Restorable Image Operators with Quasi-Invertible Networks

Hao Ouyang*, **Tengfei Wang***, **Qifeng Chen**

The Hong Kong University of Science and Technology

Abstract

Image operators have been extensively applied to create visually attractive photos for users to share processed images on social media. However, most image operators often smooth out details or generate textures after the processing, which removes the original content and raises challenges for restoring the original image. To resolve this issue, we propose a quasi-invertible model that learns common image processing operators in a restorable fashion: the learned image operators can generate visually pleasing results with the original content embedded. Our model is trained on input-output pairs that represent an image processing operator's behavior and uses a network that consists of an invertible branch and a non-invertible branch to increase our model's approximation capability. We evaluate the proposed model on ten image operators, including detail enhancement, abstraction, blur, photographic style, and non-photorealistic style. Extensive experiments show that our approach outperforms relevant baselines in the restoration quality, and the learned restorable operator is fast in inference and robust to compression. Furthermore, we demonstrate that the invertible operator can be easily applied to practical applications such as restorable human face retouching and highlight preserved exposure adjustment.

Introduction

Image operators are broadly utilized for sharing visually appealing images on the internet as social media plays an increasingly important role in daily life. As shown in the second row of Fig. 1, there is a diverse set of image operators used in practice nowadays. However, a prevalent issue is how to deal with the original image after we have the processed version. On the one hand, users are often reluctant to keep the original image as it roughly doubles the storage requirement (on a mobile phone). On the other hand, we may need the original image for inspection or alternative editing in the future. Being able to restore the original image from the processed one will be the desired solution. In this paper, we explore the problem of restorable image operators, which performs high-quality image processing while the original image can be restored from the processed one.

The challenges in learning a restorable operator exist in two main aspects. As shown in Fig. 1, after the processing, the output image tends to lose important details or inject new texture patterns. When restoring the original image, the former case raises the difficulty of recovering correct image details, and the latter case increases the possibility of creating undesired content because of the artificially injected texture patterns. Although several previous works have investigated using deep neural networks to accelerate image operators (Chen, Xu, and Koltun 2017) or simulate specific invertible process (e.g., RGB2Gray (Xiao et al. 2020)), directly applying these works leads to low restoration quality as in Sec. . In this paper, we seek for a general framework for a broad range of image operators with the possibility of faithfully restoring the original image.

Our proposed quasi-invertible network depends on the recent advancement of the deep invertible network (Kingma and Dhariwal 2018). Specifically, we adopt the model with invertible split-and-transform-based layers (Dinh, Sohl-Dickstein, and Bengio 2017) to approximate the image processing operators. However, we empirically find that existing invertible models fail to approximate an image operator if it loses many details (e.g., L_0 smooth). We thus design a quasi-invertible scheme with a non-invertible branch that significantly increases the approximation capacity. Since the model is nearly invertible, we can restore the original image by reverse inference in theory. However, due to the information losses caused by computational inaccuracy, quantization and compression, the restored image may contain severe noises and artifacts. Therefore, we adopt a training strategy that learns these degradation jointly with the backward restoration loss. This greatly strengthens the robustness of our model to non-intentional modifications.

We conduct extensive experiments to evaluate the proposed model on ten image processing operators from various categories, including detail enhancement, style transfer and image abstraction. The proposed quasi-invertible model outperforms baselines on both operator approximation quality and image restoration quality . Our model is also fast when inference and restoration: it takes around 75ms on a 480p image. Our contribution can be summarized as follows.

- We demonstrate the feasibility of learning restorable complex image operators. Extensive experiments show that our approximated operator generates perceptually indistinguishable results from the desired operator while restoring high-quality original images.

*Equal contribution

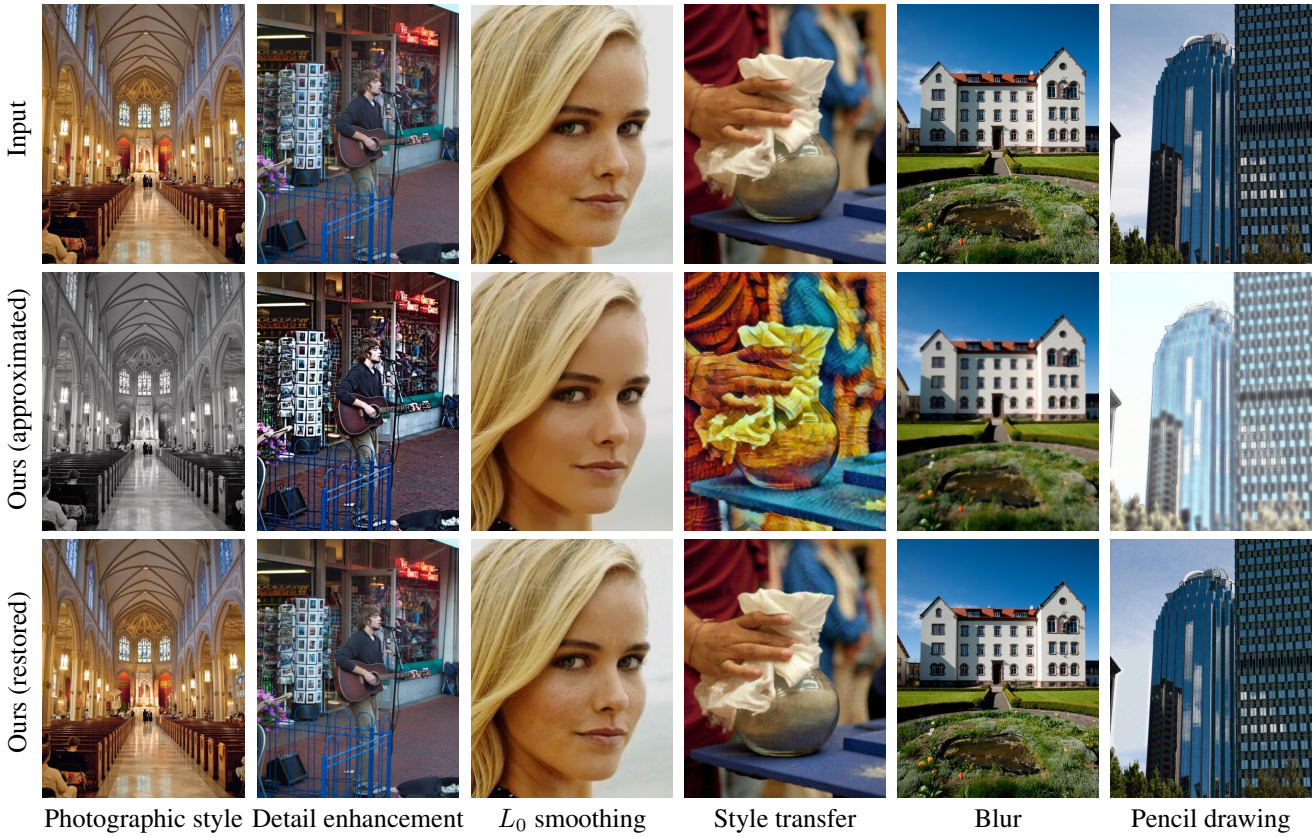


Figure 1: The results of our learning-based restorable image operators. The figure presents the approximation and restoration results of six image processing operators: black-and-white photographic style, detail enhancement, L_0 smooth, style transfer, weighted median filter, and color pencil drawing filter. Our method can perform high-quality image processing while recovering the original image. Zoom in for details.

- The proposed quasi-invertible network greatly benefits the approximation quality, and our scheme is robust in restoration against undesired quantization or compression than the previous methods.
- We show that the proposed scheme can be easily applied to real-life applications such as restorable human face retouching and highlight preserved exposure adjustment.

Related Work

Image Processing Operators

Improving image operators for different low-level vision tasks (Wang, Ouyang, and Chen 2021; Wang et al. 2021) has been a long-standing problem in computer vision research. Previous works have developed various image processing techniques such as detail enhancement (Subr, Soler, and Durand 2009; Farbman et al. 2008; Fattal 2009), edge-preserving image smoothing and abstraction (Xu et al. 2011, 2012), removing undesired noise (Zhang et al. 2014; Zhang, Xu, and Jia 2014), and artistic or photographic style generation (Bae, Paris, and Durand 2006). Recent works have successfully generated stylized images based on neural networks by separating style and content (Gatys, Ecker, and Bethge 2016). Many approaches have been proposed for accelerating these image processing operators. The fully-convolutional networks have shown the capability of approximating, accelerating, and improving them at the same time (Chen, Xu, and Koltun 2017; Fan et al. 2018). In this work, we take a further step: we demonstrate the possibility of using the invertible deep structure to simulate these operators, and thus the high-quality original image can be restored. Several recent works focus on a specific task such as invertible grayscale (Xia, Liu, and Wong 2018), raw-to-srgb image signal processing (Zamir et al. 2020; Xing, Qian, and Chen 2021) and image rescaling (Etmann, Ke, and Schönlieb 2020; Xiao et al. 2020), while we propose a more general scheme for versatile image operators including blur, smoothing, and artistic style generation with the quasi-invertible design. Milanfar (Milanfar 2018) also focuses on recovering black box filtering based on per-image optimization while our method is learning a quasi-invertible neural network for fast inference.

Deep Invertible Networks

Deep invertible networks (Rezende and Mohamed 2015) consist of a sequence of invertible transformations that maps from a simple distribution (e.g., Gaussian) to a complex distribution. Recently invertible models have become a popular choice for high-quality speech (Kim et al. 2019; Prenger,

Valle, and Catanzaro 2019) and image generation. Dinh et al. (Dinh, Krueger, and Bengio 2015; Dinh, Sohl-Dickstein, and Bengio 2017) designed the affine coupling layer, which can integrate arbitrary functions to increase the learning capacity of the model. Glow (Kingma and Dhariwal 2018) further proposed to adopt a 1×1 convolutional layer for information propagation. Ho et al. (Ho et al. 2019) further improved on the limitations on noise generation, inexpressive affine flows, and conditioning networks in coupling layers. In our work, we choose the invertible flow-based model due to the direct access to the inverse mapping.

Image Steganography

Our work is also related to image steganography (Morkel, Eloff, and Olivier 2005), where the original and the processed image can be considered as the secret and the cover image, respectively. However, traditional image steganography methods (Pevnỳ, Filler, and Bas 2010; Tamimi, Abdalla, and Al-Allaf 2013) are subject to relatively low capacity, and hiding the secret image in the cover image with the same resolution is impracticable. Recent deep image steganography methods (Baluja 2017; Zhu et al. 2018; Yang et al. 2019) usually adopt an encoder-decoder structure: the hidden image is encoded into the cover image and retrieved by the decoder. Our proposed scheme has two significant advantages over these methods. Firstly, in our approach, the hiding and image processing steps are performed simultaneously, while extra hiding steps need to be conducted in the steganography-based approach. Secondly, the steganography method is designed for hiding general images where the secret images and the cover images are not related. In our case, the processed images and the original images share similar structures, and using specifically designed architecture can lead to higher restoration quality.

Method

Preliminaries

Let I be the original input image and f be the desired image operator. The desired processed content can be represented by $f(I)$ where the resolutions of $f(I)$ and I are the same. We aim in approximating the operator f with a network $\hat{f}$ such that $f(I) \approx \hat{f}(I)$ for all images and $\hat{f}$ is quasi-invertible. The training pairs can be easily simulated by applying f to the training dataset to obtain $f(I)$ as ground truths. We utilize the simulated pairs for the proposed network to learn the behavior of f. As our goal is a widely applicable architecture for versatile image operators, we divide these operators into three types based on the invertibility:

- **Type I**: nearly lossless operator such as detail enhancement, where the original f itself is almost invertible. Learning a new invertible operator $\hat{f}$ of this type is relatively easy because the task is naturally invertible.
- **Type II**: lossy operator such as image abstraction and smoothing, where the original operator f loses details after filtering. It requires the learned operator $\hat{f}$ to recover the lost details to achieve the restoration.
- **Type III**: generative operator such as style transfer, where the original f synthesizes textures in the processed image. In this case, the newly generated textures should not be propagated to the original image when restoration.

Although approximating restorable operators from Type I is straightforward to achieve, learning an invertible model for operators from Type II and Type III requires extra efforts. The proposed method is to approximate the original operator f with a quasi-invertible network, which should be as invertible as possible. Specifically, our model is mainly composed of the invertible affine coupling layers (Dinh, Sohl-Dickstein, and Bengio 2017). Given an input state h (e.g., feature maps), an affine coupling layer is defined as follows:

$$h_1, h_2 = split(h), (\log s, t) = G(h_1),$$
$$h_2' = s \odot h_2 + t, h' = concat(h_1, h_2'),$$

where $split$ and $concat$ are performed along the channel dimension. G is an arbitrary function, where we use a four-layer convolutional network with LeakyReLU activations (Maas et al. 2013). h' is the output state of this layer.

Quasi-Invertible Network

The difficulty of learning a restorable image operator lies in designing a model that performs both high-quality approximation and restoration. Directly applying existing invertible blocks leads to inaccurate forward approximation for Type II/III operators. On the other hand, we observed that these operators can be approximated with a simple non-invertible CNN structure in (Chen, Xu, and Koltun 2017). To combine the best of both worlds, we propose a **quasi-invertible module** that fuses the main invertible branch and a non-invertible encoder-decoder branch. In this work, we assume the output image $\hat{f}(I)$ is saved with 8-bit integers for each RGB channel using PNG or JPEG, and the saved output can be used to recover the original image. However, the information loss brought by quantization or compression can result in inaccurate restoration. To compensate for this loss, we respectively adopt Quantization module for PNG and **Compression-aware optimization (CO)** for JPEG.

Fig. 2 illustrates the overview architecture of the proposed method in which the blue and red arrows denote the forward approximation and reverse restoration pass, respectively. In the forward approximation stage, the model takes an image I as the input, and generates the output processed image $\hat{f}(I)$. As $\hat{f}(I)$ is saved in PNG or JPEG, we denote the saved output image as $\mu(\hat{f}(I))$, where μ denotes the quantization (and compression). During the restoration, the original image can be restored along the reverse pass.

Quasi-Invertible Module. The proposed architecture comprises two branches: a main invertible branch ϕ composed of affine coupling layers described above, and an auxiliary encoder-decoder branch ψ that improves the approximation capacity. We denote each block in the invertible branch with ϕ_i, and the output of this branch I_ϕ is represented as

$$I_\phi = \phi_k \circ \ldots \circ \phi_2 \circ \phi_1(I), \quad (1)$$

where k is the number of blocks. The auxiliary branch ψ takes I as input, and the output of this branch I_ψ can be represented as $\psi(I)$. The detailed structure of ψ is in the supplement. Considering the inference speed and model size, the

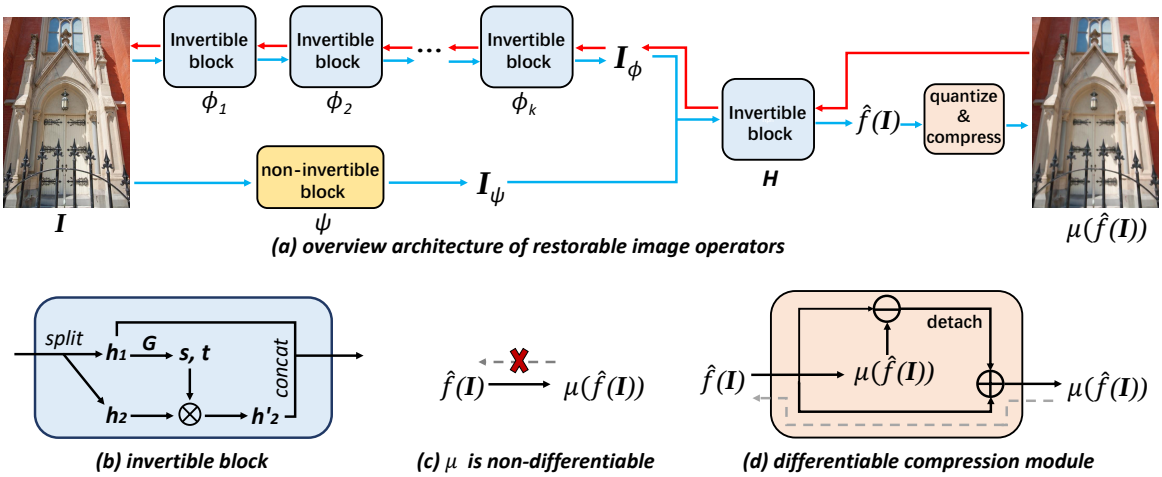


Figure 2: Overview of the proposed quasi-invertible network. In the forward approximation pass (blue arrow), the input image I goes through a two-branch quasi-invertible network and a quantization and compression module. In the reverse restoration pass (red arrow), we can feed the stored output image $\mu(\hat{f}(I))$ in the reverse direction for restoring the original image.

fusion process is only performed in the last block. To fuse these two branches, we concatenate I_ψ and I_ϕ channel-wise and feed it to the invertible block H for output:

$$\hat{f}(I), \Delta = H(I_\phi, I_\psi), \tag{2}$$

where Δ is a dummy image designed to be close to zero. Note that different from most of the previous works using invertible networks (Lugmayr et al. 2020; Xiao et al. 2020),we expect the approximation and restoration process to introduce the least level of randomness. Instead of regularizing Δ (three additional output channels) introduced by the non-invertible branch as Gaussian noises, we directly regularize Δ to approximate zero with a regularization term $\|\Delta\|^2$ during training. In the restoration phase, we set $\Delta = \mathbf{0}$. With the auxiliary branch, the network increases the approximation ability while becomes quasi-invertible as the information in additional channels is discarded. Without considering effect of quantization and compression, the restored image $\hat{I}$ can be directly retrieved from $\hat{f}(I)$ by

$$\hat{I}_\phi, \hat{I}_\psi = H^{-1}(\mu(\hat{f}(I)), \mathbf{0}), \tag{3}$$

$$\hat{I} = \phi_1^{-1} \circ \ldots \circ \phi_k^{-1}(\hat{I}_\phi). \tag{4}$$

Compression-Aware Optimization

Our propose a training strategy called compression-aware optimization to make the framework robust against image quantization and JPEG compression, which is practically important as compressed images are ubiquitous. The distortion caused by JPEG compression can be viewed as a pseudo-noise dependent on the processed image (Zhang et al. 2020). Although the compression is not differentiable (Figure 2c), we can still use the Straight-Through Estimator (Bengio, Léonard, and Courville 2013; Razavi, Oord, and Vinyals 2019) via reparametrization (Figure 2d): $\mu(\hat{f}(I)) = \hat{f}(I) + \epsilon$ where $\epsilon = \mu(\hat{f}(I)) - \hat{f}(I)$, and ϵ can be regarded as pseudo-noise. During forward inference, we compute $\mu(\hat{f}(I)) = \hat{f}(I) + \epsilon$; during backpropagation, we assume that ϵ is a constant and simply use the gradient on $\mu(\hat{f}(I))$ as the gradient on $f(I)$ for gradient descent.

Losses

The quasi-invertible model specifies the correspondence between the desired processed image $f(I)$ and the input image I. We include different optimization objectives for compelling performance on various types of operators.

Approximation Loss. The output image $\hat{f}(I)$ of the learned operator should be as similar to the ground truth processed image $f(I)$ as possible. We use $L2$ loss to minimize the approximation error as $L_a = ||\hat{f}(I) - f(I)||^2$.

Restoration Loss. In real scenarios, to compensate for quantization and compression, we add a reverse restoration regularization. We expect the restored image $\hat{I}$ to be similar to the input image: $L_r = ||\hat{I} - I||^2$.

Adversarial Loss. To improve the style generation for Type III operators, we also adopt the adversarial loss (Goodfellow et al. 2014) on Type III operator outputs.

Experiments

Experimental Settings

Image Operators. We evaluate the proposed method on ten operators from the three types mentioned in Sec. :

Type I: multiscale tone enhancement (WLS) (Farbman et al. 2008) and local Laplacian filtering (LLF) (Paris, Hasinoff, and Kautz 2011).

Type II: relative total variation filter (RTV) (Xu et al. 2012), rolling guidance filter (RGF) (Zhang et al. 2014), fast weighted median filter (WMF) (Zhang, Xu, and Jia 2014), L_0 smoothing (Xu et al. 2011), and black-and-white photographic style (BWP) (Aubry et al. 2014).

Type III: colored pencil drawing (CPD) (Lu, Xu, and Jia 2012), fast style transfer (Rain Princess, Scream).

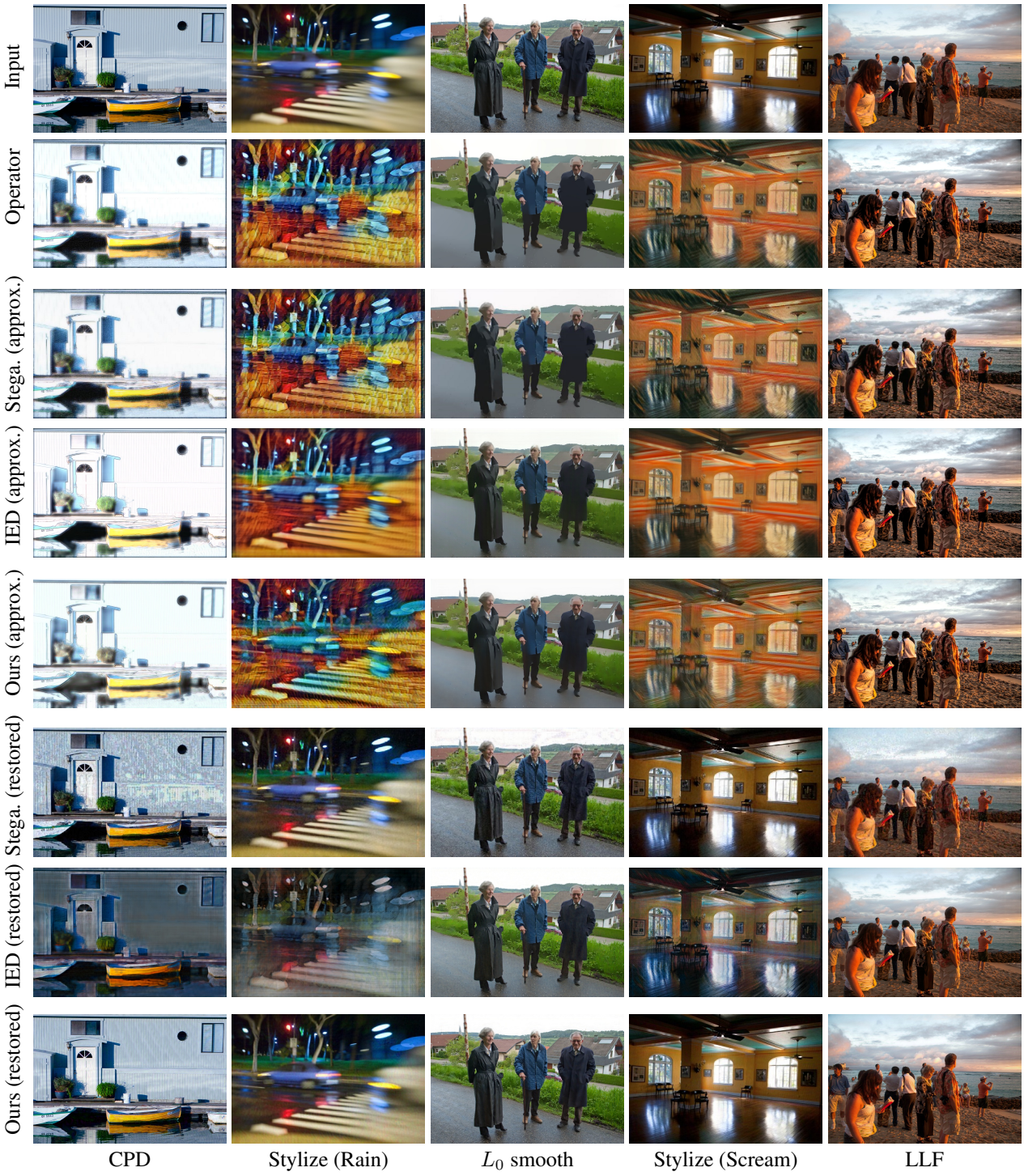


Figure 3: Visual comparisons of different methods on learning various image operators. Our method achieves high-quality approximation and restoration performance. More results on diverse operators are shown in the Supplementary Material.

Baselines. We adapt two previous works as our baselines:

- **Invertible encoder-decoder (IED)** adapted from (Chen, Xu, and Koltun 2017). We initialize two encoder-decoder networks: one maps the input to the processed image, and the other maps the processed image back to the input.
- **Image Steganography (Stega.)** adapted from (Weng et al. 2019). We first process images with original operators as the cover image. We then apply the steganography to hide the original input in the cover image.

Operators	Type	Approximation (PSNR / SSIM)			Restoration (PSNR / SSIM)		
		Ours	IED	Operator + Stega.	Ours	IED	Operator + Stega.
WLS	Type I	**32.58/0.982**	32.45/0.982	26.74/0.920	**37.31/0.985**	31.73/0.970	27.89/0.905
LLF	Type I	32.29/0.976	**35.97/0.993**	27.55/0.927	**38.16/0.988**	34.65/0.984	28.20/0.907
L_0 smooth	Type II	**32.54/0.952**	32.19/0.961	30.59/0.945	**36.46/0.975**	28.30/0.876	28.43/0.906
WMF	Type II	**40.81/0.967**	38.27/0.986	33.77/0.966	**41.25/0.980**	29.65/0.937	28.26/0.903
RTV	Type II	**38.13/0.984**	35.85/0.976	34.06/0.972	**38.76/0.987**	28.98/0.902	28.66/0.915
RGF	Type II	**38.01/0.988**	36.72/0.984	34.09/0.973	**40.06/0.991**	28.47/0.905	28.65/0.915
BWP	Type II	45.86/**0.998**	**47.03**/0.997	30.10/0.936	**40.50/0.991**	25.41/0.917	28.32/0.905
CPD	Type III	15.71/0.748	16.72/0.753	**30.22/0.964**	**32.33/0.946**	14.99/0.573	25.37/0.837
stylize (Rain)	Type III	14.69/0.626	15.37/0.618	**21.54/0.864**	**36.24/0.984**	15.27/0.569	28.17/0.915
stylize (Scream)	Type III	19.30/0.733	19.71/0.769	**31.11/0.941**	**37.84/0.988**	16.45/0.670	28.65/0.926

Table 1: Quantitative comparisons with baselines on different operators on Adobe-MIT 5K test set in the PNG format.

LLF (PSNR / SSIM)	Approximation	Restoration
MIT-Adobe → RAISE	32.17/0.985	36.20/0.992
RAISE → RAISE	31.94/0.972	38.13/0.976
RAISE → MIT-Adobe	30.24/0.950	36.61/0.970
MIT-Adobe → MIT-Adobe	32.29/0.976	38.16/0.988

Table 2: Cross-dataset validation (training set → test set).

Image Quality Evaluation

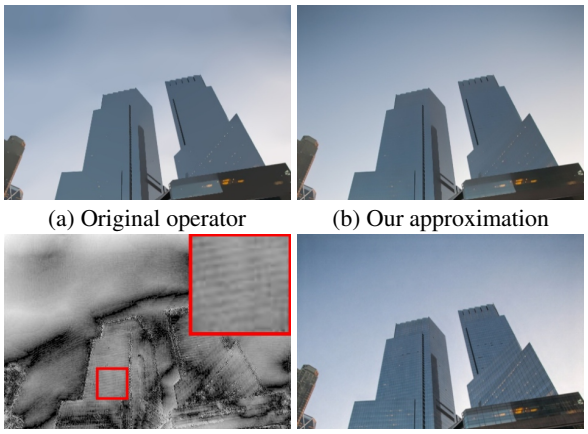
Table 1 shows quantitative results on Adobe-MIT 5K.

- **Approximation.** Our method achieves comparable performance with baselines in terms of operator approximation. Note that for Type III generative operators, PSNR/SSIM do not necessarily measure the image processing quality as the synthesized textures can be random (Huang and Belongie 2017). The metrics of the steganography-based method on Type III operators are much higher because the ground-truth processed images are used as the cover images for hiding and thus the output images match the ground-truth textures. Fig. 3 provides a qualitative comparison, which shows that the perceptual quality of different methods is comparable.
- **Restoration.** Our method achieves comparable restoration results with baselines for Type I operators, which are naturally invertible. For hard cases such as Type II and III, the proposed model achieves significantly stronger performance than others. As in Fig. 3, the restoration of our method can recover the original details and contains less noise compared with other methods.

Analysis

Hidden Details

As our method can recover high-quality images even for the challenging smoothing tasks that should inevitably lose many details, an interesting question is how the network can restore the same details as the original one. We thus magnify the difference map between the original processed image and our approximated image. As shown in Fig. 4 (c), although the difference is not visible in the normal condition,



(a) Original operator (b) Our approximation

(c) Magnified error map (d) Our restoration

Figure 4: Magnified error map between (a.) original L_0 smoothing result and (b.) our approximated result. Darker color indicates smaller difference. Zoom in for details.

Parameters	L_0 smooth / σ			LLF / λ		
	0.01	0.02	0.04	0.1	0.25	1.0
Approximation	32.83	32.37	31.84	32.53	32.29	32.16
Restoration	37.18	36.11	35.28	37.20	37.14	37.00

Table 3: PSNR with different hyper-parameters.

we observe the "missing" details after applying the magnification (e.g., windows in the building/textures in the cloud). It indicates that during the training, the quasi-invertible network learns to approximate the operator while simultaneously hiding the restoration cues in the processed images.

Cross-Dataset Validation

We evaluate the generalization ability of learned operators by cross-dataset validation. We apply the model trained on MIT-Adobe 5K to RAISE (Dang-Nguyen et al. 2015) and also the model trained on RAISE to MIT-Adobe 5K. Table 2 shows the quantitative results, which indicate that

L_0 smooth (PSNR / SSIM)	Approximation	Restoration
Ours (remove restoration loss)	32.10/0.946	16.69/0.672
Ours (remove auxiliary branch)	29.78/0.921	43.32/0.993
Ours (8 $\rightarrow$ 4 invertible blocks)	31.26/0.947	35.08/0.971
Ours (full)	32.54/0.952	36.46/0.975

Table 4: Ablation study on different modules. Ours (full) is composed of 8 invertible blocks and an auxiliary non-invertible branch, and trained with full loss.

L_0 smooth	JPEG-70	JPEG-80	JPEG-90
Ours (w/o CO)	26.48/0.652	26.91/0.669	28.06/0.690
Ours (w/ CO)	26.95/0.711	27.78/0.743	28.87/0.786

Table 5: Ablation study on the compression quality. We save the L_0 smoothing images with different JPEG qualities and report the PSNR/SSIM of the restoration results.

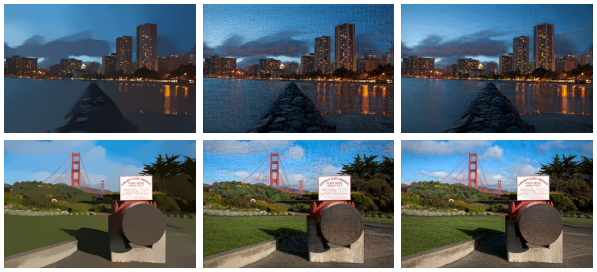
our method achieves comparable performance on the new dataset with the original one. More qualitative results are shown in the supplement. Both quantitative and visual results show that the trained operators can generalize well to other data distributions.

Controlled Experiments

Effect of Variable Hyper-Parameters. For some operators, the hyper-parameters can be adjusted on an image-by-image basis. To study the robustness of our method, we evaluate the proposed method on various parameters. Following previous work (Chen, Xu, and Koltun 2017), we concatenate a hyper-parameter channel with the original image as the new input of the network. As shown in Table 3, this simple technique can achieve a coarse control of parameters for Type I/II operators with only one model. Future work can further introduce adaptive modules (Fan et al. 2018) to improve the approximation accuracy.

Ablation Study. Table 4 shows that without the restoration loss, the restoration quality decreases sharply. The results also show that without the auxiliary branch, the model suffers degradation in faithfully approximating operators. However, the restoration quality can drop if we include this branch. We attach more visual results without this branch in the supplement, which show that the improvement in the approximation is perceptually obvious, and the loss in restoration is comparably negligible. If we reduce the number of invertible blocks from 8 to 4, both approximation and restoration qualities drop, but the inference time is reduced from 75ms to 50ms. It remains to be discussed how to choose the best model achieving the balance among the approximation quality, restoration quality, and inference time.

We also study the effect of different JPEG qualities. Fig. 5 demonstrates that the restoration contains strong noise without the compression-aware optimization. Table 5 shows that for different JPEG compression qualities, our method achieves consistent restoration results.



JPEG approximated Restored (w/o CO) Restored (w/ CO)

Figure 5: Restoration results on JPEG-80 approximated images with and without compression-aware optimization.

Input Ours (approximated) Ours (restored)

Figure 6: Visual results of two applications: restorable face retouching and exposure adjustment. Zoom in for details.

Applications

Restorable Human Face Retouching. A straightforward application of our method is to retouch human faces without losing details. As shown in Fig. 6, by applying our learned edge-aware smoothing filter, we can remove the undesired face spots. However, different from traditional non-invertible operators, which often damage the original contents, we can recover the original input images with high fidelity by reverse inference.

Highlight Preserved Exposure Adjustment. Exposure adjustment, especially increasing the brightness, is a common cause of losing details for photographers or Photoshop/Snapshot users, as the overexposed pixels are permanently clipped. In Fig. 6, we show the results of brightening the image twice, and our model learns to hide the details in highlight in the processed image for restoration.

Conclusion

We explore the possibility of learning a restorable image operator for general fast image processing, and approach this novel problem with a quasi-invertible network. Extensive experiments show promising results on approximating ten image operators with the high-quality restoration of input images. However, the robustness of the model still suffer from non-intentional degradation or using arbitrary parameterizations. We expect future work to further improve the robustness and capacity of restorable image operators.

References

Aubry, M.; Paris, S.; Hasinoff, S. W.; Kautz, J.; and Durand, F. 2014. Fast local laplacian filters: Theory and applications. *ACM Transactions on Graphics (TOG)*, 33(5): 1–14.

Bae, S.; Paris, S.; and Durand, F. 2006. Two-scale tone management for photographic look. *ACM Transactions on Graphics (TOG)*, 25(3): 637–645.

Baluja, S. 2017. Hiding images in plain sight: Deep steganography. In *Conference on Neural Information Processing Systems (NeurIPS)*.

Bengio, Y.; Léonard, N.; and Courville, A. 2013. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv:1308.3432*.

Chen, Q.; Xu, J.; and Koltun, V. 2017. Fast image processing with fully-convolutional networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

Dang-Nguyen, D.-T.; Pasquini, C.; Conotter, V.; and Boato, G. 2015. Raise: A raw images dataset for digital image forensics. In *Proceedings of the 6th ACM multimedia systems conference*.

Dinh, L.; Krueger, D.; and Bengio, Y. 2015. NICE: Nonlinear Independent Components Estimation. In Bengio, Y.; and LeCun, Y., eds., *The International Conference on Learning Representations (ICLR) Workshops*.

Dinh, L.; Sohl-Dickstein, J.; and Bengio, S. 2017. Density estimation using real nvp. In *The International Conference on Learning Representations (ICLR)*.

Etmann, C.; Ke, R.; and Schönlieb, C.-B. 2020. iUNets: Fully invertible U-Nets with learnable up-and downsampling. *arXiv:2005.05220*.

Fan, Q.; Chen, D.; Yuan, L.; Hua, G.; Yu, N.; and Chen, B. 2018. Decouple learning for parameterized image operators. In *European Conference on Computer Vision (ECCV)*.

Farbman, Z.; Fattal, R.; Lischinski, D.; and Szeliski, R. 2008. Edge-preserving decompositions for multi-scale tone and detail manipulation. *ACM Transactions on Graphics (TOG)*, 27(3): 1–10.

Fattal, R. 2009. Edge-avoiding wavelets and their applications. *ACM Transactions on Graphics (TOG)*, 28(3): 1–10.

Gatys, L. A.; Ecker, A. S.; and Bethge, M. 2016. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Goodfellow, I. J.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial networks. *Conference on Neural Information Processing Systems (NeurIPS)*.

Ho, J.; Chen, X.; Srinivas, A.; Duan, Y.; and Abbeel, P. 2019. Flow++: Improving flow-based generative models with variational dequantization and architecture design. In *International Conference on Machine Learning (ICML)*.

Huang, X.; and Belongie, S. 2017. Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

Kim, S.; Lee, S.-g.; Song, J.; Kim, J.; and Yoon, S. 2019. FloWaveNet: A generative flow for raw audio. *International Conference on Machine Learning (ICML)*.

Kingma, D. P.; and Dhariwal, P. 2018. Glow: Generative Flow with Invertible 1x1 Convolutions. In *Conference on Neural Information Processing Systems (NeurIPS)*.

Lu, C.; Xu, L.; and Jia, J. 2012. Combining sketch and tone for pencil drawing production. In *Proceedings of the symposium on non-photorealistic animation and rendering*.

Lugmayr, A.; Danelljan, M.; Van Gool, L.; and Timofte, R. 2020. Srflow: Learning the super-resolution space with normalizing flow. In *European Conference on Computer Vision (ECCV)*, 715–732.

Maas, A. L.; Hannun, A. Y.; Ng, A. Y.; et al. 2013. Rectifier nonlinearities improve neural network acoustic models. In *International Conference on Machine Learning (ICML)*.

Milanfar, P. 2018. Rendition: Reclaiming What a Black Box Takes Away. *SIAM J. Imaging Sci.*, 11(4): 2722–2756.

Morkel, T.; Eloff, J. H.; and Olivier, M. S. 2005. An overview of image steganography. In *ISSA*, volume 1.

Paris, S.; Hasinoff, S. W.; and Kautz, J. 2011. Local Laplacian filters: Edge-aware image processing with a Laplacian pyramid. *ACM Transactions on Graphics (TOG)*, 30(4): 68.

Pevnỳ, T.; Filler, T.; and Bas, P. 2010. Using high-dimensional image models to perform highly undetectable steganography. In *International Workshop on Information Hiding*.

Prenger, R.; Valle, R.; and Catanzaro, B. 2019. Waveglow: A flow-based generative network for speech synthesis. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 3617–3621.

Razavi, A.; Oord, A. v. d.; and Vinyals, O. 2019. Generating Diverse High-Fidelity Images with VQ-VAE-2. In *Conference on Neural Information Processing Systems (NeurIPS)*.

Rezende, D.; and Mohamed, S. 2015. Variational inference with normalizing flows. In *International Conference on Machine Learning (ICML)*.

Subr, K.; Soler, C.; and Durand, F. 2009. Edge-preserving multiscale image decomposition based on local extrema. *ACM Transactions on Graphics (TOG)*, 28(5): 1–9.

Tamimi, A. A.; Abdalla, A. M.; and Al-Allaf, O. 2013. Hiding an image inside another image using variable-rate steganography. *International Journal of Advanced Computer Science and Applications (IJACSA)*, 4(10).

Wang, T.; Ouyang, H.; and Chen, Q. 2021. Image Inpainting With External-Internal Learning and Monochromic Bottleneck. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 5120–5129.

Wang, T.; Xie, J.; Sun, W.; Yan, Q.; and Chen, Q. 2021. Dual-Camera Super-Resolution With Aligned Attention Modules. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2001–2010.

Weng, X.; Li, Y.; Chi, L.; and Mu, Y. 2019. High-capacity convolutional video steganography with temporal residual modeling. In *Proceedings of the 2019 on International Conference on Multimedia Retrieval*.

Xia, M.; Liu, X.; and Wong, T.-T. 2018. Invertible grayscale. *ACM Transactions on Graphics (TOG)*, 37(6): 1–10.

Xiao, M.; Zheng, S.; Liu, C.; Wang, Y.; He, D.; Ke, G.; Bian, J.; Lin, Z.; and Liu, T.-Y. 2020. Invertible image rescaling. In *European Conference on Computer Vision (ECCV)*.

Xing, Y.; Qian, Z.; and Chen, Q. 2021. Invertible Image Signal Processing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Xu, L.; Lu, C.; Xu, Y.; and Jia, J. 2011. Image smoothing via L 0 gradient minimization. *ACM Transactions on Graphics (TOG)*, 30(6): 174.

Xu, L.; Yan, Q.; Xia, Y.; and Jia, J. 2012. Structure extraction from texture via relative total variation. *ACM Transactions on Graphics (TOG)*, 31(6): 1–10.

Yang, H.; Ouyang, H.; Koltun, V.; and Chen, Q. 2019. Hiding Video in Audio via Reversible Generative Models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

Zamir, S. W.; Arora, A.; Khan, S.; Hayat, M.; Khan, F. S.; Yang, M.-H.; and Shao, L. 2020. Cycleisp: Real image restoration via improved data synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Zhang, C.; Karjauv, A.; Benz, P.; and Kweon, I. S. 2020. Towards Robust Data Hiding Against (JPEG) Compression: A Pseudo-Differentiable Deep Learning Approach. *arXiv:2101.00973*.

Zhang, Q.; Shen, X.; Xu, L.; and Jia, J. 2014. Rolling guidance filter. In *European Conference on Computer Vision (ECCV)*.

Zhang, Q.; Xu, L.; and Jia, J. 2014. 100+ times faster weighted median filter (WMF). In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Zhu, J.; Kaplan, R.; Johnson, J.; and Fei-Fei, L. 2018. Hidden: Hiding data with deep networks. In *European Conference on Computer Vision (ECCV)*.