Liu, B. Jiang, Y. Xiao, C. Yang, Coherent semantic attention for image inpainting, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 4170–4179. 2.1
- [42] G. Liu, F. A. Reda, K. J. Shih, T.-C. Wang, A. Tao, B. Catanzaro, Image inpainting for irregular holes using partial convolutions, in: Proceedings of the European conference on computer vision (ECCV), 2018, pp. 85–100. 2.1
- [43] C. Wang, M. Shao, D. Meng, W. Zuo, Dual-pyramidal image inpainting with dynamic normalization, IEEE Transactions on Circuits and Systems for Video Technology 32 (9) (2022) 5975–5988. 2.1
- [44] L. Chi, B. Jiang, Y. Mu, Fast fourier convolution, Advances in Neural Information Processing Systems 33 (2020) 4479–4488. 2.1
- [45] Z. Lu, J. Jiang, J. Huang, G. Wu, X. Liu, Glama: Joint spatial and frequency loss for general image inpainting, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 1301–1310. 2.1
- [46] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, S. Ganguli, Deep unsupervised learning using nonequilibrium thermodynamics, in: International Conference on Machine Learning, PMLR, 2015, pp. 2256–2265. 2.1
- [47] X. Shan, Y. Lu, Q. Li, Y. Wen, Model-based transfer learning and sparse coding for partial face recognition, IEEE Transactions on Circuits and Systems for Video Technology 31 (11) (2021) 4347–4356. 2.2

- [48] A. Sendjasni, M.-C. Larabi, F. A. Cheikh, Convolutional neural networks for omnidirectional image quality assessment: A benchmark, *IEEE Transactions on Circuits and Systems for Video Technology* 32 (11) (2022) 7301–7316. 2.2
- [49] A. Graves, A.-r. Mohamed, G. Hinton, Speech recognition with deep recurrent neural networks, in: 2013 IEEE international conference on acoustics, speech and signal processing, Ieee, 2013, pp. 6645–6649. 2.2
- [50] M. Wortsman, G. Ilharco, S. Y. Gadre, R. Roelofs, R. Gontijo-Lopes, A. S. Morcos, H. Namkoong, A. Farhadi, Y. Carmon, S. Kornblith, et al., Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time, in: International Conference on Machine Learning, PMLR, 2022, pp. 23965–23998. 2.2
- [51] M. Wortsman, G. Ilharco, J. W. Kim, M. Li, S. Kornblith, R. Roelofs, R. G. Lopes, H. Hajishirzi, A. Farhadi, H. Namkoong, et al., Robust fine-tuning of zero-shot models, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 7959–7971. 2.2
- [52] R. Gontijo-Lopes, Y. Dauphin, E. D. Cubuk, No one representation to rule them all: Overlapping features of training methods, *arXiv preprint arXiv:2110.12899*. 2.2
- [53] Y. Wang, C. Wu, L. Herranz, J. van de Weijer, A. Gonzalez-Garcia, B. Raducanu, Transferring gans: generating images from limited data, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 218–234. 2.2
- [54] A. Noguchi, T. Harada, Image generation from small datasets via batch statistics adaptation, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 2750–2758. 2.2, 3, 4.2
- [55] Y. Wang, A. Gonzalez-Garcia, D. Berga, L. Herranz, F. S. Khan, J. v. d. Weijer, Minegan: effective knowledge transfer from gans to target domains

with few images, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 9332–9341. 2.2

[56] Y. Li, R. Zhang, J. Lu, E. Shechtman, Few-shot image generation with elastic weight consolidation, arXiv preprint arXiv:2012.02780. 2.2

665 [57] U. Ojha, Y. Li, J. Lu, A. A. Efros, Y. J. Lee, E. Shechtman, R. Zhang, Few-shot image generation via cross-domain correspondence, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 10743–10752. 2.2

[58] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, et al., Overcoming catastrophic forgetting in neural networks, Proceedings of the national academy of sciences 114 (13) (2017) 3521–3526. 2.2

675 [59] U. Evci, V. Dumoulin, H. Larochelle, M. C. Mozer, Head2toe: Utilizing intermediate representations for better transfer learning, in: International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA, Vol. 162 of Proceedings of Machine Learning Research, PMLR, 2022, pp. 6009–6033. 3.3

[60] F. Yu, A. Seff, Y. Zhang, S. Song, T. Funkhouser, J. Xiao, Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop, arXiv preprint arXiv:1506.03365. 4.1, 3, 4.3, 7, 7

[61] T. Karras, M. Aittala, J. Hellsten, S. Laine, J. Lehtinen, T. Aila, Training generative adversarial networks with limited data, Advances in Neural Information Processing Systems 33 (2020) 12104–12114. 4.1, 4.3

685 [62] B. Liu, Y. Zhu, K. Song, A. Elgammal, Towards faster and stabilized gan training for high-fidelity few-shot image synthesis, in: International Conference on Learning Representations, 2021. 4.1

- [63] A. Sauer, K. Chitta, J. Müller, A. Geiger, Projected gans converge faster, Advances in Neural Information Processing Systems 34 (2021) 17480–17492. 4.1
- 690 [64] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, O. Wang, The unreasonable effectiveness of deep features as a perceptual metric, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 586–595. 4.1
- 695 [65] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, S. Hochreiter, Gans trained by a two time-scale update rule converge to a local nash equilibrium, Advances in neural information processing systems 30. 4.1
- [66] S. Varshney, V. K. Verma, P. Srijith, L. Carin, P. Rai, Cam-gan: Continual adaptation modules for generative adversarial networks, Advances in Neural Information Processing Systems 34 (2021) 15175–15187. 4.3
- 700 [67] A. Kolesnikov, L. Beyer, X. Zhai, J. Puigcerver, J. Yung, S. Gelly, N. Houlsby, Big transfer (bit): General visual representation learning, in: European conference on computer vision, Springer, 2020, pp. 491–507. 4.3
- [68] G. Lee, H. Kim, J. Kim, S. Kim, J.-W. Ha, Y. Choi, Generator knows what discriminator should learn in unconditional gans, in: European Conference on Computer Vision, Springer, 2022, pp. 406–422. 4.4
- 705 [69] A. Savitzky, M. J. Golay, Smoothing and differentiation of data by simplified least squares procedures., Analytical chemistry 36 (8) (1964) 1627–1639. 7

Declaration of interests

☒The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

☐The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: