J. Vis. Commun. Image R. 79 (2021) 103231

Contents lists available at ScienceDirect

J. Vis. Commun. Image R.

journal homepage: www.elsevier.com/locate/jvci

Full length article

Generative image inpainting with salient prior and relative total variation ☆

Hang Shao *, Yongxiong Wang

School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, Shanghai, 200093, China

ARTICLE INFO

Keywords:
Image inpainting
GAN
Corruption recognition
Salient prior
Relative total variation

ABSTRACT

Image inpainting is an important research direction of image processing. The generative adversarial network (GAN), which can reconstruct new reasonable content in the corrupted region, is the most interesting tool in current inpainting technologies. However, the previous deep methods generally need to be pre-added the binary mask representing the corruption location as the extra input. A novel inpainting algorithm which does not require additional external labels is proposed in this paper. The algorithm consists of two parts: corruption recognition module and content inpainting module, which can recognize and fill random corruption. In the recognizer, the salient object from the uncorrupted region is used as the prior for distinguishing corruption. In the inpainting module, a two-stage network is applied to reconstruct the image from coarse content to texture details. To avoid the misdetection in recognition which has a negative impact on the restoration in inpainting, we perform relative total variational filtering on the corrupted image, and use the salient map as the supervision of detail reconstruction. Qualitative and quantitative experiments on multiple datasets verify the effectiveness of our recognition module, the competitive advantage of our inpainting module, and the enlightening significance of our total algorithm in image inpainting.

1. Introduction

The restoration of corrupted images into complete images with reasonable scenes has been being received widespread attention in industry and academia. In the field of classic image processing, inpainting techniques based on low-level pixels are divided into two major categories: pixel expansion based on partial differential equations [1–4] and patch matching based on exemplar blocks [5–8]. These methods can compensate for minor corruption, but they are not suitable for large-scale corruption. This is because that they cannot capture and reconstruct complex high-level semantics. With the development of deep learning [9,10], researchers have been exploring whether and how to employ convolutional neural networks (CNNs) for inpainting. Pathak et al. [11] proposed an encoder–decoder with discriminative loss, new pixels which do not present in the input can be predicted by this model. Since then, deep inpainting methods have been flourishing [12–24]. These methods ameliorate the image quality of the results continuously by using network architecture upgrading, sampling strategy improvement, objective function optimization, or prior object supervision. However, these methods need to be given special binary masks which indicate corruption in advance, and these masks require manual calibration. Therefore, these methods are inefficient for applications with massive corrupted images.

In order to address this aforementioned problem, a novel inpainting approach is proposed in this paper. Irregular corruption can be recognized and restored by our approach without the need for additional external input labels to the network. The complexity and irregularity of natural corruption make it difficult to be summarized, modeled and learned. Therefore, the idea of our approach is to capture the object from the uncorrupted region and use it as the prior to reversely infer the corrupted region. Our experiments have found that training an appropriate deep salient detector can circumvent non-semantic corruption. Based on the advantages of the salient features [25,26], we use the salient map as the prior and build a model to capture the prior object. There is a certain edge mutation between the corrupted region and the uncorrupted region. In order to take full use of this edge discontinuity, we perform the relative total variation (RTV) [27,28] on the corrupted image. Meanwhile, we incorporate multiple edge supervision layers in the upsampling of the prior capture network. After obtaining the prior, we use a series of calculations to infer corruption. Then, we use a set of two-stage GANs to inpaint the corrupted image progressively. We insert the feature aggregation module (FAM) [29] to improve the traditional inpainting generators to make them more suitable for the irregular region reconstruction tasks. Importantly, to improve the fault tolerance and robustness of our total model, we invoke RTV again. In the inpainting module, the processed object is modified from the

☆ This paper has been recommended for acceptance by Liu Haowei.

* Corresponding author.
E-mail addresses: 932390809@qq.com (H. Shao), wyxiong@usst.edu.cn (Y. Wang).

https://doi.org/10.1016/j.jvcir.2021.103231
Received 6 March 2021; Received in revised form 2 June 2021; Accepted 10 July 2021
Available online 16 July 2021

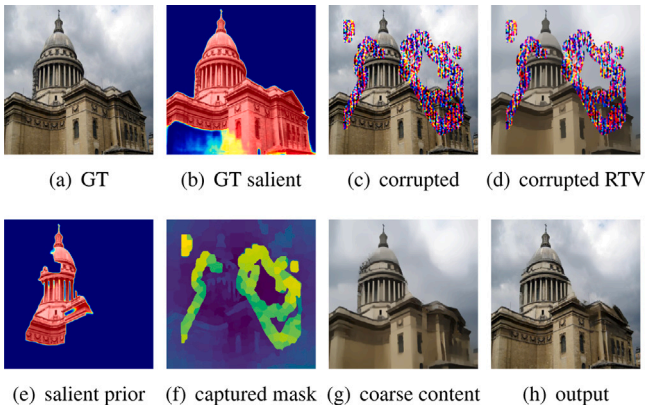


Fig. 1. Illustration of the proposed deep image inpainting algorithm. Given an image (a) with corruption (c), our algorithm extracts the salient object (e) of the corrupted RTV map (d) and uses the map as a prior. Compared with the ground truth salient (b), our detector can avoid the corrupted region. Subsequently, a set of GANs are used for content filling (g) and reconstruction (h) of the corrupted image.

original corrupted image to the RTV map. However, since the filtering characteristics of RTV, details and textures will be lost. We use the salient map as the supervision to guide the second-stage GAN to restore and reconstruct these textures more correctly. The illustration of our algorithm is shown in Fig. 1.

Our method is dedicated to making the deep model better fit for image inpainting tasks. Since there is no large-scale public dataset for random image corruption, we use the binary masks provided by Liu et al. [30] to simulate the irregular corrupted regions, and use the noise maps generated by a Markovian GAN [31] to simulate the random non-binary corrupted contents. We train, validate and test our model on the Places2 [32], CelebA [33], Oxford Building [34], and DUTS [35] datasets. Qualitative and quantitative experiments show that our algorithm can recognize and inpaint image corruption automatically. Comparative experiments demonstrate that the inpainting performance of our algorithm outperforms the state-of-the-art methods.

The main contributions of this paper are as follows:

- A novel deep network is built to solve the problem of blind image inpainting.
- Image corruption which is difficult to be modelled can be recognized and inpainted by our algorithm based on the salient prior without the need for additional labels or manual annotations.
- The fault tolerance and robustness of the total network is improved by the RTV filtering.
- The inpainting performance of our network can be raised compared with the state-of-the-art methods by invoking the FAM module and salient map guidance.

2. Related work

GAN provides a new horizon on the issue of image inpainting, that is, the repaired content can be generated based on learning instead of based on variational equations or patch matching. However, traditional inpainting networks might cause artifacts or blurring. In order to make the model fit the context, Yang et al. [36] embedded the texture supervision mechanism into the generator to improve the result details. Zhang et al. [37] proposed a guided enhanced perception module to make the results more vivid. Since GAN's training can be seen as seeking the Nash equilibrium between the generator and the discriminator, the process needs to be controlled carefully to avoid gradient disappearance or model collapse. Therefore, while improving the generator, Iizuka et al. [38] changed the discriminator to a combination of global and local. Although the strategy of separating the local critic from the global critic can equilibrate the joint training by delaying discrimination, the local critic can only distinguish rectangular blocks. PartialConv [30] and GatedConv [39] are dedicated for inpainting irregular corruption. Random holes can be filled based on the spatial position of their masks. Li et al. [40] designed a circular reasoning module to make the prediction results approach the existing clues gradually. Chen et al. [20] proposed a squeeze-and-excitation network to ensure that the semantic features of the image can be learned fully. Jam et al. [24] improved the utilization of positive and negative masks based on the learnable forward module. In addition, similar networks have been widely used for object removal, image upsampling, zooming, style transfer, and super-resolution reconstruction [15,41–43]. However, when faced with complex scenes, implementing a single-stage GAN is usually inadequate.

Based on the schemes of boosting and progressive, researchers decomposed complex problems by integrating multi-stage architecture. Yu et al. [44] divided the network into a coarse content module and a refinement module. Although the design will increase a certain number of weight parameters, the robustness and generalization of a single network is enhanced. Since then, the coarse-and-fine architectures have been adopted widely [16,17,19]. EdgeConnect [13] further changes the internal labeling path of the coarse-and-fine network. In the first stage, EdgeConnect captures the edge maps of the corrupted images and completes the edges. The second stage network performs color filling on the completed edges. The label format is also adopted by PRVS [45] and EC-GAN [21]. However, since many pixels in the global edge map have no actual values, binary edge labels are more prone to cause generator gradient oscillations than color labels. StructureFlow [12] improves the label of EdgeConnect into a structure diagram, which makes the result more realistic.

However, it is difficult for these above methods to identify image corruption before inpainting. For blind image inpainting, Liu et al. [46] applied a set of two-stage architectures which can learn complementary priors to capture the residuals between the damaged image and the output image. Then, they determined the residual map as the corruption. However, the network is not suitable for identifying large-region corruption. Wang et al. [47] proposed a two-stage visual consistency model (VCNet). The first stage of VCNet predicts the mask of the corrupted image, while the second stage performs inpainting based on the mask. However, VCNet's strategy to improve the robustness of the total network is to supervise the prediction mask into each convolution group of the inpainting network, which is limited to natural mask detection.

3. The proposed approach

3.1. Prior object recognition

The traditional GAN-based blind inpainting algorithms [46,47] use the deep network to learn the corrupted region directly. However, since the wide variety of natural defects, it is difficult to model them effectively. Therefore, we adopt the scheme of finding the prior object in the uncorrupted region to reversely infer the corrupted region. We use an FPN [48] to capture salient objects of the corrupted image. Since the underlying characteristics of salient features, the network can satisfy to a certain extent that the extracted salient objects do not contain the non-semantic corruption. In order to improve the corruption avoidance of the salient network, we establish the edge supervision mechanism in the FPN backbone based on the edge pixel mutation between the corrupted region and the uncorrupted region. The mechanism is to embed 4 residual modules [49,50] in parallel on the 4 skip connections of the FPN's top-down path. The feature map of each level is not only transferred to the subsequent convolutional layer, but also detected by the corresponding residual block to generate the aggregation edge map. In order to further improve the utilization of edge information, we replace the original corrupted image with the corrupted RTV map as

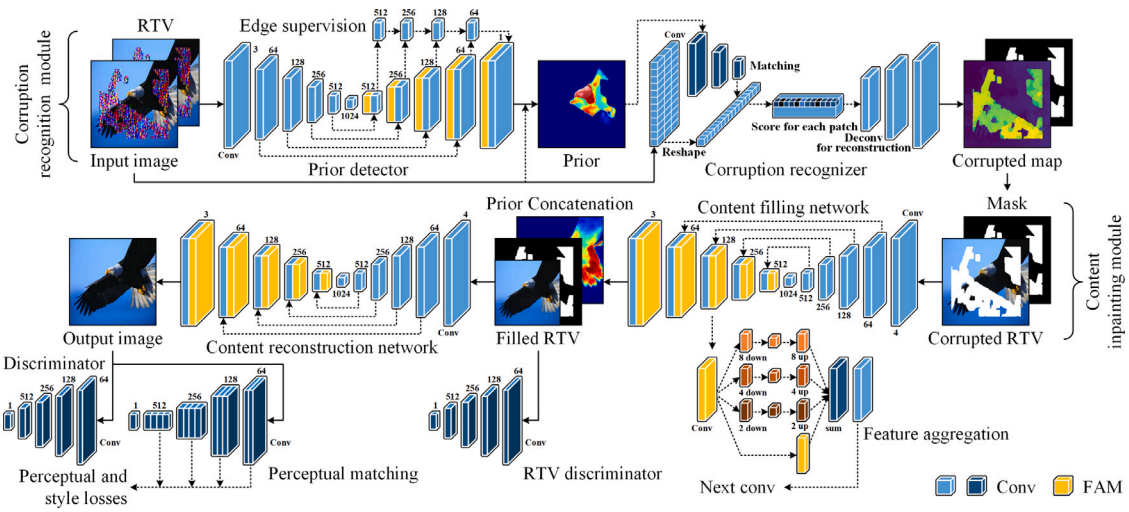


Fig. 2. The total framework of the proposed deep image inpainting algorithm. The corruption recognition module identifies defective regions based on the salient prior, and the content inpainting module reconstructs the defective image from coarse content to texture details.

the input pattern of the salient network. RTV is an improved model of the traditional total variation model [1], which aims to eliminate high-frequency noise in the original image while preserving a clearer edge structure. The traditional TV model performs filtering by minimizing the energy function of the Euclidean distance and regularizing the total variation between the structure map and the image. The RTV model modifies the anisotropy of pixels to form a combination of the Gaussian function and the anisotropy of all pixels in a domain. The module can strip the structure from non-uniform textures properly without specifying whether the textures are regular or symmetrical. After determining the input pattern and the specific architecture of the salient prior capture network, the network can output objects that only from the uncorrupted region of the input corrupted image. In the algorithm execution project, the corrupted image is represented as $I_{corrupt}$, and the ground truth image is represented as I_{gt}. The relationship between I_{gt} and $I_{corrupt}$ is:

$$I_{corrupt} = I_{gt} \odot (1 - M) + C \odot M \tag{1}$$

where C is the corrupted content, M is the binary representation of C (1 is the corruption, 0 is the uncorrupted region), and $\odot$ is the Hadamard product. During the training process, the objective function of the prior detector is:

$$\begin{aligned}\mathcal{L}_{prior} = &-\left[S \log(S_{prior}) + (1 - S) \log(1 - S_{prior})\right] \\ &- \sum_{k=1}^{4} \left[E \log(E_{patch}^{(k)}) + (1 - E) \log(1 - E_{patch}^{(k)})\right]\end{aligned} \tag{2}$$

where S is the ground truth salient of $I_{corrupt}$, E is the edge of the corrupted RTV map $R_{corrupt}$, $E_{patch}^{(k)}$ is the edge obtained from the kth convolutional level, and S_{prior} is the generated salient prior.

3.2. Corruption inference

After making the salient prior object S_{prior} avoid the corrupted region as much as possible, we fuse S_{prior} with the corrupted image $I_{corrupt}$ to obtain a map I_{prior} containing the content details of the original input:

$$I_{prior} = S_{prior} \odot I_{corrupt} \tag{3}$$

Then, we convert the corrupted image into multiple patches (as shown in Fig. 2, the size of the patch is set according to the fine requirements of the task and the input image), and divide these patches into two categories according to whether they come from the corrupted region. After that, we transform I_{prior} to the feature filter, and reshape the patches according to the size of I_{prior}. Then, we cascade each patch with I_{prior} to obtain the corresponding number of 6-channel matrices. According to whether the patches are coming from the corrupted region, we train a residual module [50] to classify these patches. The trained model can calculate the predicted values based on the relationship between the patch and I_{prior}. Finally, we perform K-Means two-clustering operator on these predicted values, and reconstruct the corrupted map C according to the patches.

3.3. Image inpainting module

Since the misdetection and misidentification of corruption will have a significant negative impact on the inpainting module, and the image background may be quite different from the salient objects. We adopt a set of GANs to perform the task progressively to ensure the robustness of our algorithm. In the first-stage, GAN generates the coarse content I_{coarse} based on the corruption C and the corrupted RTV map $R_{corrupt}$. The second-stage GAN reconstructs the global textures and details of the coarse content I_{coarse} based on the salient prior S_{prior}.

The generator of our first-stage content filling module is built upon a U-Net [51]. Liu et al. [30] proved that the structure of U-Net has favorable performance in inpainting irregular regions. Meanwhile, we improve the architecture and introduce FAM to the inpainting task. A series of FAM modules are inserted at the skip connections of the GAN's decoder. In an FAM, a feature level is downsampled by different multiples, followed by average pooling and upsampling of the corresponding multiples. Then, these sampling results are aggregated into a new feature map. In this paper, the sampling magnifications are set to 1, 2, 4, and 8 times of the feature level, respectively. This module can assist the network to extract features from the multi-scale space, reduce the aliasing effect caused by the deconvolution operation, and increase the network receptive field of the generator. After $R_{corrupt}$ is filled by the first-stage generator, the coarse map I_{coarse} can be obtained:

$$I_{coarse} = G_1(R_{corrupt}, C) \tag{4}$$

where G_1 is the processing of the coarse content filling generator.

We apply the discrimination in the 70 × 70 PatchGAN [42] as the discriminator D_1 of the content filling module. For training, the adversarial loss $\mathcal{L}_{adv}^1$ is:

$$\mathcal{L}_{adv}^1 = \mathbb{E}\left[\log\left(1 - D_1(I_{coarse})\right)\right] + \mathbb{E}\left[\log\left(D_1(R_{gt})\right)\right] \tag{5}$$

where R_{gt} is the RTV map of the ground truth uncorrupted image I_{gt}.

Algorithm 1 The proposed inpainting approach

Input: corrupted image $I_{corrupt}$
Output: inpainted result $I_{inpaint}$

$R_{corrupt} \leftarrow \text{RTV}(I_{corrupt})$;
$S_{prior} \leftarrow \text{SalientDetect}(R_{corrupt})$;
$I_{prior} \leftarrow \text{Fusion}(I_{corrupt}, S_{prior})$;
$FeatureFilter \leftarrow \text{Transfer}(I_{prior})$;
$Patch_n \leftarrow \text{Reshape}(I_{corrupt})$;
for $i = 0$, i in range(n), $i++$ **do**
    $\alpha_i \leftarrow \text{Score}(Patch_i, FeatureFilter)$;
end for
for $i = 0$ to n **do**
    $\alpha_i^* = (\alpha_i - \text{Min}(\alpha_n)) \,/\, (\text{Max}(\alpha_n) - \text{Min}(\alpha_n))$;
    $(Cluster_0, Cluster_1) \leftarrow \text{K-Means}(\alpha_0^*, \alpha_1^*, ..., \alpha_n^*)$;
    if α_i^* in $Cluster_0$ **then**
        $\alpha_i^* = 1$;
    else
        $\alpha_i^* = 0$;
    end if
end for
$C \leftarrow \text{Reconstruct}(\alpha_n^*)$;
$I_{coarse} \leftarrow G_1(R_{corrupt}, C)$;
$I_{inpaint} \leftarrow G_2(I_{coarse}, I_{corrupt}, S_{prior}, C)$;
return $I_{inpaint}$

We use the ℓ_2 distance $\mathcal{L}^1_{\ell_2}$ to measure the low-frequency difference between R_{gt} and the generated coarse map I_{coarse}:

$$\mathcal{L}^1_{\ell_2} = \|R_{gt} - I_{coarse}\|_2 \tag{6}$$

We optimize the joint training of the first-stage generator G_1 and its corresponding discriminator D_1 according to the following objective function:

$$\mathcal{L}^1_{inpaint} = \lambda^1_{adv}\mathcal{L}^1_{adv} + \lambda^1_{\ell_2}\mathcal{L}^1_{\ell_2} \tag{7}$$

where λ^1_{adv} and $\lambda^1_{\ell_2}$ are hyperparameters. Based on our experiments, λ^1_{adv} and $\lambda^1_{\ell_2}$ are set to 1 and 10, respectively.

In the second-stage of our inpainting network, we reconstruct the global details and textures of the coarse map I_{coarse}. The generator and discriminator in the second-stage are similar to the content filling network. Moreover, we concatenate the salient prior S_{prior} to the module input and use it as the texture supervision item for content reconstruction. Then, we can obtain the refined completed image $I_{inpaint}$:

$$I_{inpaint} = G_2(I_{coarse}, I_{corrupt}, S_{prior}, C) \tag{8}$$

where G_2 represents the generation process of the content reconstruction module.

We use the adversarial loss $\mathcal{L}^2_{adv}$ and the ℓ_1 distance $\mathcal{L}^2_{\ell_1}$ to measure I_{gt} and $I_{inpaint}$ in training. The adversarial loss $\mathcal{L}^2_{adv}$ is:

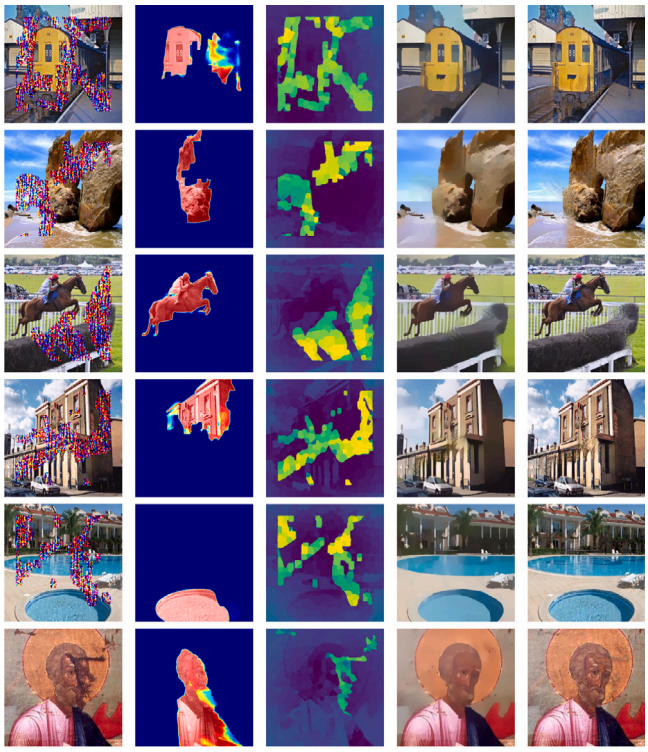
$$\mathcal{L}^2_{adv} = \mathbb{E}\big[\log\big(1 - D_2(I_{inpaint})\big)\big] + \mathbb{E}\big[\log\big(D_2(I_{gt})\big)\big] \tag{9}$$

where D_2 represents the discriminator of the content reconstruction network. The ℓ_1 distance is:

$$\mathcal{L}^2_{\ell_1} = \|I_{gt} - I_{inpaint}\|_1 \tag{10}$$

In addition, we construct a perceptual matching module based on the pre-trained VGG-19 network in parallel with the discriminator. I_{gt} and $I_{inpaint}$ are the features extracted by the perceptual matching module, which are used to calculate the perceptual loss $\mathcal{L}^2_{petl}$ and the style loss $\mathcal{L}^2_{stl}$ [13]. The perceptual loss $\mathcal{L}^2_{petl}$ is:

$$\mathcal{L}^2_{petl} = \mathbb{E}\Big[\sum_i \frac{1}{N_i}\|\Phi_i(I_{gt}) - \Phi_i(I_{inpaint})\|_1\Big] \tag{11}$$



(a) corrupted (b) prior (c) corruption (d) coarse (e) output

Fig. 3. The total model validation of our deep inpainting approach. For the corrupted image (a), our model obtains its salient object (b), then reverses the corruption (c), and then reconstructs the image according to the corruption from coarse (d) to fine (e). The first five rows are synthetic corruption, and the sixth row is natural corruption.

where i is the number of convolutional layers of the perceptual matching module, Φ_i is the activation in the ith layer, and N_i is the number of elements in the ith layer.

The style loss $\mathcal{L}^2_{stl}$ is expressed as:

$$\mathcal{L}^2_{stl} = \mathbb{E}\Big[\|\Psi^{\Phi}_j(I_{gt}) - \Psi^{\Phi}_j(I_{inpaint})\|_1\Big] \tag{12}$$

where Ψ^{Φ}_j is the Gram matrix of $P_j \times P_j$ constructed according to the activation of Φ_j (the feature map size is $H_j \times W_j \times P_j$). A structure similar to the perceptual matching module is proved by Johnson et al. [41], which can eliminate the checkerboard artifacts caused by the transposition of the convolutional layers.

The total optimization function of the second-stage content reconstruction network is:

$$\mathcal{L}^2_{inpaint} = \lambda^2_{adv}\mathcal{L}^2_{adv} + \lambda^2_{\ell_1}\mathcal{L}^2_{\ell_1} + \lambda^2_{petl}\mathcal{L}^2_{petl} + \lambda^2_{stl}\mathcal{L}^2_{stl} \tag{13}$$

where λ^2_{adv}, $\lambda^2_{\ell_1}$, λ^2_{petl} and λ^2_{stl} are hyperparameters. Based on our experiments, we set $\lambda^2_{adv} = 1$, $\lambda^2_{\ell_1} = 50$, $\lambda^2_{petl} = 2$ and $\lambda^2_{stl} = 200$, respectively.

The framework of our approach is shown in Fig. 2, and the algorithm pipeline is described in Algorithm 1.

4. Experiment

4.1. Implementation details

In order to verify the effectiveness of our network, we train, validate and test it on the Places2 [32], CelebA [33], Oxford Building [34], and DUTS [35] datasets. Places2 contains 8,000,000 images which are divided into 365 major categories based on their approximate probability of appearing in nature. CelebA contains 202,599 character

Table 1
The comparison results obtained by using RAS, U2Net, PoolNet, and our model to detect salient objects as corresponding priors for corruption recognition. ⋆ higher is better, † lower is better.

Results of similarity discrimination			
	SSIM⋆	MAE†	MSE†
RAS [52]	0.8217	0.3904	0.3940
U2Net [26]	0.8643	0.4541	**0.3760**
PoolNet [29]	0.8719	0.4279	0.3978
Ours	**0.9172**	**0.3230**	0.3825
Results of binarization			
	SSIM⋆	MAE†	MSE†
RAS	0.9681	0.0734	0.0496
U2Net	0.8937	0.1281	0.0513
PoolNet	0.9252	0.1526	0.0488
Ours	**0.9699**	**0.0710**	**0.0355**

images, which are used to provide verification in portrait scenes. The resolutions of Oxford Building are up to 1024 × 1024, which can verify the effect of our algorithm at high-resolution. DUTS is the largest public dataset in the current salient detection tasks, which has 10,533 images for training (DUTS-TR) and 5019 images for testing (DUTS-TE). Each image in DUTS has a corresponding ground truth salient map, which can be used to evaluate our salient prior module. Moreover, the images in DUTS-TR provide the ground truth edge contour maps in addition to the salient maps, which can enable us to further train the edge utilization mechanism in the salient detection module. We use the masks provided by Liu et al. [30] to simulate the image corrupted regions. The mask dataset contains 12,000 irregular binary mask images, which are classified according to the occupied image regions from 1% to 60%. Since the corruption faced by natural images is very random, there is no dedicated dataset. Simulating the corrupted content as realistically as possible can make the trained model has better generalization performance. We employ a Markovian GAN [31] to generate random noise maps to represent randomly corrupted contents.

In the construction phase of our network, the corruption recognition module and the two-stage inpainting module are trained separately. The input size is set to 256 × 256, the batch size is set to 12, the optimizer is Adam [53], and the initial learning rate is set to 10^{-4}. The experiments are conducted on a workstation with an Intel Xeon Silver 4116 CPU and a TITAN XP GPU. In particular, we apply spectral normalization [54] in the inpainting module to suppress sudden changes in parameters and gradients. In the evaluation, the corrupted image can be inpainted end-to-end without additional external labels.

4.2. Validation of the total algorithm

The qualitative demonstrations of our total model are shown in Fig. 3. It can be seen that our model can avoid the corrupted region appropriately and capture the salient prior automatically. Then, our model uses the salient prior to identify corruption reversely, and completes image filling and reconstruction gradually. In Section 4.3, we compare our prior detection module with the reverse attention network (RAS) [52], U2Net [26], and PoolNet [29]. We use these 4 different methods to capture salient objects in the corrupted images and obtain different priors for identifying corrupted regions. The qualitative and quantitative results show that our algorithm has excellent performance. In Section 4.4, we compare our inpainting module with multiple advanced inpainting methods, including the PatchMatch (PM) [5], global & local consistency network (GLN) [38], contextual attention network (GCAN) [44], PartialConv [30], GatedConv [39], EdgeConnect [13], progressive reconstruction network (PRVS) [45], StructureFlow [12], and recurrent feature reasoning network (RFR) [40]. The comparison results show that the inpainting effect of our inpainting module is better than the state-of-the-art methods. In Section 4.5, we establish a series of ablation studies to further show the robustness of our network and the scientificity of the module built in this paper.

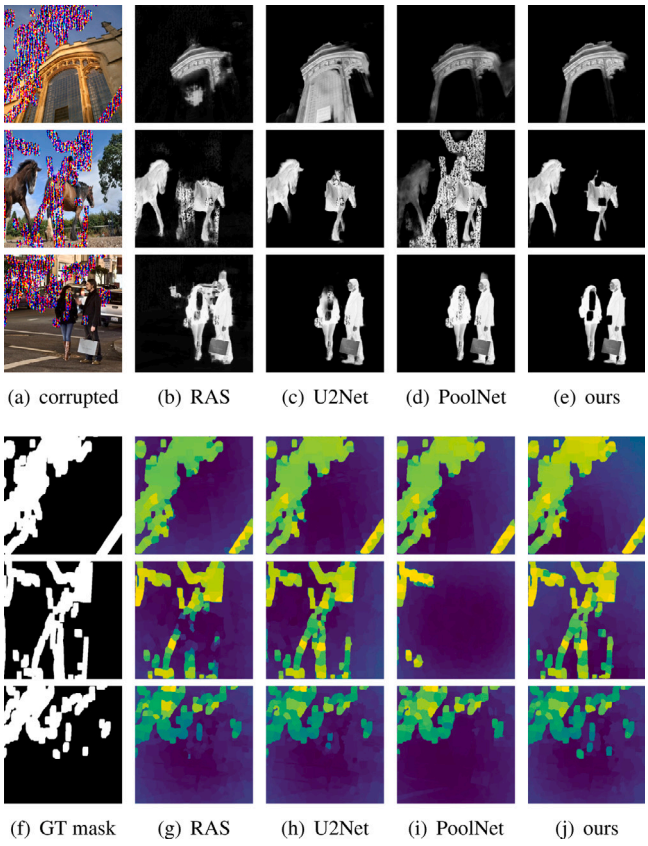


Fig. 4. Salient detection and corruption recognition of the corrupted images. Given corrupted images (a), we use RAS (b), U2Net (c), PoolNet (d), and our detector (e) to obtain salient objects. Meanwhile, (g) to (j) are corruption based on different salient prior identification. It can be seen that the corruption determined by our method is closest to the ground truth mask (f).

4.3. Validation of the corruption recognition module

We integrate the edge utilization mechanism on DUTS-TR to train our salient prior catcher. We use RAS, U2Net, PoolNet, and our salient detection model to identify salient objects in corrupted images. The qualitative results are shown in Fig. 4(b)–(e). It can be seen that when facing the corrupted images, RAS with the global salient map as the reverse supervision condition can avoid corruption to a certain extent. However, since the influence of the global salient map, RAS will score some background regions. When the salient fusion map is converted into the feature prior map, these scored low-frequency texture regions will interfere with the recognition of corruption. U2Net based on the two-level nested U-structure can capture more salient regions as priors, but its corrupted avoidance ability is weaker than our model slightly. PoolNet does not make full use of boundary information, so when salient regions and corrupted regions are interlaced, it is difficult for PoolNet to separate them from each other. On the contrary, our salient detector with the enhanced edge utilization mechanism, which can avoid corrupted regions effectively.

At the same time, we add corruption to 2000 images on DUTS-TE and perform RTV mapping on them. After that, we use different methods to detect salient objects on these images with and without RTV filtering, and compare the results with their corresponding ground truth salient maps. In order to prevent the corrupted regions from the affecting statistical results, the ground truth salient maps calculated by our approach do not contain corruption. In order to avoid the mask region occupying too much content and covering the ground truth salient objects, in this test, we only use the mask that occupies 1%–20% of the image. We plot the comparison results as precision recall

Table 2
Quantitative comparison results between our inpainting model and multiple baseline inpainting algorithms.

	PSNR*			SSIM*			MAE†		
Mask	1–20%	20–40%	40–60%	1–20%	20–40%	40–60%	1–20%	20–40%	40–60%
PM [5]	27.350	23.080	16.946	0.9244	0.7100	0.5763	0.0149	0.0329	0.0662
GLN [38]	25.695	19.950	15.866	0.9004	0.6935	0.5628	0.0245	0.0493	0.0792
GCAN [44]	27.150	20.001	16.911	0.9269	0.7613	0.5718	0.0205	0.0469	0.0743
PartialConv [30]	31.030	23.673	19.743	0.9070	0.7310	0.5325	0.0147	0.0325	0.0649
GatedConv [39]	27.119	19.970	16.878	0.9240	0.7583	0.5686	0.0208	0.0472	0.0746
EdgeConnect [13]	29.972	23.321	19.641	0.9603	0.8600	0.6916	0.0151	0.0328	0.0650
PRVS [45]	33.435	24.560	20.625	0.9655	0.8875	0.7345	0.0125	**0.0314**	0.0644
StructureFlow [12]	32.001	25.176	21.061	0.9698	0.8989	0.7539	0.0168	0.0340	0.0665
RFR [40]	**33.960**	24.473	20.256	0.9610	0.8622	0.6763	**0.0119**	0.0343	0.0700
Ours	33.356	**25.961**	**21.623**	**0.9705**	**0.9079**	**0.7583**	0.0166	0.0327	**0.0639**

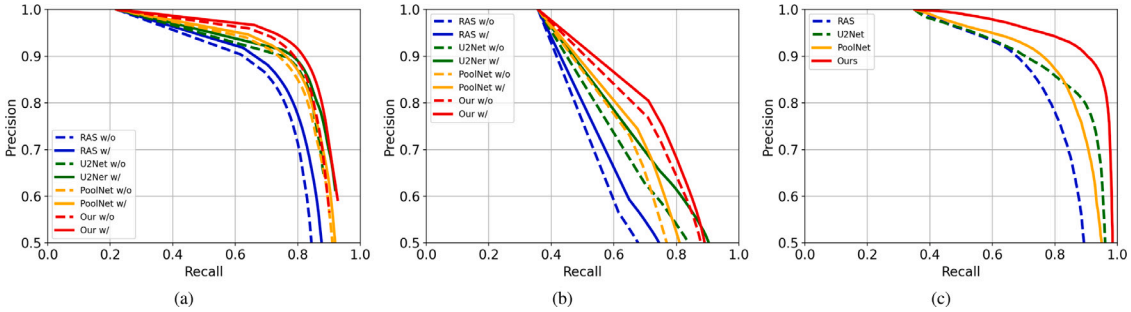


Fig. 5. PR curves for the performance verification of corruption avoidance and corruption recognition of the models with and without RTV images as inputs.

(PR) curves (as Fig. 5(a)). It can be seen that RTV can enhance the anti-corruption ability of the network. U2Net and PoolNet have similar performance in the face of corrupted images, and our improved model can make the traditional PoolNet more suitable for the corruption avoidance task of this paper. After that, we add masks and corrupted contents that account 40%–60% of regions to the 2000 images in DUTS-TE, convert them into RTV maps, and continue to use different methods to detect their salient maps. We compare the results of using and not using RTV on the inverted color maps of the mask maps (uncorrupted regions) to verify the corruption avoidance ability of our model. We plot the results into PR curves (as Fig. 5(b)). It can be seen that our model in the corrupted images is better than these 3 baseline methods.

Furthermore, we use the different salient maps extracted by these 3 baseline models and our model as the corresponding prior conditions for corruption recognition. The visualization results of different corruption recognizers are shown in Fig. 4(g)–(j). It can be seen that the corruption detection results with our salient maps as the priors are closest to the ground truth masks. Correspondingly, corrupted regions will be missed or misdetected by these 3 baseline methods. We detect 2000 corrupted images with the corruption of 40%–60% on DUTS-TE and compare them with their ground truth masks under the evaluation indicators of SSIM, MAE, and MSE. SSIM is the structural similarity index. MAE is used to reflect the actual situation of the predicted value error. MSE is the mean square of the predicted value and the ground truth value reflecting the total robustness of the algorithm. The evaluation process is divided into two parts. In the first part, we compare the similarity discriminant score maps generated to identify corruption to compare with the ground truth masks directly. In the second part, we compare the generated binary maps with the ground truth masks. The relevant results are shown in Table 1. As can be seen that although the MSE index is higher in the similarity discrimination pattern since larger regions of salient regions extracted by U2Net, while our model is still outperforms these 3 baseline methods. Moreover, we draw PR curves (as Fig. 5(c)) to compare the similarity maps and the ground truth masks to further prove the scientificity and validity of our model.

4.4. Validation of the content inpainting module

Since the different scenarios, we train a set of two-stage inpainting models in this paper on Places2 and CelebA separately. We reserve 5000 images from each dataset for model verification. We do not perform model training on the Oxford Building, but we use the model trained on Places2 to test the images of the Oxford Building to verify the processing effect of our total inpainting model on high-resolution images. In order to demonstrate the scientificity and effectiveness of our inpainting model, we conduct comparative experiments with multiple advanced inpainting methods. The qualitative results are shown in Fig. 6. It can be seen that the PM based on random sample block filling can ensure that the color, texture, resolution of the filled contents are in harmony with the existing regions. However, it is difficult for PM to capture and reconstruct high-level semantics. Therefore, when it faces an extensive region where the image content is missing, it will cause the result to lose the rationality. The generator used by GCAN exploits a two-stage coarse-and-fine architecture, which can be regraded as a combination of two context encoder modules. As for the discriminator, GCAN inherits GLN's global-and-local design solutions. It can be seen that GCAN can capture the high-level semantics of the image existing region, and the contextual attention module can restore content which approximates the ground truth structure. However, since the limitations of the GCAN's optimization function and training strategy, the inpainting results are over-smoothing. GatedConv improves GCAN and PartialConv based on irregular corruption characteristics, which can refine the color and texture of the result more realistic. However, some obvious chessboard artifacts by GatedConv are difficult to filter out. An additional edge map label is added in the middle of the two-stage EdgeConnect, where can guide the generated content with a more reasonable structure. However, for the second-stage of EdgeConnect, it is difficult to predict a color image based on a binary map where only a few pixels have actual values. Therefore, it can be seen that there may be some abrupt and uncoordinated contents in the results of EdgeConnect. StructureFlow changes the internal edge map label of EdgeConnect to the structure map label, and obtains relatively fine results. However, since the limitations of the network receptive field

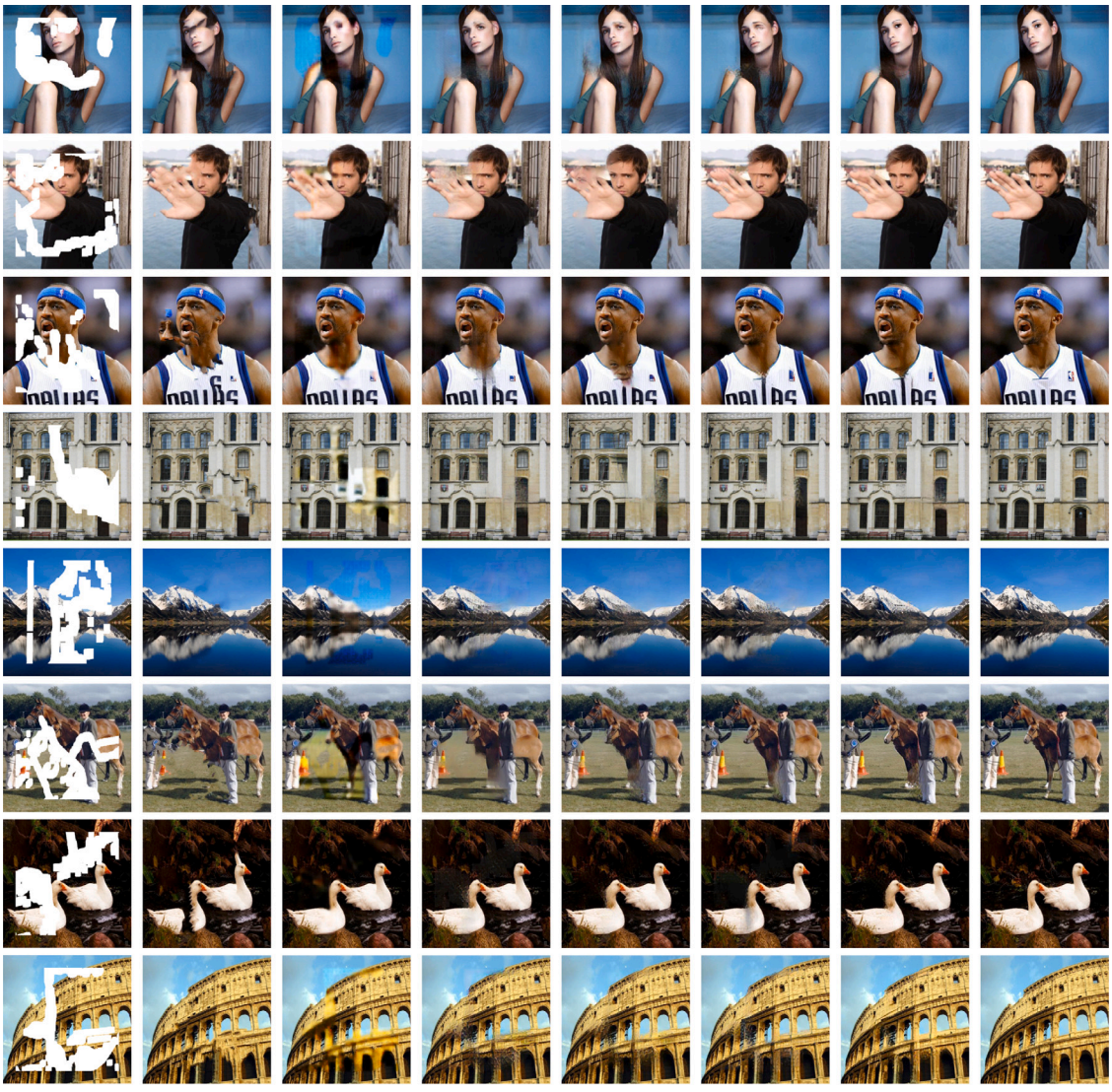


Fig. 6. Qualitative comparison results between our approach and multiple baseline inpainting algorithms. Columns from left to right: (a) corrupted images, (b) results of PM [5], (c) GCAN [44], (d) GatedConv [39], (e) EdgeConnect [13], (f) StructureFlow [12], (g) our inpainting module, (h) ground truth images. From top to bottom, the first three rows are from CelebA [33], the fourth row is from Oxford Building [34], and the last four rows are from Places2 [32].

and feature fusion performance in the skip connections of StructureFlow, some details are still unsatisfactory. Moreover, it can be seen that since the introduction of FAM, our model can suppress checkerboard artifacts better than StructureFlow. Meanwhile, our scheme based on the intact region in the salient map can also make the predicted texture closer to the existing texture, and the final output image will be more coordinated.

In addition, we compare our method with multiple inpainting methods under the peak signal-to-noise ratio (PSNR), SSIM and MAE indicators based on 2000 images of the Places2 dataset. PSNR is used to measure the quality of the generated map by calculating the difference between the result and the ground truth. The quantitative results are shown in Table 2.

It can be seen that, compared under the PSNR indicator, when the mask region occupies less than 20% of the image, the performance of our approach is inferior to the RFR slightly. However, as the mask region increases, our model can achieve the best results. This is because that the baseline methods cannot identify the corruption and can only rely on the mask label to inpaint the corruption, so the non-masked part comes from the original input image. Therefore, when there are fewer masks and more parts from the original image, the relevant indicators will be satisfactory relatively. However, our approach performs global reconstruction. Even if the mask region is small, our model is used to compare all the reconstructed content with the ground truth map, so the evaluation performance is slightly lower. As the mask region increases, this advantage from the ground truth will decrease, and our improved network can have more space for display, so the index will be higher than the state-of-the-art methods. The same is true for the principle of MAE evaluation results. Since MAE is more sensitive to changes in low-level pixels, our method is only superior to the baseline methods when the corruption is 40%–60%.

4.5. Ablation studies

In this section, we conduct a series of ablation studies to demonstrate the model design of this article. First, we verify the effectiveness of the RTV model in this paper. We use our RTV map, the filter map with a Gaussian kernel of 9, and the filter map with a Gaussian kernel of 49 as the internal label of our two-stage inpainting network, respectively. The comparison results are shown in Fig. 7. As can be seen from Fig. 7(b)–(e) that the map obtained after filtering with the Gaussian kernel of 9 is similar to the RTV map we set in terms of

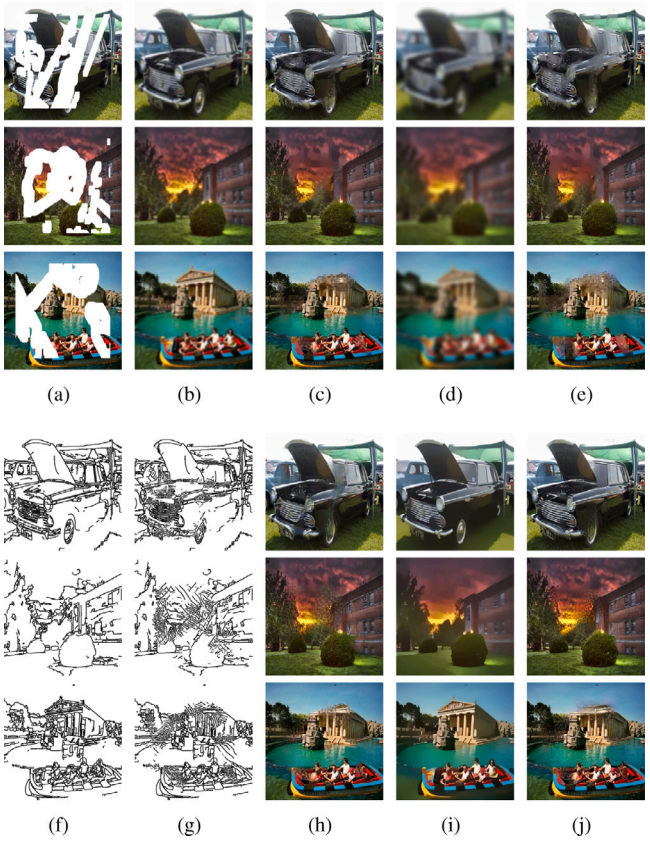


Fig. 7. The comparison results of the internal label of our inpainting model, the Gaussian filter label, and the edge label. Images with masks (a), filter maps with the Gaussian kernel as 9 (b), inpainting results of the inter label of filtered images with the Gaussian kernel as 9 (c), filter maps with the Gaussian kernel as 49 (d), results of the inter label of filtered images with the Gaussian kernel as 49 (e), ground truth edge maps (f), edge maps generated by EdgeConnect (g), results of the inter label of edge maps (h), our RTV maps (i), our final inpainting results (j).

Table 3
The significance of FAM and salient map concatenation module.

	w/o Salient	w/o FAM	Ours
PSNR*	25.785	25.062	**26.163**
SSIM*	0.9079	0.8919	**0.9158**
MAE†	0.0245	0.0265	**0.0234**

filtering noise and high-frequency texture, but it will cause global blurring and it is difficult to retain sharp edges. Therefore, the result of global filter will be biased toward fuzzy. With the increase of the Gaussian kernel size, although the color information of the map as the intermediate label can still be retained, the result will become more blurred.

Then we compare the performance of our RTV map with EdgeConnect's internal edge labels. It can be seen from Fig. 7(f)–(j) that compared with the edge map obtained by using the Canny operator (threshold interval is set from 1 to 25) to perform edge detection on the ground truth image. The generated edges are distributed evenly in the texture based on the background sparsity, which makes it difficult for EdgeConnect to express semantic features effectively. Moreover, the color generating process based on the binary edge map is much more complicated than the content restoring process based on the 3-channel color map, so we do not apply this internal label pattern.

After that, 3 models are trained on Places2, including the model without the salient concatenation module, the model without the FAM, and our total model, respectively. We count 2000 images processed on these 3 models in the PSNR, SSIM and MAE indicators, and the results

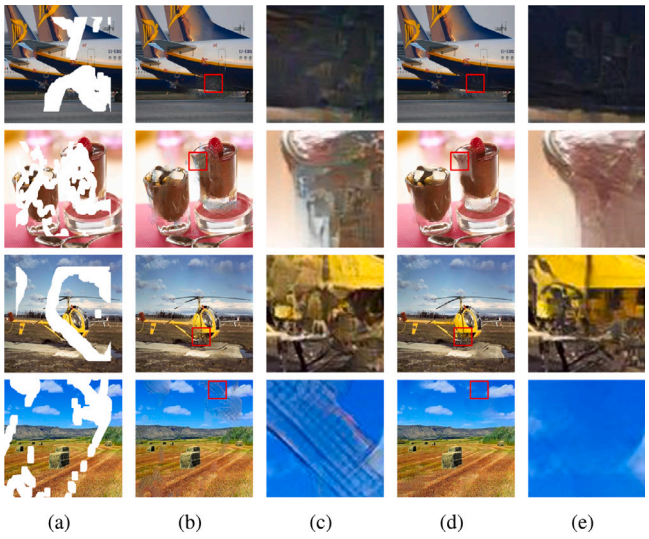


Fig. 8. The significance of the perceptual matching module. Images with masks (a), results (b) and their partial enlarged views (c) of the uncascaded perceptual matching module, results (d) and their partial enlarged views (e) of our model.

are shown in Table 3. It can be seen that the first two modules play the certain role in improving system performance, and the improvement effect based on FAM is more obvious.

Finally, we show the several groups of results and their partial method diagrams obtained by inpainting with and without using the perceptual matching module , as shown in Fig. 8. It can be seen that when GAN is used to guide fine texture inpainting in image processing, it will produce some artifacts or over-fitting details, and our perceptual matching module can eliminate these details.

5. Conclusion

This paper proposes a novel generative image inpainting network, which can get rid of the limitations of external input labels, and can automatically identify, locate, and inpaint irregular corruption. We use the salient object fusion map and the RTV structure feature map to integrate the corruption recognition module and the content inpainting module. The salient information as the prior can be used to avoid corruption validly, and the original image can be filtered properly. Moreover, the edge detection module is integrated to improve the intact region utilization of our algorithm, and the FAM feature aggregation module is invoked to improve our network receptive field. The experimental results verify that irregular and non-binary image corruption can be identified effectively, and the inpainting effect is better than the state-of-the-art methods. However, our inpainting results may cause differences in brightness values between the inpainted region and the uncorrupted region. Meanwhile, for images with very complex semantics or strong stylization, our algorithm may still cause artifacts. In addition, the dataset is the key to the continued research of our algorithm, the related dataset for natural corruption is still close to blank at present. There will be directions for us to continue research.

Declaration of competing 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.

Acknowledgments

This research is supported by the National Natural Science Foundation of China under Grant 61673276.

References

[1] T. Chan, J. Shen, Variational image inpainting, Comm. Pure Appl. Math. (2005) 579–619.

[2] L. Hoeltgen, A. Kleefeld, I. Harris, M. Breuss, Theoretical foundation of the weighted laplace inpainting problem, Appl. Math. (2019) 281–300.

[3] Y. Chen, H. Zhang, L. Liu, J. Tao, Q. Zhang, K. Yang, R. Xia, J. Xie, Research on image inpainting algorithm of improved total variation minimization method, J. Ambient Intell. Humaniz. Comput. (2021) 1–10.

[4] D. Thanh, V. Prasath, S. Dvoenko, An adaptive image inpainting method based on euler's elastica with adaptive parameters estimation and the discrete gradient method, Signal Process. (2021) 107797.

[5] C. Barnes, E. Shechtman, A. Finkelstein, D. Goldman, PatchMatch: A randomized correspondence algorithm for structural image editing, ACM Trans. Graph. (2009) 24–33.

[6] D. Ding, S. Ram, J. Rodríguez, Image inpainting using nonlocal texture matching and nonlinear filtering, IEEE Trans. Image Process. (2018) 1705–1719.

[7] D. Helbert, M. Malek, P. Bourdon, P. Carré, Patch graph-based wavelet inpainting for color images, J. Vis. Commun. Image Represent. (2019) 102614.

[8] H. Pen, Q. Wang, Z. Wang, Boundary precedence image inpainting method based on self-organizing maps, Knowl.-Based Syst. (2021) 106722.

[9] C. Hu, Y. Wang, An efficient convolutional neural network model based on object-level attention mechanism for casting defect detection on radiography images, IEEE Trans. Ind. Electron. (2020) 10922–10930.

[10] Y. Chen, L. Liu, J. Tao, X. Chen, R. Xia, Q. Zhang, J. Xiong, K. Yang, X. J., The image annotation algorithm using convolutional features from intermediate layer of deep learning, Multimedia Tools Appl. (2021) 4237–4261.

[11] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, A. Efros, Context encoders: Feature learning by inpainting, in: IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 2536–2544.

[12] Y. Ren, X. Yu, R. Zhang, T. Li, S. Liu, G. Li, Structureflow: Image inpainting via structure-aware appearance flow, in: IEEE/CVF International Conference on Computer Vision, 2019, pp. 181–190.

[13] K. Nazeri, E. Ng, T. Joseph, F. Qureshi, M. Ebrahimi, Edgeconnect: Generative image inpainting with adversarial edge learning, 2019, arXiv preprint arXiv: 1901.00212.

[14] Z. Zhang, X. Pan, S. Jiang, P. Zhao, High-quality face image generation based on generative adversarial networks, J. Vis. Commun. Image Represent. (2020) 102719.

[15] J. Jam, C. Kendrick, K. Walker, V. Drouard, J. Hsu, M. Yap, A comprehensive review of past and present image inpainting methods, Comput. Vis. Image Underst. (2020) 103147.

[16] N. Wang, S. Ma, J. Li, Y. Zhang, L. Zhang, Multistage attention network for image inpainting, Pattern Recognit. (2020) 107448.

[17] Y. Shin, M. Sagong, Y. Yeo, S. Kim, S. Ko, Pepsi++: Fast and lightweight network for image inpainting, IEEE Trans. Neural Netw. Learn. Syst. (2020) 252–265.

[18] L. Zhao, Q. Mo, S. Lin, Z. Wang, Z. Zuo, H. Chen, W. Xing, D. Lu, Uctgan: Diverse image inpainting based on unsupervised cross-space translation, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 5741–5750.

[19] Z. Yi, Q. Tang, S. Azizi, D. Jang, Z. Xu, Contextual residual aggregation for ultra high-resolution image inpainting, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 7508–7517.

[20] Y. Chen, L. Liu, J. Tao, R. Xia, Q. Zhang, K. Yang, J. Xiong, X. Chen, The improved image inpainting algorithm via encoder and similarity constraint, Vis. Comput. (2020) 1–15.

[21] H. Shao, Y. Wang, Y. Fu, Z. Yin, Generative image inpainting via edge structure and color aware fusion, Signal Process., Image Commun. (2020) 115929.

[22] N. Wang, W. Wang, W. Hu, A. Fenster, S. Li, Thanka mural ipainting based on multi-scale adaptive partial convolution and stroke-like mask, IEEE Trans. Image Process. (2021) 3720–3733.

[23] N. Wang, Y. Zhang, L. Zhang, Dynamic selection network for image inpainting, IEEE Trans. Image Process. (2021) 1784–1798.

[24] J. Jam, C. Kendrick, V. Drouard, K. Walker, G. Hsu, M. Yap, R-mnet: A perceptual adversarial network for image inpainting, in: IEEE/CVF Winter Conference on Applications of Computer Vision, 2021, pp. 2714–2723.

[25] K. Chen, Y. Wang, C. Hu, H. Shao, Salient object detection with boundary information, in: IEEE International Conference on Multimedia and Expo, 2021, pp. 1–6.

[26] X. Qin, Z. Zhang, H. C., M. Dehghan, O. Zaiane, U2-Net: Going deeper with nested U-structure for salient object detection, Pattern Recognit. (2020) 107404.

[27] L. Xu, Q. Yan, Y. Xia, J. Jia, Structure extraction from texture via relative total variation, ACM Trans. Graph. (2012) 139–148.

[28] W. Wang, Y. Jia, Q. Wang, P. Xu, An image enhancement algorithm based on fractional-order phase stretch transform and relative total variation, Comput. Intell. Neurosci. (2021).

[29] J. Liu, Q. Hou, M. Cheng, J. Feng, J. Jiang, A simple pooling-based design for real-time salient object detection, in: IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 3917–3926.

[30] G. Liu, F. Reda, K. Shih, T. Wang, A. Tao, B. Catanzaro, Image inpainting for irregular holes using partial convolutions, in: European Conference on Computer Vision, 2018, pp. 89–100.

[31] C. Li, M. Wand, Precomputed real-time texture synthesis with Markovian generative adversarial networks, in: European Conference on Computer Vision, 2016, pp. 702–716.

[32] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, A. Torralba, Places: A 10 million image database for scene recognition, IEEE Trans. Pattern Anal. Mach. Intell. (2017) 1452–1464.

[33] Z. Liu, P. Luo, X. Wang, X. Tang, Deep learning face attributes in the wild, in: IEEE International Conference on Computer Vision, 2015, pp. 3730–3738.

[34] F. Radenovic, A. Iscen, G. Tolias, Y. Avrithis, O. Chum, Revisiting Oxford and Paris: Large-scale image retrieval benchmarking, in: IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 5706–5715.

[35] L. Wang, H. Lu, Y. Wang, M. Feng, D. Wang, B. Yin, X. Ruan, Learning to detect salient objects with image-level supervision, in: IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 136–145.

[36] C. Yang, X. Lu, Z. Lin, E. Shechtman, O. Wang, H. Li, High-resolution image inpainting using multi-scale neural patch synthesis, in: IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 6721–6729.

[37] Y. Zhang, Y. Wang, J. Dong, L. Qi, H. Fan, X. Dong, M. Jian, H. Yu, A joint guidance-enhanced perceptual encoder and atrous separable pyramid-convolutions for image inpainting, Neurocomputing (2020) 1–12.

[38] S. Iizuka, E. Simo-Serra, H. Ishikawa, Globally and locally consistent image completion, ACM Trans. Graph. (2017) 1–14.

[39] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, T. Huang, Free-form image inpainting with gated convolution, in: IEEE International Conference on Computer Vision, 2019, pp. 4471–4480.

[40] J. Li, N. Wang, L. Zhang, B. Du, D. Tao, Recurrent feature reasoning for image inpainting, in: IEEE Conference on Computer Vision and Pattern Recognition, 2020, pp. 7760–7768.

[41] J. Johnson, A. Alahi, L. Fei-Fei, Perceptual losses for real-time style transfer and super-resolution, in: European Conference on Computer Vision, 2016, pp. 694–711.

[42] P. Isola, J. Zhu, T. Zhou, A. Efros, Image-to-image translation with conditional adversarial Networks, in: IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 1125–1134.

[43] Y. Chen, L. Liu, V. Phonevilay, K. Gu, R. Xia, J. Xie, Q. Zhang, K. Yang, Image super-resolution reconstruction based on feature map attention mechanism, Appl. Intell. (2021) 1–14.

[44] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, T. Huang, Generative image inpainting with contextual attention, in: IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 5505–5514.

[45] J. Li, F. He, L. Zhang, B. Du, D. Tao, Progressive reconstruction of visual structure for image inpainting, in: IEEE International Conference on Computer Vision, 2019, pp. 5962–5971.

[46] Y. Liu, J. Pan, Z. Su, Deep blind image inpainting, in: International Conference on Intelligent Science and Big Data Engineering, 2019, pp. 128–141.

[47] Y. Wang, Y. Chen, X. Tao, J. Jia, VCNet: A robust approach to blind image inpainting, in: European Conference on Computer Vision, 2020.

[48] T. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, S. Belongie, Feature pyramid networks for object detection, in: IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 936–944.

[49] Y. Liu, M. Cheng, X. Hu, J. Bian, L. Zhang, X. Bai, J. Tang, Richer convolutional features for edge detection, IEEE Trans. Pattern Anal. Mach. Intell. (2019) 1939–1946.

[50] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.

[51] O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, 2015, pp. 234–241.

[52] S. Chen, X. Tan, B. Wang, H. Lu, X. Hu, Y. Fu, Reverse attention-based residual network for salient object detection, IEEE Trans. Image Process. (2020) 3763–3776.

[53] D. Kingma, J. Ba, Adam: A method for stochastic optimization, 2014, arXiv preprint arXiv:1412.6980.

[54] T. Miyato, T. Kataoka, M. Koyama, Y. Yoshida, Spectral normalization for generative adversarial networks, in: International Conference on Learning Representations, 2018.