electronics

MDPI

Article

A Codec-Unified Deblurring Approach Based on U-Shaped Invertible Network with Sparse Salient Representation in Latent Space

Meng Wang [1,2,*], Tao Wen [1,2] and Haipeng Liu [1,2]

[1] Faculty of Information Engineering and Automation, Kunming University of Science and Technology, Kunming 650500, China; wentao997540054@163.com (T.W.); ran@kust.edu.cn (H.L.)
[2] Yunnan Key Laboratory of Artificial Intelligence, Kunming University of Science and Technology, Kunming 650500, China
* Correspondence: wangmeng@kmust.edu.cn

Abstract: Existing deep learning architectures usually use a separate encoder and decoder to generate the desired simulated images, which is inefficient for feature analysis and synthesis. Aiming at the problem that the existing methods fail to fully utilize the correlation of codecs, this paper focuses on the codec-unified invertible networks to accurately guide the image deblurring process by controlling latent variables. Inspired by U-Net, a U-shaped multi-level invertible network (UML-IN) is proposed by integrating the wavelet invertible networks into a supervised U-shape architecture to establish the multi-resolution correlation between blurry and sharp image features under the guidance of hybrid loss. Further, this paper proposes to use $L1$ regularization constraints to obtain sparse latent variables, thereby alleviating the information dispersion problem caused by high-dimensional inference in invertible networks. Finally, we fine-tune the weights of invertible modules by calculating a similarity loss between blur-sharp variable pairs. Extensive experiments on real and synthetic blurry sets show that the proposed approach is efficient and competitive compared with the state-of-the-art methods.

Keywords: invertible networks; image deblurring; U-Net; multi-resolution correlations; $L1$ regularization; similarity loss

Citation: Wang, M.; Wen, T.; Liu, H. A Codec-Unified Deblurring Approach Based on U-Shaped Invertible Network with Sparse Salient Representation in Latent Space. *Electronics* **2022**, *11*, 2177. https://doi.org/10.3390/electronics11142177

Academic Editor: Dah-Jye Lee

Received: 21 June 2022
Accepted: 5 July 2022
Published: 12 July 2022

1. Introduction

The purpose of image deblurring is to restore a low-quality degraded image to a high-quality image with sharp spatial details. An efficient deblurring method can not only enhance visual perception, but also assist with high-level vision tasks such as image classification [1] and object detection [2]. However, image deblurring is a highly ill-posed problem because there are infinitely feasible solutions. In order to constrain the solution space to valid images, early deblurring methods typically use empirical observations to handcraft image priors to improve image quality [3–7]. In recent years, with the successful application of deep learning [2,8–10], the deblurring methods based on convolutional neural networks (CNNs) that implicitly learn more general priors by capturing the statistical information of natural images from large-scale data have developed rapidly [11–16].

Compared with earlier methods, the CNN-based methods have significantly improved the model's performance, which is mainly due to the diversity of generative framework design. At present, the main solutions include the module structures of single decoding, codec separation and codec-unified. The representative of model design based on single decoding is GAN, which has mature applications in image deblurring tasks [17–21]. GAN maps the input noise (i.e., latent variables) to the generated results. The former is usually set to obey the Gaussian distribution or uniform distribution independent of the training data (or application scenarios). However, the research of Karras et al. [22] showed that the generated results obtained by using the noise constrained by a prior distribution were

not accurate, and it also implies that there are limitations in the model design of single decoding. Thereafter, the coding module is applied, and the structure design based on codec separation has attracted extensive attention [23–26]. In 2020, Kaufman et al. [27] used two independent modules to encode and decode images respectively, and introduced the U-shaped structure that could help information recovery at the same level into image deblurring. Furthermore, An et al. [28] introduced the blur kernel adaptive block into the deblurring task on the basis of codec separation. The model structure of codec separation alleviates the problem of insufficient noise constraints to a certain extent, and improves the deblurring performance. However, it does not take full advantage of the strong correlation in the process of image encoding and decoding, so the commonly applied strategy of separately training the encoder and decoder is inefficient, which is not conducive to compact feature analysis and synthesis.

Recently, the codec-unified invertible networks have re-entered the limelight with their amazing generation quality and efficient training methods [29,30]. The invertible networks map the data distribution perfectly to a Gaussian distribution, and are exact mapping models with no information loss. Moreover, the networks can be optimized in both directions in one training, so the training process is efficient. Although invertible networks have many advantages, the existing research on them is mainly focused on the unsupervised domain, and to our knowledge there is no precedent for using invertible networks to implement image deblurring. Besides, in practical training, we found that the invertible networks requiring the same number of dimensions of input and output would disperse the salient information and thus be detrimental to the learning of image details. These issues are what the present study aims to overcome.

Overall, the above approaches that employ different generative frameworks have three major limitations:

- Non-codec-unified architectures are not conducive to analyze and reconstruct features efficiently and accurately;
- The unsupervised invertible networks cannot be directly applied to supervised deblurring tasks;
- The massive dimension learning of invertible networks can influence the utilization efficiency of salient detail features.

To handle these limitations, this paper focuses on introducing the invertible networks with a codec-unified framework into deblurring scenarios. One challenge lies in the input features being blurry while the output features should be sharp, so the two corresponding independent distributions should be efficiently analyzed and transferred. Inspired by wavelet flow [31], wavelet decomposition can decompose the image losslessly into high-frequency features and low-frequency features. It only maps high-frequency features that affect image sharpness (such as contours and details) to latent variables, and can reduce the computational effort of the model. Thus, we expand this module of wavelet flow to a fully invertible U-network similar to the framework of U-net [32]. In this paper, the invertible networks are nested in two invertible wavelet transforms to decompose and map the blurry and sharp images respectively and establish correlations in the latent space, thus enabling the transferring of blurry features to sharp features. Furthermore, two invertible wavelet flows are used, which not only introduce the invertible networks to image deblurring, but also allow for a multi-level image analysis and reconstruct. Finally, we attempt to sparse the latent variables in different levels in order to optimize the information distribution and thus alleviate the inefficiency caused by massive dimension learning of invertible networks. Extensive experiments verified that our proposed deblurring model performs better in a variety of scenarios than other models. The main contributions in this paper can be summarized as follows:

- A novel framework for image deblurring is proposed using invertible network modules with a codec-unified structure. For the first time, the wavelet invertible network is introduced into the deblurring tasks, making full use of the correlation between coding and decoding to compactly guide the image reconstruction in the latent space;

- A U-shaped multi-level invertible network (UML-IN) is developed by integrating the wavelet invertible networks into a two-branch U-shaped supervised architecture instead of the existing single-branch unsupervised schemes. Therefore, the multi-level feature learning of the two branches corresponds to each other, and the sharp-feature branch explicitly guides the learning of the blur-feature branch in the same level of latent space;
- $L1$ regularization is applied to alleviate the defect of dimensional redundancy in the original invertible networks. The model invertibility requires the same number of input and output dimensions, which causes information dispersion that is not conducive to represent salient image details. Therefore, we introduce sparse regularization for the learning of latent variables in order to aggregate these high-quality features. The experimental results illustrate that the proposed model achieves excellent performance in a variety of visual perception scenarios.

The rest of the paper is organized as follows. Section 2 briefly reviews the research background and provides the basic background information. Section 3 introduces the principle of the model in detail. Section 4 illuminates the comparison results through in depth analysis. Finally, we conclude the study in Section 5.

2. Research Background

The invertible networks are exact mapping models that map the data distribution to a Gaussian distribution perfectly. In 2014, Dinh et al. first proposed NICE [33] for learning a highly non-linear bijective transformation that maps the training data to a space where its distribution is approximately factorized, and a framework to achieve this by directly maximizing log-likelihood. Two years later, Dinh et al. extended NICE using real-valued non-volume preserving (real NVP [34]) transformations, and introduced a new multi-scale structure and a variable splitting method, and most of the subsequent studies were based on the expansion of the Real NVP idea. The original invertible networks are implemented by mapping the input images into latent variables of the same number of dimensions. Let $x \in R^D$ be an input image and $z \in R^D$ be a latent variable obeying standard Gaussian distribution. According to [34], If the forward inference map $f : x \mapsto z$ is bijective and the corresponding inverse map is $g \triangleq f^{-1}$, then we can obtain this map with an equation using the criterion of maximum likelihood as follows:

$$p_X(x) = p_Z(z)\left|\det\left(\frac{\partial g(z)}{\partial z}\right)\right|^{-1}, \tag{1}$$

where $x = g(z)$ and $\mathrm{J}(z) = \partial x/\partial z$ are the Jacobi matrix of x with respect to z. It can be seen that the exact mutual mapping between x and z can be achieved using only the same invertible function. Therefore, the model can be optimized in both the forward and the backward directions simultaneously without information loss, reflecting the efficiency and accuracy of the invertible networks training. In addition, this single maximum likelihood training strategy has advantages compared to the GAN. The maximum likelihood is used to increase the probability of the target sample (and also decrease the probability of the non-target sample), and the discriminator of GAN does nothing more than that. The difference is that GAN calculates the loss using the results obtained by sampling as negative samples with the target samples, however the invertible networks calculate it by treating all non-target samples as negative samples. Therefore, it might imply that the method of maximum likelihood can handle the work of the discriminator and does it more adequately.

The invertibility of layer functions and the calculability of Jacobian determinants are the key issues for the invertible networks. The same input and output dimensions ensure the invertibility of the map functions. According to Real-NVP [34], in order to achieve the easy computation of the Jacobian determinant, the invertible networks flatly divide the input x into x_1 and x_2 on the channel dimension and then the outputs y_1 and y_2 will be expressed as:

$$\begin{cases} y_1 = x_1 & (a) \\ y_2 = x_2 \odot \exp(s(x_1)) + t(x_1) & (b), \end{cases} \tag{2}$$

where $s(.)$ and $t(.)$ are arbitrarily complex neural networks (i.e., CNNs used to learn features). The derivative of y with respect to x_1 and x_2 is then obtained as a triangular determinant with the diagonal as the product of $\exp(s(x_1))$. High computability of the Jacobian determinant can thus be achieved. The inverse process can also be conveniently performed as $x_2 = (y_2 - t(y_1)) \odot \exp(-s(y_1))$ in terms of this designed coupling layer.

Based on them, in 2018, Kingma et al. proposed Glow [35], an invertible network with 1×11 convolution, which turned the operation of segmenting x_1 and x_2 into a learnable process as well, and eventually achieved an amazing generation quality. In 2019, Ardizzone et al. proposed cINN [36] which combines the purely generative INN model with an unconstrained feed-forward network, thus being able to efficiently preprocess the conditioning input into useful features. In 2020, SRFlow [37] proposed by Andreas et al. explored the diversity of super-resolution images generated by invertible networks and obtained a variety of realistic high-resolution images, demonstrating the feasibility of using invertible networks in recovering high-quality images. Furthermore, Yu et al. [31] introduced the idea of wavelet decomposition into the invertible networks for the first time, forming wavelet invertible networks. Let the high-frequency features $x^{H(l)} = \{x^{h(l)}, x^{v(l)}, x^{d(l)}\}$, and the low-frequency images $x^{(l)}$ are the results of l-th level wavelet decomposition $h_{\varphi}^{(l)}$ for the original input image $x^{(0)} \in R^{H \times W \times C}$, respectively. According to [31], the image decomposition process can be expressed as:

$$(x^{(l)}, x^{H(l)}) = h_{\varphi}^{(l)}(x^{(l-1)}). \tag{3}$$

Yu et al. first mapped the high-frequency feature distribution $p_{x^H}^{(l)}$ at level l-th to the corresponding latent variable distribution $p_z^{(l)}$ through invertible networks, and then used $p_z^{(l)}$ was sampled to obtain $z^{(l)}$ and $x^{H(l)}$ was obtained by inverse mapping $g^{(l)}(.)$. According to [31], the image generation process can be recursively expressed as:

$$\begin{cases} x^{H(l)} = g^{(l)}(z^{(l)}) & (a) \\ x^{(l-1)} = h_{\varphi^{-1}}^{(l)}(x^{(l)}, x^{H(l)}) & (b), \end{cases} \tag{4}$$

where $1 \leq l \leq L$, $h_{\varphi^{-1}}^{(l)}(.)$ is the inverse wavelet transform [38]. Compared with the previous invertible networks, this extended wavelet architecture can obtain lossless multi-level images and reduce the computational complexity of the model. In addition, since features that affect image sharpness (such as contours and details) are usually high-frequency features, wavelet invertible networks are better suited for the task of image deblurring. The above invertible networks perform forward encoding and backward decoding with shared network configurations that unify the learning process. On this basis, image generation can be optimized by controlling latent variables. Therefore wavelet invertible networks can directly learn the latent variables of high-frequency features that are closely related to image sharpness, thus handling many types of blur without priori information.

Considering that unsupervised invertible networks cannot be directly applied to supervised deblurring tasks, this paper expands wavelet invertible networks into a two-branch U-shaped structure (see Section 3.1 for details) and designs a hybrid loss step-by-step guidance model for optimization (see Section 3.2 for details). In addition, the invertible networks require the same dimensions for both the inputs and the outputs, while the learned features actually need more compact dimensional representations. Thus, this paper proposed to use $L1$ sparse regularization to alleviate this problem, which will be presented in Section 3.3.

3. The Proposed Approach for Image Deblurring

3.1. U-Shaped Architecture Using Invertible Networks

The invertible networks utilize a fully shared architecture for both the forward inference procedure $f_\theta(.)$ and the backward reconstruction procedure $g_\theta(.) = f_\theta^{-1}(.)$ with the same network parameters θ, in order to directly control the image generation in the latent space. Based on this, the wavelet invertible networks use a novel framework with multi-resolution analysis to perform the invertible analysis procedures in high-frequency bands that affect the sharpness of the image, such its as contours and details while reducing the computational effort of the model. Therefore, we focus on utilizing this extended architecture for the deblurring tasks in this paper.

Let the blurry image and the corresponding sharp image be $\tilde{x}^{(0)}$ and $x^{(0)}$ respectively, where (0) denotes the original space level. In training phase, we feed these blur-sharp pairs into the U-shaped architecture to guide the learning of invertible networks as shown in Figure 1, and thus obtain the reconstructed sharp image $\hat{x}^{(0)}$ as predictions. The process of multi-resolution decomposition by Wavelets base φ to obtain the respective low-frequency and high-frequency features is illuminated in the following equations.

$$\begin{cases} \{\tilde{x}^{(l)}, \tilde{x}^{h(l)}, \tilde{x}^{v(l)}, \tilde{x}^{d(l)}\} = h_\varphi^{(l)}(\tilde{x}^{(l-1)}) & (a) \\ \{x^{(l)}, x^{h(l)}, x^{v(l)}, x^{d(l)}\} = h_\varphi^{(l)}(x^{(l-1)}) & (b), \end{cases} \tag{5}$$

where $\tilde{x}^{H(l)} = \{\tilde{x}^{h(l)}, \tilde{x}^{v(l)}, \tilde{x}^{d(l)}\}$ and $x^{H(l)} = \{x^{h(l)}, x^{v(l)}, x^{d(l)}\}$ with the decomposing level $1 \leq l \leq L$, which denotes the high-frequency features of the blurry and the sharp images, respectively. Notice that at the last level, the feature $\tilde{x}^{H(L)}$ no longer needs decomposition, and the same operation of $x^{H(L)}$ is discarded. After these Wavelet decomposition, the features $\tilde{x}^{H(l)}$ and $x^{H(l)}$ are fed into the invertible networks to infer the corresponding latent variables as:

$$\begin{cases} \tilde{z}^{(l)} = \tilde{f}^{(l)}(\tilde{x}^{H(l)}) & (a) \\ z^{(l)} = f^{(l)}(x^{H(l)}) & (b), \end{cases} \tag{6}$$

where $\tilde{f}^{(l)}(.)$ and $f^{(l)}(.)$ respectively denote the blur-feature and the sharp-feature invertible maps operated on the l^{-th} level spatial resolution. So far, the forward feature analysis of the proposed U-shaped architecture is completed.

Furthermore, we expect to implement a non-invertible transferring map $h_{\mathrm{T}}^{(l)} : \tilde{z}^{(l)} \mapsto \hat{z}^{(l)}$ to predict the reference label value $z^{(l)}$, which also establishes the correlation between $\tilde{z}^{(l)}$ in the blurry variable space and $z^{(l)}$ in the sharp variable space. This process can be expressed as:

$$\hat{z}^{(l)} = h_{\mathrm{T}}^{(l)}(\tilde{z}^{(l)}), \tag{7}$$

where $\hat{z}^{(l)}$ is the transferring prediction of $z^{(l)}$ by the trainable hidden module $h_{\mathrm{T}}^{(l)}$ at l^{-th} level, optimized by MSE. Therefore, the reconstruction process of the high-frequency features $\hat{x}^{H(l)}$ can be expressed by:

$$\hat{x}^{H(l)} = g^{(l)}(\hat{z}^{(l)}). \tag{8}$$

The reconstruction of the deblurring images at different decomposition levels can be represented as follows:

$$\hat{x}^{(l-1)} = h_{\varphi^{-1}}^{(l)}(\hat{x}^{(l)}, \hat{x}^{H(l)}), \tag{9}$$

where $h_{\varphi^{-1}}^{(l)}(.)$ is the inverse transformation of Wavelet decomposition and the $\hat{x}^{(L)}$ at the last level is initialized by the $\tilde{x}^{(L)}$ as a direct connection contained in the U-shaped architecture as shown in Figure 1. By recursively applying Equation (9), this proposed architecture hierarchically reconstructs the final deblurring prediction $\hat{x}^{(0)}$.

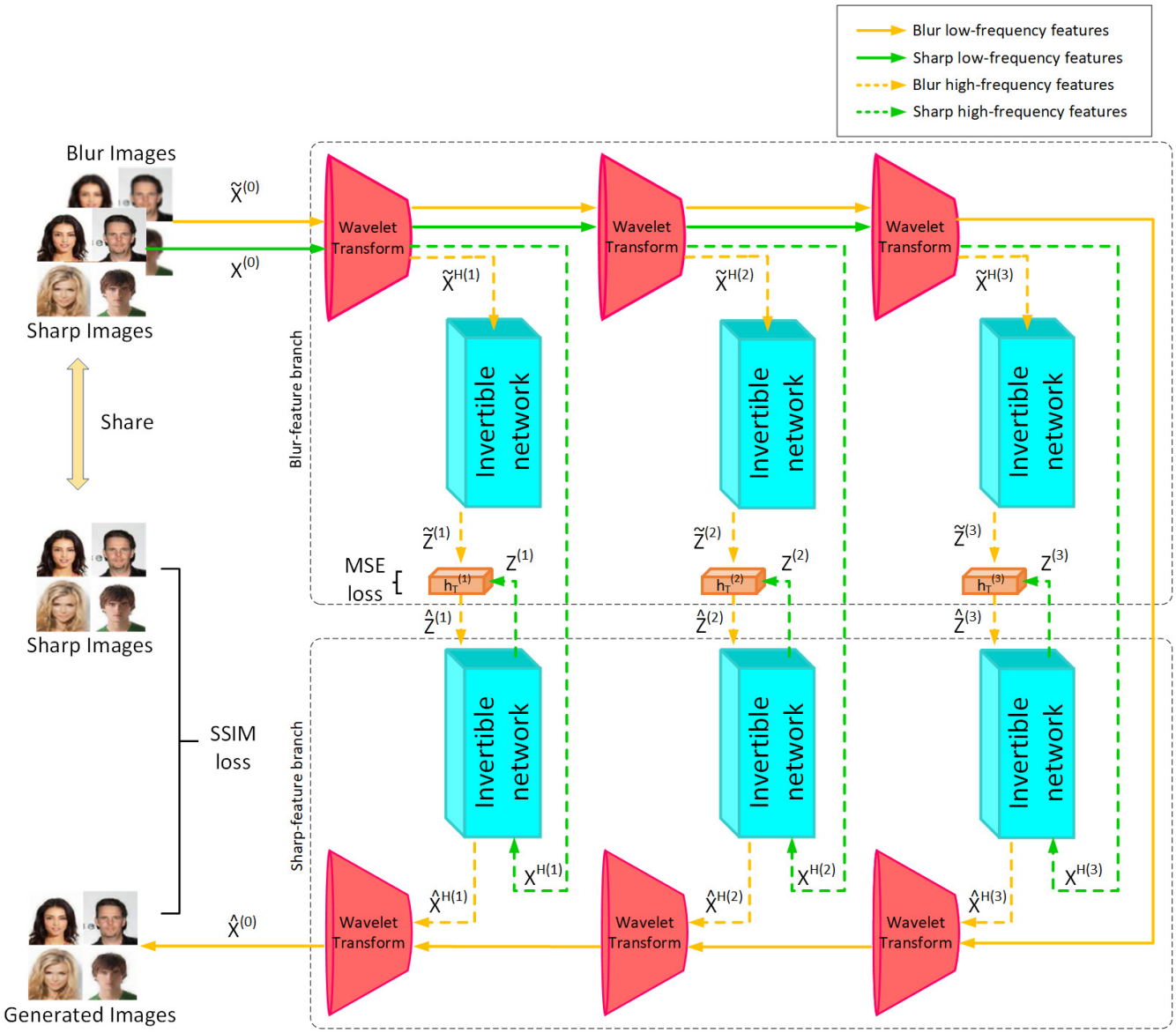


Figure 1. The proposed network architecture for image deblurring. The solid line indicates the low-frequency features and the dashed line indicates the high-frequency features. First, the blur-sharp pairs are decomposed by wavelet transform to obtain the corresponding high-frequency features and low-frequency features, then the latter continues to be decomposed downward, and the former is mapped to latent variables by the invertible networks and the hierarchical optimization loss is calculated in this space. Finally, the sharp-feature branch is further optimized by using the similarity loss.

After finishing the training of all the invertible networks, we only need to input the blurry images to obtain the final deblurring results in an end-to-end manner. The blurry image $\tilde{x}^{(0)}$ is firstly transformed by Equation (5)a and Equation (6)a to the latent variable $\tilde{z}^{(l)}$ on each level l, and then mapped to the estimated sharp variable $\hat{z}^{(l)}$ by Equation (7). Finally, the $\hat{x}^{(0)}$ is obtained by Equation (8) and Equation (9) as the reconstruction procedure.

3.2. Hybrid Losses to Guide Detail Reconstruction

The proposed U-shaped architecture contains the blur-feature branch and the sharp-feature branch, which are used to transform high-frequency features to the corresponding latent variables based on the prior assumption of Gaussian distribution as in [33]. Therefore,

these modules should be first pretrained under the guidance of the maximum-likelihood loss as follows:

$$L_{\mathrm{ML}} = \sum_{l=1}^{L} \left[\frac{\left\| \tilde{f}^{(l)}(\tilde{x}^{H(l)}) \right\|^2 + \left\| f^{(l)}(x^{H(l)}) \right\|^2}{2} - \log \left| \mathrm{J}(\tilde{x}^{H(l)}) \mathrm{J}(x^{H(l)}) \right| \right], \tag{10}$$

where $\mathrm{J}(\tilde{x}^{H(l)}) = \det\left(\partial \tilde{f}^{(l)}(\tilde{x}^{H(l)}) / \partial \tilde{x}^{H(l)}\right)$ and $\mathrm{J}(x^{H(l)}) = \det\left(\partial f^{(l)}(x^{H(l)}) / \partial x^{H(l)}\right)$. These invertible modules transform $\tilde{x}^{H(l)}$ and $x^{H(l)}$ to $\tilde{z}^{(l)}$ and $z^{(l)}$ respectively, which approximates a compact Gaussian distribution in terms of the guidance of Equation (6).

Now that $\tilde{z}^{(l)}$ and $z^{(l)}$ have been aligned to two different distributions of latent variables for the blurry and the sharp features, we need to establish their correlation. The $\tilde{z}^{(l)}$ can be regarded as the degraded high-frequency variable affected by image blurring, which has similarity with the $z^{(l)}$, but cannot be directly applied to the reconstruction of sharp images. Therefore, we utilize the U-shaped architecture to transfer these different latent variables on the same decomposing level, thus avoiding the errors caused by calculating between different spatial levels. In addition, this multi-level architecture plays an important role in vision tasks when realizing the coarse-to-fine reconstruction of spatial details [39–42]. Therefore, the proposed approach optimizes the blur-feature branch by computing the loss at different levels of $\hat{z}^{(l)}$ and $z^{(l)}$ to achieve the full latent variable mapping by Equation (7). The hierarchical optimization loss can be written as follows:

$$L_{\mathrm{MSE}} = \frac{1}{2L} \sum_{l=1}^{L} \frac{1}{C_l W_l H_l} \left\| z^{(l)} - \hat{z}^{(l)} \right\|^2, \tag{11}$$

where H_l, W_l and C_l are the height, width and number of channels for the l^{-th} level decomposing image, respectively. The estimation $\hat{z}^{(l)}$ of sharp high-frequency variable can be iteratively optimized using this loss. $\hat{z}^{(l)}$ can also be directly used in the reconstruction process of the image at the corresponding level. Then, the high-frequency features $\hat{x}^{H(l)}$ obtained by Equation (8) can be used for the reconstruction process at the correct discomposing levels.

Moreover, we perform the calculation of the similarity loss between the $\hat{x}^{(0)}$ and the $x^{(0)}$ to fine-tune the sharp-feature branch. Specifically, the SSIM measurement is applied to evaluate the similarity. Thus, the similarity loss can be denoted by:

$$L_{\mathrm{SSIM}} = h_\sigma(Q_{\mathrm{SSIM}}(x^{(0)}, \hat{x}^{(0)})), \tag{12}$$

where $Q_{\mathrm{SSIM}}(.)$ is to calculate the similarity, and $h_\sigma(.)$ is a Sigmoid function for output smoothing. Considering the small difference in the structural similarity information among the levels, we only calculate the loss on the highest reconstruction level (0) to guide a level-by-level optimization in this paper. After the training, these invertible modules can map $\tilde{x}^{H(l)}$ to $\hat{x}^{H(l)}$, and the $\hat{x}^{H(l)}$ are further utilized to obtain the $\hat{x}^{(0)}$ through the multi-resolution reconstruction.

3.3. Latent Variable Learning by L1 Regularization

To ensure the invertibility, the invertible networks require that the inputs and outputs have the same number of dimensions. However, the high dimensions lead to information dispersion and thus affect the detail generation of the final image. Regularization is a common technique in machine learning, which main purpose is to control model complexity and reduce overfitting. It makes some elements of the weight vector zero or limits the number of non-zero elements, and thus sparsifying the weight vector. Therefore, we attempt to integrate a $L1$ sparse regularization for the latent variable learning as follows:

$$L_{\mathrm{MLR}} = L_{\mathrm{ML}} + \lambda \sum_{l=1}^{L} \left(\left| \tilde{z}^{(l)} \right| + \left| z^{(l)} \right| \right), \tag{13}$$

where $\lambda \in (0,1)$ is the weighting coefficient of the $L1$ regular term, $L_{ML}$is the original likelihood loss.

The strong sparsity of image intensities and gradients has been widely used for low-level computer vision processing problems. It also has mature applications in the field of image deblurring [43–45], such as $L1/L2$ norm [3], reweighted $L1$ norm [46], $L0$ norm prior [47] and sparse prior-Local Maximum Gradient (LMG) [48]. Different from the previous sparse, we consider the dimensions of latent variables as the elements in a weight vector to make the information more focused by sparsifying these variables. Additionally, the mapping of the invertible networks are defined accurately, thus the latent variables can contain all the information in the original images. Note that the $L1$-norm of latent variables is directly used as a regularization term, which will also introduce the adversarial competition. The reason is that the $L1$ regularization will force certain dimensions of latent variables to lose information, hence the adversarial optimization occurs. Such an adversarial learning contributes to a rational distribution of elements in different dimensions, avoiding the extremes of excess or scarcity of information in dimensions and thus further increases the generative details.

4. Results And Discussion

4.1. Experimental Settings

To illuminate deblurring performance, extensive experiments are performed on different scenes including CelebA [49], LSUN [50], GoPro [9] and Lai's dataset [51]. CelebA is a large-scale dataset containing 202,599 faces with 40 attributes. LSUN is another one containing about 1 million images of complex scenes. The GoPro dataset consists of 3214 pairs of blur-sharp images, which is obtained by rapidly recording a sequence of sharp images from a high-speed camera and then averaging these very short intervals of sharp images to obtain blurry images, incorporating complex camera shake and motion. In addition, Lai et al. constructed a blurry dataset containing natural landscapes with complex features by using multiple non-uniform blur kernels.

To build a suitable training set, we used the same data processing method as in [21]. LSUN and CelebA were first randomly sampled to obtain 100,000 images, and then a Gaussian blur kernel (size = 3×3, stride = 1) was used to produce a blurry dataset that could be used for training. For the GoPro and Lai's dataset, we did not perform any processing but used them directly. Finally, we divided the training set, validation set and test set according to the ratio of 8:1:1.

Our model consists of a blur-feature branch and a sharp-feature branch. Inspired by [31], We use three levels of wavelet flow for each branch, and the number of flow steps (see [35] for details) for each wavelet flow is set to 4. The convolution stride used by the network is set to 1, and the kernel size is 5×5. The optimizer follows the Adamx of wavelet flow which has a learning rate that varies with the number of epochs, as detailed in [31].

The number of training epochs per dataset is set to 20,000, and the batch size is set to 32 for CelebA and LSUN and 16 for GoPro and Lai's datasets. The above parameters are set according to experimental experience. The different levels of the images are achieved by wavelet decomposition without the process of downsampling and upsampling. Our model is optimized step-by-step by a likelihood loss, a hierarchical optimization loss and a SSIM loss. The actual training images are first converted the mode from RGB to Ycbcr and then learn only on the Y channel, which is the same as SelfDblur [52]. After training, the images are reconverted to RGB mode for display. The training process is shown in Algorithm A1.

The comparison models in the experiments include DeblurGAN [2], DeblurGAN-v2 [10], SelfDblur [52], ResCGAN [21], Tran et al. [53] method and DED [16]. As in previous work, we use PSNR, MSE and SSIM on different models to measure the performance.

4.2. Comparison of Results on Different Datasets

In this section, we compared the different advanced methods tested in multiple scenarios. Fairness of the evaluation was ensured by comparing the deblurring results on the same images.

4.2.1. Experimental Results on CelebA

The results of the test on CelebA are shown in Figure 2. Each face image is overall blurred. As shown in Figure 2b, the visual perception of DeblurGAN-v2 deblurring is not obvious, and other details such as eyes and the letter on the male background are still disturbed by noise. Self-Deblur recovers spatial details such as facial features and hair texture, but undesired artifacts appear. For instance, the abnormal black traces appear on the upper edge of the female as shown in Figure 2c. This might be due to the fact that the random noise used to generate the images lacked the constraints of the input data coding process and used inputs that are not suitable for deblurring. The visual perception of DED results are better than the previous models, and more details are restored, but there is still some degradation compared with the sharp images. As shown in Figure 2e, our model generate more details such as facial features, hair color and texture without causing artifacts. Although our model computes losses in the latent space to optimize the latent variables, the generated images become sharp. This is because the invertible network's codecs share features and the generated images are optimized at the same time when the latent variables are optimized.

The quantitative results in Table 1 shows that Ours achieved the highest scores in the evaluation of the metrics of PSNR, SSIM and MSE. The PSNR of the proposed model is 2.19 higher than that of the second-ranked Tran et al. method, which has a significant improvement.

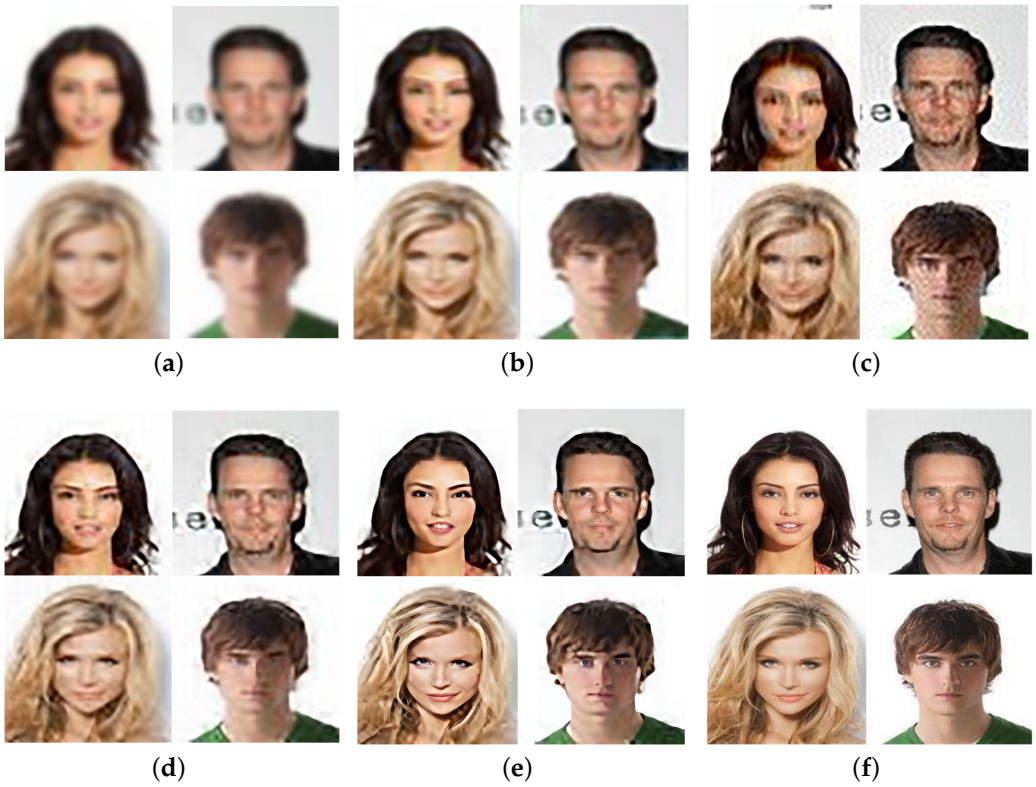


Figure 2. CelebA dataset test results. (**a**) Blur images; (**b**) DeblurGAN-v2; (**c**) Self-Deblur; (**d**) DED; (**e**) Ours; (**f**) Sharp images.

Table 1. Quantitative comparisons evaluated on different datasets with different methods. The best measurements for each of the datasets are in bold.

Method	Measurement	GoPro	Lai	CelebA	LSUN	Ground Truth
DeblurGAN [2]	PSNR	22.98	20.36	19.32	22.29	∞
	SSIM	0.824	0.809	0.801	0.817	1
	MSE	0.005	0.012	0.017	0.004	0
DeblurGAN-V2 [10]	PSNR	23.55	19.59	21.33	22.49	∞
	SSIM	0.835	0.791	0.828	0.832	1
	MSE	0.005	0.011	0.009	0.005	0
ResCGAN [21]	PSNR	25.38	23.42	22.29	24.12	∞
	SSIM	0.898	0.887	0.839	0.915	1
	MSE	0.003	0.004	0.005	0.008	0
Selef-Deblur [52]	PSNR	23.63	22.05	21.85	22.92	∞
	SSIM	0.841	0.811	0.821	0.835	1
	MSE	0.004	0.004	0.004	0.005	0
Tran et al. [53]	PSNR	31.29	28.81	29.23	30.04	∞
	SSIM	0.948	0.929	0.940	0.943	1
	MSE	**0.002**	0.003	**0.001**	**0.001**	0
DED [16]	PSNR	30.98	28.64	27.15	28.72	∞
	SSIM	0.941	0.925	0.937	0.959	1
	MSE	**0.002**	0.003	0.002	**0.001**	0
Ours	PSNR	**32.49**	**29.35**	**31.42**	**32.73**	∞
	SSIM	**0.954**	**0.938**	**0.943**	**0.966**	1
	MSE	**0.002**	**0.002**	**0.001**	**0.001**	0

4.2.2. Experimental Results on LSUN

The test results on LSUN are shown in Figure 3. Each image is achieved with an overall blurring process. The results of DeblurGAN-v2 deblurring, similar to those of CelebA as shown in Figure 3b,g,l, exhibits no significant improvement. The recovery results of Self-Deblur are unstable. Although the building in Figure 3c recovers more spatial details, noise perturbations and poor visual perception enhancement still exist in Figure 3h,m. The performance of DeblurGAN-v2 and Self-Deblur with non-coding structure on building images that have more features is similar to that of CelabA, where the increase in details exerts no large impact on these models. It is because the random noise is weakly correlated with the detail features and the performance of the model is limited by the suitability of the randomly obtained noise for deblurring. By comparison, the proposed model restores a large number of spatial details. For example, the details of the building are restored as shown in Figure 3d,n, and the texture of the building itself is well recovered. In Figure 3i, the details of the leaves are reconstructed, and the house columns were easy to distinguish. The quantitative results in Table 1 show that ours achieved the highest scores in the evaluation metrics of PSNR, SSIM and MSE, outperforming all the other models, and the PSNR is 2.69 higher than the second-ranked Tran et al. method.

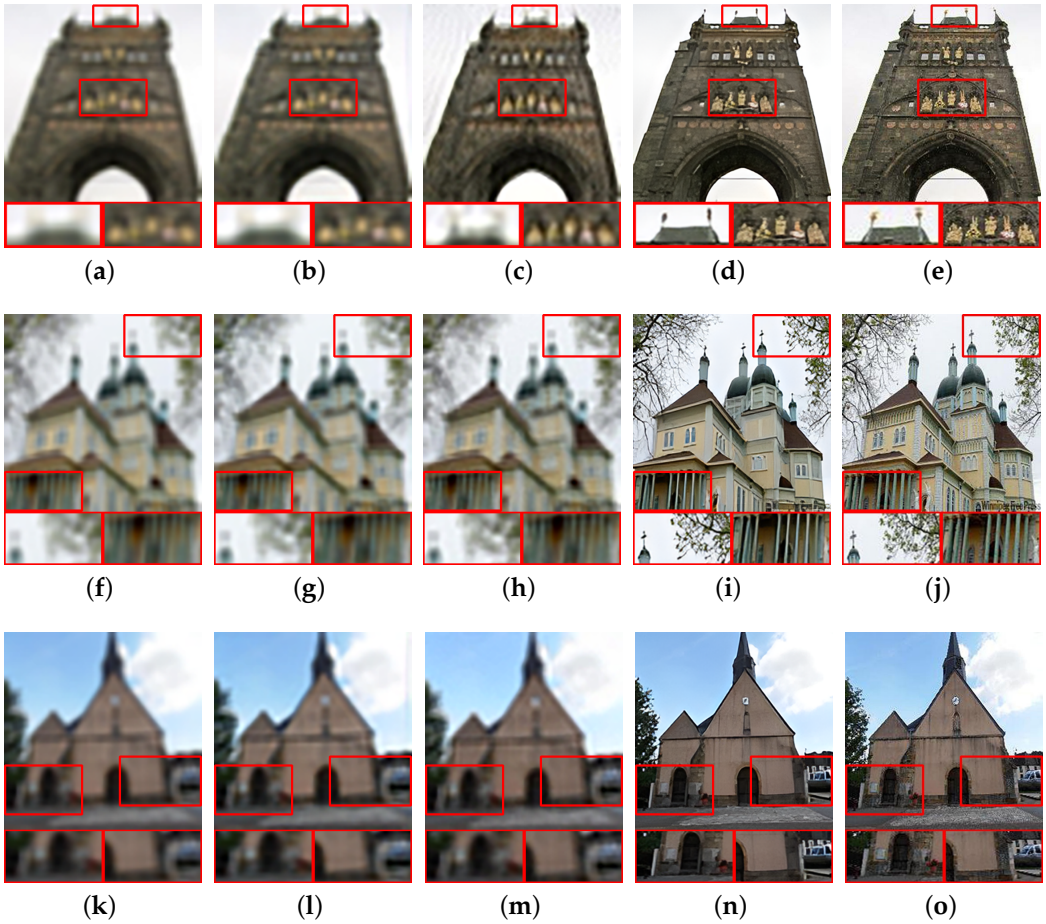


Figure 3. LSUN dataset test results. (**a**) Blur image; (**b**) DeblurGAN-v2; (**c**) Self-Deblur; (**d**) Ours; (**e**) Sharp image; (**f**) Blur image; (**g**) DeblurGAN-v2; (**h**) Self-Deblur; (**i**) Ours; (**j**) Sharp image; (**k**) Blur image; (**l**) DeblurGAN-v2; (**m**) Self-Deblur; (**n**) Ours; (**o**) Sharp image.

We also investigate the effect of using the $L1$ regular term as shown in Figure 4. Since our model is only trained on the Y channel (single channel), the image is wavelet decomposed to obtain three single-channel high-frequency features horizontally, vertically, and diagonally. We show the results by stitching the high-frequency features directly on the channels. As shown in Figure 4a, the image can be divided into the building and the background. Since the high-frequency features are initialized by zero, the brightness of a single-channel image element point is closely related to the amount of information obtained for that element point. There are many bright element points on the background when there is no $L1$ regular term, and the brightness of element points in the same color area of the building itself is cluttered. This is due to the information being randomly distributed across dimensions, with some dimensions featured by information excess or scarcity. The deblurring results with rough details of the building are shown in Figure 4c.

Compared with Figure 4a, the bright element points on the background are significantly reduced after using the $L1$ regular term. The brightness of the element points in the same color area of the building itself changes smoothly, and the details of the building are highlighted in Figure 4b. The deblurring results are shown in Figure 4d, where the spatial details of the building increase and the distinction between the building and the background is more obvious.

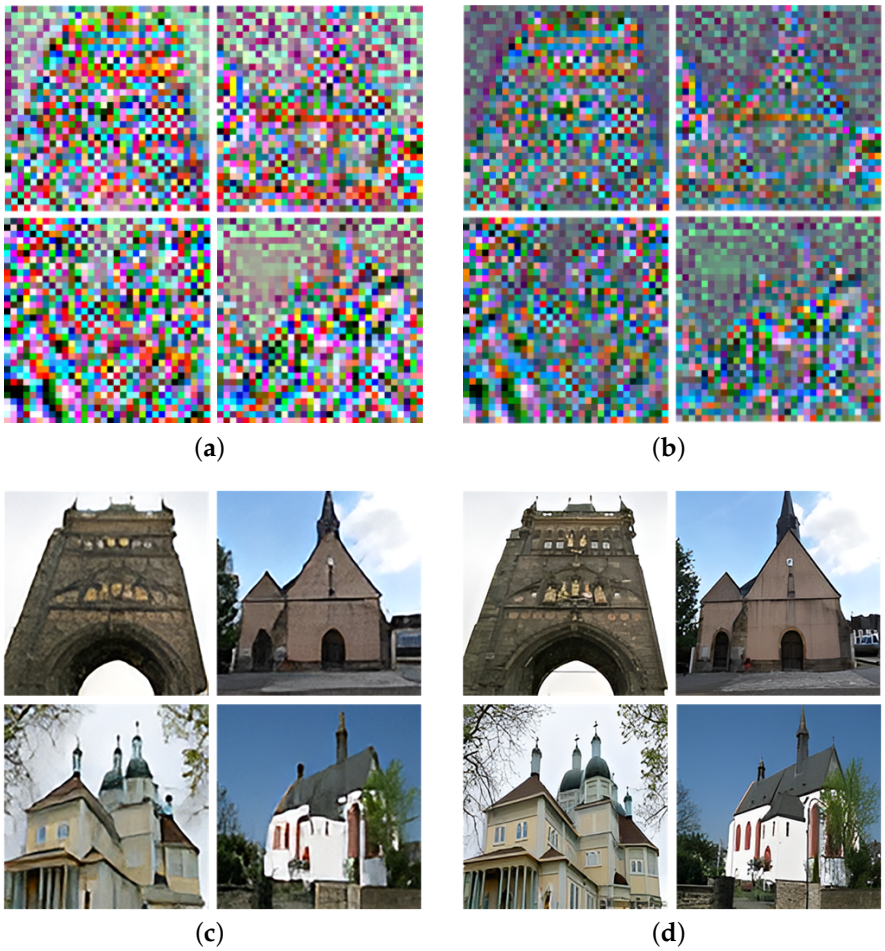


Figure 4. Results of LSUN using $L1$ regular term. (**a**) High-frequency features without $L1$ regular term; (**b**) High-frequency features with $L1$ regular term; (**c**) Deblur images without $L1$ regular term; (**d**) Deblur images with $L1$ regular term.

4.2.3. Experimental Results on GoPro

We also test on a GoPro dataset with complex camera shake and target movement. As shown in Figure 5b,c, the GAN-based deblurring models recover most of the high-frequency features. The building artifacts disappear, and the spatial details of the plants increase so that the overall visual perception becomes significantly better. However, the recovery of some areas contains undesired noise. For example, the windows are distorted after DeblurGAN deblurring, and the branches of the plants are still blurry as displayed in Figure 5b. DeblurGAN-v2 recovers more details and the overall visual perception is better as shown in Figure 5c. The visual perception of the Self-Deblur recovery is further improved despite the slight artifacts in the windows as shown in Figure 5d. In general, the networks with a non-coding structure perform better on this dataset compared to that on CelebA and LSUN. The possible reason for this is that careful tuning of the parameters weakens the effect of noise randomness on the model performance. Our proposed model recovers most of the spatial details as shown in Figure 5g,h. It achieves well-defined plant branches and leaves, increases shrub details, and has significantly better visual perception despite slight artifacts in the windows. Please note that our model is trained well without human intervention. This is because the invertible network's codecs are unified and the model forms mutual constraints spontaneously on the latent variables and the generated images during the learning process. In addition, we use the same settings as CelebA and LSUN except that the batch size is reduced. This again demonstrates that the proposed model can learn sharp features directly, and it is also applicable to motion blur. The quality

evaluation in Table 1 shows that ours has the best overall score among all the methods evaluating PSNR, SSIM, and MSE. It is important to note that the proposed model can be trained efficiently and stably in both forward and backward directions, thereby accelerating the obtaining of the desired results in the actual training process.

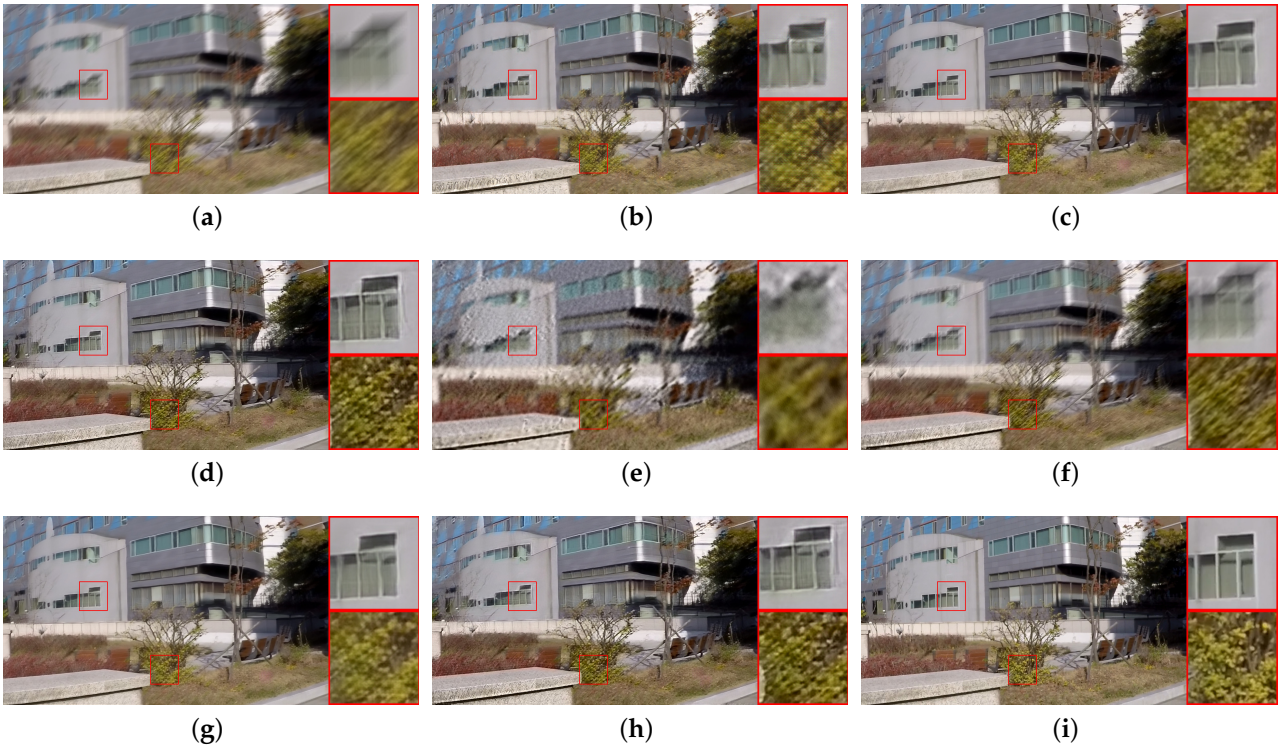


Figure 5. GoPro dataset test results. (**a**) Blur image; (**b**) DeblurGAN; (**c**) DeblurGAN-v2; (**d**) Self-Deblur; (**e**) UML-IN; (**f**) UML-IN + SSIM; (**g**) UML-IN + *L*1; (**h**) UML-IN + *L*1 + SSIM; (**i**) Ground Truth.

In addition, we also compare the convergence of the different models on the GoPro dataset, as shown in Figure 6. Two subplots are used to show the convergence process of each model more clearly.

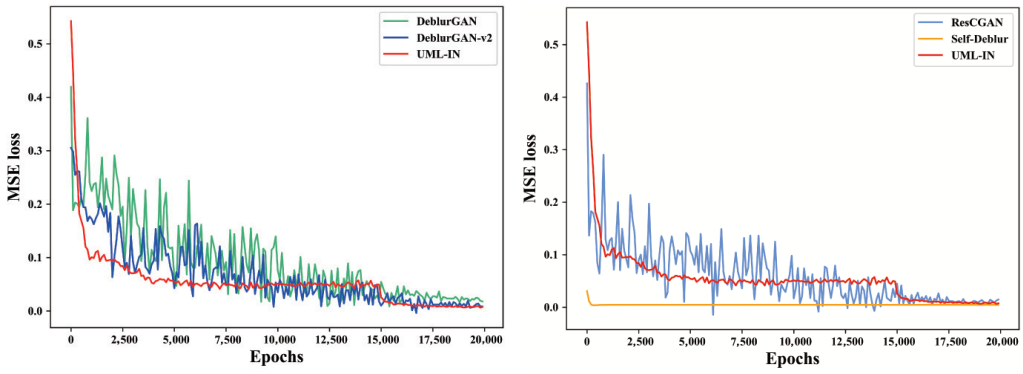


Figure 6. Generate convergence curves of each model on the GoPro dataset.

Convergence curves are plotted based on the MSE calculated from the generated results of the model (i.e., the result of deblurring) with the real sharp images. For the GAN-based deblurring models, we adjust the parameters several times, and obtain the optimal solution for the model convergence, as displayed in Figure 6. We can see that the model without an invertible structure is not stable. The subfigures on the left of Figure 6 illustrate that DeblurGAN and DeblurGAN-v2 oscillated violently in the interval of 0

to 15,000 epochs, and even if they converge, there are still slight fluctuations. Similarly, as shown in the right subfigures, ResCGAN is also difficult to converge. This reflects that the input noise of these models lacks the common constraints of the encoder and decoder, resulting in extremely unstable generated results. Self-Deblur experiences severe overfitting during training. This is because Self-Deblur is trained specifically for a single image and its corresponding kernel, and the resulting blur kernel cannot be used by other images. The convergence curve of UML-IN drops sharply and converges from 0 to 1000 epochs. Due to the addition of hierarchical optimization loss, the curve drops again at 15,000 epochs and finally converges. Compared to the non-invertible networks, the proposed model is significantly more stable and converges faster. This is due to the invertible networks taking full advantage of the codec correlation.

4.2.4. Experimental Results on Lai

To explore the upper limit of the deblurring ability of the proposed model, we also conducted motion deblurring tests on the Lai's dataset for more complex natural landscapes. As shown in Figure 7b,c, the GAN-based deblurring model recovers a large number of spatial details after carefully tuning the parameters. Self-Deblur with the appropriate blur kernel size set also recovers the image well as shown in Figure 7d. Since most of the models can recover the details of waves, rocks, and the left rock wall well, we focus on the deblurring effect of the right rock wall with a complex texture. As shown in Figure 7b–d, DeblurGAN-v2 retains more details, but the edges are recovered slightly coarser. ResCGAN recovers the edges better, but appears over-smoothing and lost some details (e.g., wave ripples). The details recovered by Self-Deblur are still disturbed by noises and the overall brightness of the image becomes lower.

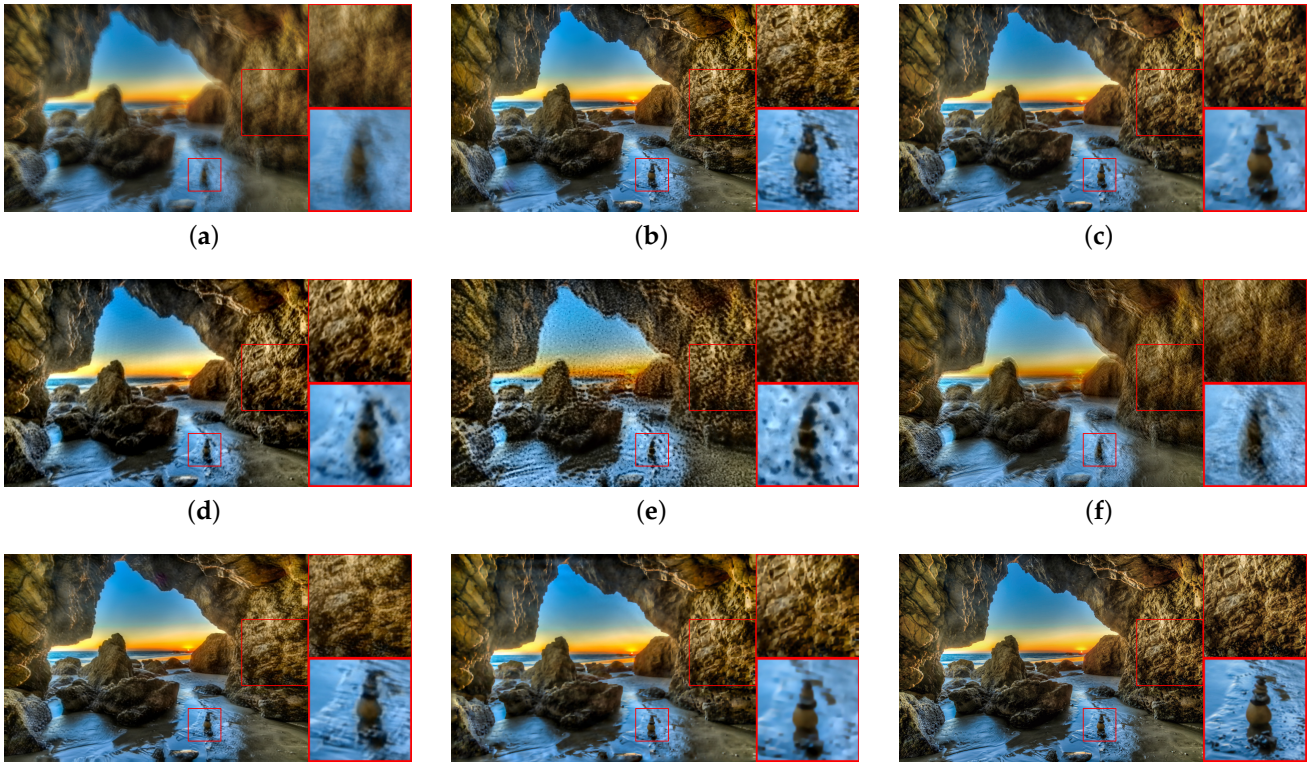


Figure 7. Lai dataset test results. (**a**) Blur image; (**b**) DeblurGAN-v2; (**c**) ResCGAN; (**d**) Self-Deblur; (**e**) UML-IN; (**f**) UML-IN + SSIM; (**g**) UML-IN + *L*1; (**h**) UML-IN + *L*1 + SSIM; (**i**) Ground Truth.

The deblurring results of the proposed model are shown in Figure 7g,h. Compared with UML-IN and UML-IN + SSIM, the model with $L1$ regular term eliminates the undesired artifacts of rocks edges and waves, increasing the spatial details. However, the recovered results regarding the complex textured rock walls on the right still suffer from noise as shown in Figure 7g. In view of this, we combine SSIM with $L1$ to add more high-frequency features to the recovery, and obtain relatively better visual perception results as in Figure 7h, and the proposed model still scores highest on all the evaluation metrics in Table 1. In general, the performance of the proposed model on this dataset is slightly lower than the other, the possible reason is that the complex features make it more difficult to distinguish sharp high-frequency features from artifacts in the case of camera shake. According to the above experiments, our model still needs to be improved in recovering complex details, which is the focus of our next research.

4.3. Ablation Study

4.3.1. Effectiveness of $L1$ Regular Term

The $L1$ regular term is usually used to sparse the weight matrix to avoid model overfitting. This paper uses the characteristics of the $L1$ regular term sparsification to sparse latent variables. The invertible networks have no information loss, while the $L1$ regular term forces the information of some dimensions of the latent variables to converge to zero, hence the adversarial process occurs. This adversarial optimization eventually leads to the redistribution of information in fewer dimensions, which improves the utilization of information and enhances the ability of the model to deblur.

Table 2 shows the advantages of the $L1$ regular term in a quantitative manner. In the evaluation of CelebA, the network with the $L1$ regular term improves the PSNR by 41% and the SSIM by 12% compared with the network without it. In the evaluation of LSUN, the network with the $L1$ regular term improves 38% in PSNR and 14% in SSIM over the network without it. In the evaluation of GoPro, the network with the $L1$ regular term improves 59% in PSNR and 45% in SSIM over the network without it. In the evaluation of Lai, the network with the $L1$ regular term improves 60% in PSNR and 33% in SSIM over the network without it. Therefore the improvement of the model performance due to the $L1$ regular term is significant.

Table 2. Quantitative comparisons evaluated on different datasets with different methods. The best measurements for each of the datasets are in bold.

Method	Measurement	GoPro	Lai	CelebA	LSUN	Ground Truth
UML-IN	PSNR	18.98	17.69	20.63	22.53	∞
	SSIM	0.642	0.707	0.809	0.826	1
	MSE	0.018	0.021	0.010	0.008	0
UML-IN + SSIM	PSNR	22.16	19.67	24.83	24.98	∞
	SSIM	0.818	0.781	0.876	0.894	1
	MSE	0.008	0.017	0.005	0.003	0
UML-IN + $L1$	PSNR	30.26	28.32	29.12	31.10	∞
	SSIM	0.931	**0.944**	0.913	0.941	1
	MSE	0.002	0.005	0.002	0.002	0
UML-IN + $L1$ + SSIM	PSNR	**32.49**	**29.35**	**31.42**	**32.73**	∞
	SSIM	**0.954**	0.938	**0.943**	**0.966**	1
	MSE	**0.002**	**0.002**	**0.001**	**0.001**	0

4.3.2. Effectiveness of SSIM Loss

Unlike MSE, which calculates the differences based on pixel points, SSIM measures the similarity between images in terms of luminance, contrast and structure, and the obtained results are more in line with the intuitive perception of human eyes. In this paper, we use SSIM loss to enhance the perceptual quality and structural details.

The quantitative results in Table 2 show that in the evaluation of CelebA, the network with SSIM loss improves by 20% in PSNR and 8% in SSIM compared to the original network. In the evaluation of LSUN, the network with SSIM loss improves by 10% and 8% in PSNR and SSIM, respectively. In GoPro's evaluation, the network with SSIM loss improves by 16% and 27% in PSNR and SSIM, respectively. In Lai's evaluation, the network with SSIM loss improves PSNR by 11% and SSIM by 10% compared to the original network. Thus, SSIM loss is more beneficial for the performance improvement of deblurring.

Based on the above ablation experimental results, the $L1$ regular term and SSIM loss can help the model improve its performance in various deblurring scenarios.

4.4. Empirical Analysis

In this section, we further analyze the experimental results to reveal the reasons for the dominance of UML-IN and the role of the $L1$ regular term.

4.4.1. Models Analysis

In this paper, we mainly compare with DeblurGAN, DeblurGAN-v2, ResCGAN and self-deblur, Tran et al. and DED for a total of six models. Among them, DeblurGAN adopts the design idea of the classical GAN, i.e., it contains two modules, a generator G and a discriminator D, which yield an image map $G : z \mapsto x$ and the binary map $D : x \mapsto b$ to distinguish whether the input image is real $b = 1$ or fake as $b = 0$. DeblurGAN-v2 introduces the feature pyramid networks (FPN) based on DeblurGAN as the the core module of the generator, resulting in a certain improvement in model performance as shown in Table 1 and a relatively more stable model training process as shown in Figure 6. ResCGAN introduces dense residual paths in the generator to obtain more abundant feature representation and a significant improvement in model performance as shown in Table 1. The above methods all improve the performance of the model by continuously enhancing the ability of the generator, which is a great improvement compared to the traditional method of manually designing a priori. However, in the actual training process, we make a lot of efforts to adjust the strength of the discriminator and generator in each scene to obtain better generation results. This is because GAN is a decoder-only non-invertible network, and the generation results are closely related to random noise lacking coding constraints. Since the random noise enjoys too many degrees of freedom, the stability of the model is weakened and thus needs to be compensated by a large number of tuning parameters.

The methods of Self-Deblur and Tran et al. avoid using random noise directly to generate images, and instead achieve image deblurring indirectly by learning a blur kernel. Although the input of this method is still random noise, the risk of lack of constraints is partly shared by the estimation of the blur kernel, so the model's performance is significantly improved. Note that Self-deblur does not perform better due to the more advanced design ideas as shown in Table 1. This is because the important parameters of Self-deblur still require careful manual tuning, such as the blur kernel size, and thus it is difficult to highlight its advantages. In contrast, the method of Tran et al. develops the estimation of blur kernel size as a learnable process as well, and thus achieves more desirable results. However, while these methods reduce the risk of lacking constraints on random noise, they introduce the risk of inaccurate estimation of the blur kernel and do not fundamentally solve the problem. DED shows the advantage of introducing coding structure to realize image deblurring and does not need to estimate the blur kernel, and achieves comparable results to Tran et al.'s method as shown in Table 1. Although the score is slightly lower than the former, it also shows that the codec structure is a more advantageous design to a certain extent.

The UML-IN proposed in this paper uses a codec-unified structure, which not only avoids the above risks, but also makes full use of the codec correlation compared to the DED codec separation structure, and the training process is also more efficient. Furthermore, random noise (i.e., latent variables) is strictly constrained by the codec process, allowing us to precisely control image details generation in the latent variable space. Therefore,

UML-IN achieves the best results on each evaluation criterion in multiple scenarios and the training process is stable. In summary, UML-IN is more advanced in terms of design and experimental results compared with other models.

4.4.2. Analysis on Various Datasets

Each model has different performances in various scenarios, and the non-invertible networks generally perform well. However, in the actual training process, we need to tune a lot of parameters for each scene to alleviate the lack of constraints of random noise. Therefore, the model generation quality and training process are not stable. For example, DeblurGAN-v2 has a good performance on GoPro and Lai's datasets with complex features as shown in Figures 5b and 7b. However, the performance on CelebA and LSUN with less complex details is not satisfactory as shown in Figures 2b and 3b. As shown in Figure 6, the convergence curves of DeblurGAN, DeblurGAN-v2, and ResCGAN oscillated violently, which manifest the unconstrained random noise and the lack of codec correlation in the model. Self-Deblur is a model for specialized single image deblurring, so it is more stable than GAN, as shown in the right subfigures of Figure 6. However, the model based on blurry kernel estimation has limitations that inaccurate estimation introduces undesired artifacts such as the anomalous black traces in the deblurring image in Figure 2c. Quantitative evaluation in Table 1 shows that Tran et al. achieves competitive results, and the performance of DED is on par with the former.

A good model should minimize human tuning and encourage itself to learn the parameters on its own. The invertible networks use the same networks for coding and decoding, and each backpropagation is optimized both forward and backward. The shared features of the networks coder and decoder unify the learning process of codec, and take full advantage of the correlation between the coder and decoder. As a result, our model performs stably on all datasets. As shown in Figure 2e, our proposed model learns a large amount of face spatial details and performs stably. It also performs well on buildings with more features shown in Figure 3d,i,n. Figure 5e–h shows the performance on motion blur, and it should be noted that we use the blurry dataset directly without adding any additional information. This reflects the advantage of our proposed model which learns sharp features directly without distinguishing blur types. In addition, the invertible networks use a more efficient maximum likelihood strategy to train the model. The maximum likelihood treats all non-target samples as negative samples to calculate the loss to improve the target sample probability, so the training is more efficient and adequate. This is also one of the reasons for the fast and stable convergence of UML-IN, as shown in Figure 6. Besides, the U-shaped structure of the model make sharp images explicitly guide the learning of blurry images, avoiding information errors caused by level transformation and ensuring the accuracy of the model guidance. Finally, compared with kernel models which learn blur laws influenced by priori information, our kernel-free model can learn sharp features directly, thus avoiding the possible effect of inaccurate kernel estimation on the deblurring results. The quantitative experimental results in Table 1 shows that the model proposed in this paper can achieve the best results in various scenarios and two blur types.

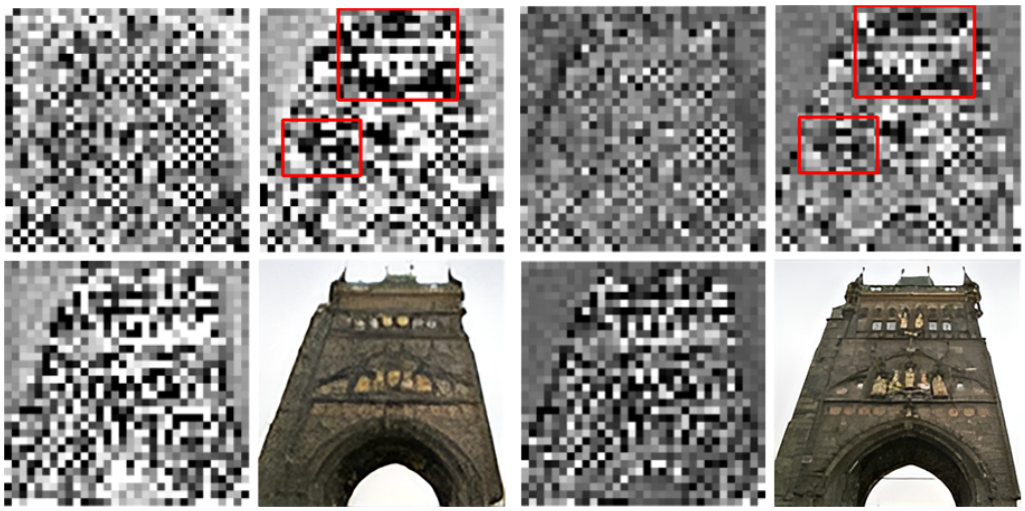
The visual perception of Figures 5e and 7e is slightly poor, probably because the motion-generated artifacts are more pronounced than the other blur types and the ability to distinguish artifacts from sharp features using hierarchical optimization loss is limited. As a result, the model mistakenly identify some artifacts as sharp features. Therefore, we add the $L1$ regular term and SSIM loss to assist the model in distinguishing the features, and the visual perception is significantly improved as shown in Figure 5f–h as well as in the evaluation metrics in Table 2. Another possible reason is that the spatial features are too complex (e.g., complex rock texture in the seascape images in the Lai's dataset) and the extraction capability of the convolutional kernel is limited. Therefore, with the research on convolutional neural networks, the performance of the proposed model may be further improved in the future.

4.4.3. $L1$ Regular Term Analysis

To reveal the role of the $L1$ regular term, we remove this component from the framework. As shown in Figure 5e, the network without the $L1$ regular term is inefficient on the GoPro dataset in recovering spatial features. After using the $L1$ regular term, more information is concentrated in important dimensions to recover a large number of details, as shown in Figure 5e. From Figure 4a,c, the network without the $L1$ regular term is slightly less consistent in recovering high frequency features. In contrast, Figure 4b,d demonstrates that the network with the $L1$ regular term on LSUN recovers more important content, such as details of buildings. In addition, from the results of the ablation experiments in Section 4.3.1, it can be seen that the PSNR of the restored images can be improved by 38% to 60% and the SSIM by 12% to 45% by using the L1 regular term in various scenarios compared to the original model, and the improvement of the model performance is extremely significant.

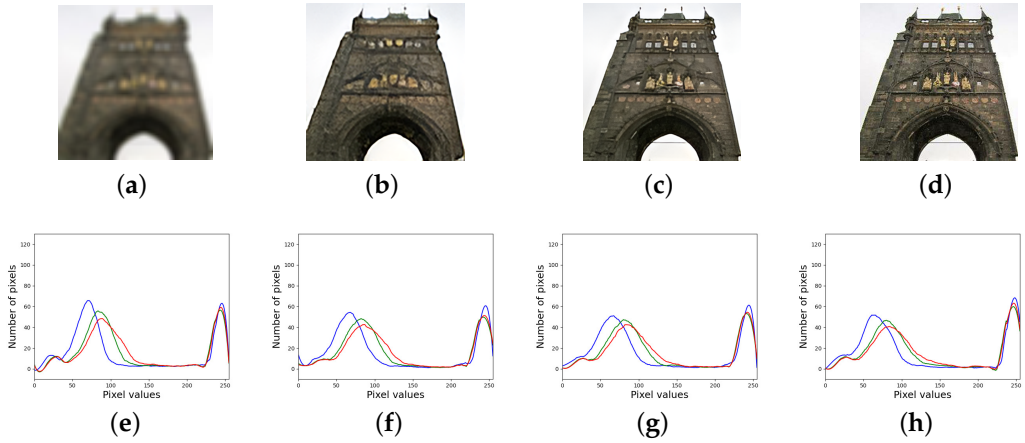
Furthermore, Figure 4a shows the information distribution before regularization is used. Since the high-frequency features are initialized by zero, the brightness of a single-channel image element point is closely related to the amount of information obtained for that element point, with higher brightness indicating more information for that element point and vice versa. Since the adversarial property of the $L1$ regular term reduces the dimensions of information excess and scarcity, the amount of information distribute on the background is reduced and thus concentrate on more important architectural details. So the image recovers more spatial details and the building itself contrasts more clearly with the background as in Figure 4d.

We also display in detail the impact of the $L1$ regular term on the learning of various types of high-frequency features for the proposed model in Figure 8. Each subplot is composed of three single-channel images of high-frequency features (i.e., horizontal, vertical, and diagonal) obtained from wavelet decomposition and their corresponding deblurring images. The vertical high-frequency features (i.e., top right) of Figure 8a can be divided into two parts: the building and the background. When there is no $L1$ regular term in the model, the image background is clearly brightened and the building has a large number of dark element points. This means that there is an excess distribution of information on the monotone background and a lack of information on the building itself with complex details. As a result, the details of the building are missing and its edges are easily affected by the background information, resulting in rough edges of the building. When the $L1$ regular term is used, the vertical high-frequency features in Figure 8b are significantly darkened in the background, and the dark element points of the building are greatly reduced. This is due to the fact that the $L1$ regular term significantly reduces the distribution of the amount of information on the background, thus concentrating the information in the building itself. Hence, the image is recovered more finely, as in the deblurring image of Figure 8b (i.e., bottom right). The reason is that the $L1$ regular term forces some dimensions to be 0, which is contrary to the invertible network without information loss. Thus, information is forced to be redistributed, alleviating to some extent the problem of excess or scarcity of information in dimensions. Figure 9g indicates that the recovered pixel distribution is more similar to the sharp one; for example, the number of pixels with values from approximately 10 to 50 is substantially reduced compared to Figure 9b. In addition, the pixel distribution with pixel values of 50 to 150 is more in line with the distribution of sharp images as shown in Figure 9c.



(**a**) (**b**)

Figure 8. Further results of LSUN using $L1$ regular term. (**a**) LSUN without $L1$ regular term; (**b**) LSUN with $L1$ regular term.



(**a**) (**b**) (**c**) (**d**)

(**e**) (**f**) (**g**) (**h**)

Figure 9. Comparison results on LSUN with their corresponding pixel histograms. (**a**) Blur image; (**b**) Without $L1$; (**c**) With $L1$; (**d**) Sharp image; (**e**) Histogram of (**a**); (**f**) Histogram of (**b**); (**g**) Histogram of (**c**); (**h**) Histogram of (**d**).

5. Conclusions

In this paper, we propose UML-IN, a deblurring framework for learning sparse representations at multi-resolution in the latent space, progressively optimized via maximum likelihood loss, hierarchical optimization loss, and SSIM loss (i.e., hybrid loss). Extensive experiments on real and synthetic blurry datasets show that UML-IN can efficiently and accurately guide the deblurring process and outperform other state-of-the-art models in both qualitative and quantitative evaluations. In addition, the proposed method and its variants are analyzed. The conclusions can be summarized into three aspects. First, in view of the shortcomings of existing methods, we introduce the codec-unified invertible networks into image deblurring to solve the problem of efficiently and accurately analyzing and reconstructing features. Then, the wavelet invertible network is extended to a U-shaped supervised structure for the first time instead of the original single-branch unsupervised scheme, which learns sharp spatial details efficiently and accurately. Finally, the $L1$ regular term is introduced to alleviate the defect of dimensional redundancy in the invertible networks, and compared with the original networks, the PSNR and SSIM can be improved by up to 60% and 45% respectively. Future research might focus on the few defects in this paper, for instance how to improve the fine-grained representation and reconstruction for high-frequency spatial details,

and enhance the accuracy of feature mappings for specific scenes with more intricate details, such as coastal rock ripples etc., under an invertible latent space.

Author Contributions: Conceptualization, M.W., T.W. and H.L.; methodology, M.W. and T.W.; software, T.W.; validation, M.W., T.W. and H.L.; formal analysis, M.W., T.W. and H.L.; investigation, M.W. and T.W.; resources, M.W. and H.L.; data curation, T.W.; writing—original draft preparation, M.W. and T.W.; writing—review and editing, M.W. and T.W.; visualization, T.W.; supervision, M.W., T.W. and H.L.; project administration, M.W. and T.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by National Natural Science Foundation of China (62062048).

Conflicts of Interest: 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.

Appendix A. Implementation of the Proposed Model

In this paper, a step-by-step training strategy is implemeted to optimize the proposed model. First, the level features $\tilde{x}^{H(l)}$ and $x^{H(l)}$ are respectively mapped to the sparse variables $\tilde{z}^{(l)}$ and $z^{(l)}$ according to the blur-feature branch and the sharp-feature branch in terms of the regularization loss L_{MLR}. Then, the loss L_{MSE} is applied to explicitly guide the latent variable alignment of invertible networks. Finally, the loss L_{SSIM} of the sharp image $x^{(0)}$ is calculated to fine-tune the sharp-feature branch.

As can be seen from Algorithms A1 and A2, the latent variables and the generated images of the invertible network are mutually constrained with strong correlation. In addition, the information transferring on the different levels can make the guidance more explicit and the model optimization more refined.

Algorithm A1: Training Procedure.

Input: Blur images: $\tilde{x}^{(0)}$; Sharp images: $x^{(0)}$.
Output: Deblur images:$\hat{x}^{(0)}$.

Initialize $l = 1$.

for $(l \leq L; l++)$ **do**
  $\tilde{x}^{(l)}, \tilde{x}^{H(l)} = h_{\varphi}^{(l)}(\tilde{x}^{(l-1)})$
  $x^{(l)}, x^{H(l)} = h_{\varphi}^{(l)}(x^{(l-1)})$
  while L_{MLR} *is not convergence* **do**
    $\tilde{z}^{(l)} = \tilde{f}^{(l)}(\tilde{x}^{H(l)})$
    $z^{(l)} = f^{(l)}(x^{H(l)})$
  end
  while L_{MSE} *is not convergence* **do**
    $\hat{z}^{(l)} = h_{\mathrm{T}}^{(l)}(\tilde{z}^{(l)})$
  end
end
$\hat{x}^{(L)} = \tilde{x}^{(L)}, l = L$
for $(l \geq 0; l--)$ **do**
  $\hat{x}^{H(l)} = g^{(l)}(\hat{z}^{(l)})$
  $\hat{x}^{(l-1)} = h_{\varphi^{-1}}^{(l)}(\hat{x}^{(l)}, \hat{x}^{H(l)})$
end
$l = L$
while L_{SSIM} *is not convergence* **do**
  for $(l \geq 0; l--)$ **do**
    $\hat{x}^{H(l)} = g^{(l)}(\hat{z}^{(l)})$
  end
end

Algorithm A2: Testing Procedure.

Input: Blur images: $\tilde{x}^{(0)}$.
Output: Deblur images:$\hat{x}^{(0)}$.

Initialize $l = 1$.

for $(l \leq L; l++)$ **do**
$\tilde{x}^{(l)}, \tilde{x}^{H(l)} = h_{\varphi}^{(l)}(\tilde{x}^{(l-1)})$
$\tilde{z}^{(l)} = \tilde{f}^{(l)}(\tilde{x}^{H(l)})$
$\hat{z}^{(l)} = h_{\mathrm{T}}^{(l)}(\tilde{z}^{(l)})$
end
$\hat{x}^{(L)} = \tilde{x}^{(L)}, l = L$
for $(l \geq 0; l--)$ **do**
$\hat{x}^{H(l)} = g^{(l)}(\hat{z}^{(l)})$
$\hat{x}^{(l)} = h_{\varphi^{-1}}^{(l)}(\hat{x}^{(l)}, \hat{x}^{H(l)})$
end

References

1. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **2012**, *25*, 1097–1105. [CrossRef]
2. Kupyn, O.; Budzan, V.; Mykhailych, M.; Mishkin, D.; Matas, J. Deblurgan: Blind motion deblurring using conditional adversarial networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 8183–8192.
3. Krishnan, D.; Tay, T.; Fergus, R. Blind deconvolution using a normalized sparsity measure. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Colorado Springs, CO, USA, 20–25 June 2011; pp. 233–240.
4. Nan, Y.; Quan, Y.; Ji, H. Variational-EM-based deep learning for noise-blind image deblurring. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 3626–3635.
5. Zhang, K.; Luo, W.; Zhong, Y.; Ma, L.; Stenger, B.; Liu, W.; Li, H. Deblurring by realistic blurring. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 2737–2746.
6. Wu, C.; Du, H.; Wu, Q.; Zhang, S. Image Text Deblurring Method Based on Generative Adversarial Network. *Electronics* **2020**, *9*, 220. [CrossRef]
7. Xiang, J.; Ye, P.; Wang, L.; He, M. A novel image-restoration method based on high-order total variation regularization term. *Electronics* **2019**, *8*, 867. [CrossRef]
8. Pan, J.; Bai, H.; Tang, J. Cascaded deep video deblurring using temporal sharpness prior. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 3043–3051.
9. Nah, S.; Hyun, Kim, T.; Mu, Lee, K. Deep multi-scale convolutional neural network for dynamic scene deblurring. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 3883–3891.
10. Kupyn, O.; Martyniuk, T.; Wu, J.; Wang, Z. Deblurgan-v2: Deblurring (orders-of-magnitude) faster and better. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 8878–8887.
11. White, R.L. Image restoration using the damped Richardson-Lucy method. *Instrum. Astron. VIII* **1994**, *2198*, 1342–1348.
12. Hiller, A.D.; Chin, R.T. Iterative Wiener filters for image restoration. In Proceedings of the International Conference on Acoustics, Speech, and Signal Processing, Albuquerque, NM, USA, 3–6 April 1990; pp. 1901–1904.
13. Pan, J.; Sun, D.; Pfister, H.; Yang, M.H. Blind image deblurring using dark channel prior. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 1628–1636.
14. Hongbo, Z.; Liuyan, R.; Lingling, K.; Xujia, Q.; Meiyu, Z. Single image fast deblurring algorithm based on hyper-Laplacian model. *IET Image Process.* **2020**, *13*, 483–490. [CrossRef]
15. Shin, C.J.; Lee, T.B.; Heo, Y.S. Dual Image Deblurring Using Deep Image Prior. *Electronics* **2021**, *10*, 2045. [CrossRef]
16. Wang, Z.; Ren, J.; Zhang, J.; Luo, P. Image Deblurring Aided by Low-Resolution Events. *Electronics* **2022**, *11*, 631. [CrossRef]
17. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B. Generative adversarial nets. *arXiv* **2014**, arXiv:1406.2661v1.
18. Li, C.; Wand, M. Precomputed real-time texture synthesis with markovian generative adversarial networks. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2016; pp. 702–716.
19. Ledig, C.; Theis, L.; Huszár, F.; Caballero, J.; Cunningham, A.; Acosta, A.; Aitken, A.; Tejani, A.; Totz, J.; Wang, Z. et al. Photo-realistic single image super-resolution using a generative adversarial network. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4681–4690.
20. Li, L.; Pan, J.; Lai, W.S.; Gao, C.; Sang, N.; Yang, M.H. Learning a discriminative prior for blind image deblurring. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 6616–6625.

21. Wang, M.; Hou, S.; Li, H.; Li, F. Generative image deblurring based on multi-scaled residual adversary network driven by composed prior-posterior loss. *J. Vis. Commun. Image Represent.* **2019**, *65*, 102648. [CrossRef]
22. Karras, T.; Laine, S.; Aila, T. A style-based generator architecture for generative adversarial networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 4401–4410.
23. Jiang, Z.; Zhang, Y.; Zou, D.; Ren, J.; Lv, J.; Liu, Y. Learning event-based motion deblurring. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 3320–3329.
24. Yuan, Y.; Su, W.; Ma, D. Efficient Dynamic Scene Deblurring Using Spatially Variant Deconvolution Network With Optical Flow Guided Training. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 3552–3561.
25. Suin, M.; Purohit, K.; Rajagopalan, A.N. Spatially-attentive patch-hierarchical network for adaptive motion deblurring. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 3606–3615.
26. Nan, Y.; Ji, H. Deep learning for handling kernel/model uncertainty in image deconvolution. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 2388–2397.
27. Kaufman, A.; Fattal, R. Deblurring using analysis-synthesis networks pair. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 5811–5820.
28. An S.; Roh, H.; Kang, M. Blur Invariant Kernel-Adaptive Network for Single Image Blind Deblurring. In Proceedings of the 2021 IEEE International Conference on Multimedia and Expo, Shenzhen, China, 5–9 July 2021; pp. 1–6.
29. Su, J.; Wu, G. f-VAEs: Improve VAEs with conditional flows. *arXiv* **2018**, arXiv:1809.05861.
30. Ho, J.; Chen, X.; Srinivas, A.; Duan, Y.; Abbeel, P. Flow++: Improving flow-based generative models with variational dequantization and architecture design. *Int. Conf. Mach. Learn.* **2019**, *97*, 2722–2730.
31. Yu, J.J.; Derpanis, K.G.; Brubaker, M.A. Wavelet flow: Fast training of high resolution normalizing flows. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 6184–6196.
32. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the International Conference on Medical image computing and computer-assisted intervention, Boston, MA, USA, 7–12 June 2015; pp. 234–241.
33. Dinh, L.; Krueger, D.; Bengio, Y. Nice: Non-linear independent components estimation. *arXiv* **2014**, arXiv:1410.8516.
34. Dinh, L.; Sohl-Dickstein, J.; Bengio, S. Density estimation using real nvp. *arXiv* **2016**, arXiv:1605.08803.
35. Kingma, D.P.; Dhariwal, P. Glow: Generative flow with invertible 1×1 convolutions. *arXiv* **2018**, arXiv:1807.03039v2.
36. Ardizzone, L.; Lüth, C.; Kruse, J.; Rother, C.; Köthe, U. Guided image generation with conditional invertible neural networks. *arXiv* **2019**, arXiv:1907.02392.
37. Lugmayr, A.; Danelljan, M.; Gool, L.V.; Timofte, R. Srflow: Learning the super-resolution space with normalizing flow. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2020; pp. 715–732.
38. Stephane M. *A Wavelet Tour of Signal Processing*; Elsevier: Amsterdam, The Netherlands, 1999.
39. Denton, E.L.; Chintala, S.; Fergus, R. Deep generative image models using a laplacian pyramid of adversarial networks. *arXiv* **2015**, arXiv:1506.05751v1.
40. Dosovitskiy, A.; Fischer, P.; Ilg, E.; Hausser, P.; Hazirbas, C.; Golkov, V.; Van Der Smagt, P.; Cremers, D.; Brox, T. Flownet: Learning optical flow with convolutional networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 2758–2766.
41. Eigen, D.; Fergus, R. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 2650–2658.
42. Eigen, D.; Puhrsch, C.; Fergus, R. Depth map prediction from a single image using a multi-scale deep network. *arXiv* **2014**, arXiv:1406.2283v1.
43. Xu, L.; Zheng, S.; Jia, J. Unnatural L_0 sparse representation for natural image deblurring. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013; pp. 1107–1114.
44. Levin, A.; Weiss, Y.; Dur, F.; Freeman, W.T. Understanding and evaluating blind deconvolution algorithms. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 1964–1971.
45. Xu, L.; Lu, C.; Xu, Y.; Jia, J. Image smoothing via L_0 gradient minimization. *Acm Trans. Graph. (TOG)* **2011**, *30*, 174. [CrossRef]
46. Yang, D.Y.; Wu, X.J.; Yin, H.F. Blind image deblurring via enhanced sparse prior. *J. Electron. Imaging* **2021**, *30*, 023031. [CrossRef]
47. Pan, J.; Sun, D.; Pfister, H.; Yang, M. Deblurring Images via Dark Channel Prior. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *40*, 2315–2328. [CrossRef] [PubMed]
48. Chen, L.; Fang, F.; Wang, T.; Zhang, G. Blind image deblurring with local maximum gradient prior. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 1742–1750.
49. Liu, Z.; Luo, P.; Wang, X.; Tang, X. Deep learning face attributes in the wild. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3730–3738.
50. Yu, F.; Seff, A.; Zhang, Y.; Song, S.; Funkhouser, T.; Xiao, J. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv* **2015**, arXiv:1506.03365.

51. Lai, W.S.; Huang, J.B.; Hu, Z.; Ahuja, N.; Yang, M.H. A comparative study for single image blind deblurring. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 1701–1709.
52. Ren, D.; Zhang, K.; Wang, Q.; Hu, Q.; Zuo, W. Neural blind deconvolution using deep priors. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 3341–3350.
53. Tran, P.; Tran, A.T.; Phung, Q.; Hoai, M. Explore image deblurring via encoded blur kernel space. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 11956–11965.