Bridging Knowledge Gap Between Image Inpainting and Large-Area Visible Watermark Removal

Yicheng Leng[1, 2], **Chaowei Fang**[1*], **Junye Chen**[3], **Yixiang Fang**[2], **Sheng Li**[4], **Guanbin Li**[3, 5]

[1] School of Artificial Intelligence, Xidian University, Xi'an, China
[2] School of Data Science, The Chinese University of Hong Kong, Shenzhen, China
[3] School of Computer Science and Engineering, Research Institute of Sun Yat-sen University in Shenzhen, Sun Yat-sen University, Guangzhou, China
[4] Afirstsoft, Shenzhen, China
[5] GuangDong Province Key Laboratory of Information Security Technology

Abstract

Visible watermark removal which involves watermark cleaning and background content restoration is pivotal to evaluate the resilience of watermarks. Existing deep neural network (DNN)-based models still struggle with large-area watermarks and are overly dependent on the quality of watermark mask prediction. To overcome these challenges, we introduce a novel feature adapting framework that leverages the representation modeling capacity of a pre-trained image inpainting model. Our approach bridges the knowledge gap between image inpainting and watermark removal by fusing information of the residual background content beneath watermarks into the inpainting backbone model. We establish a dual-branch system to capture and embed features from the residual background content, which are merged into intermediate features of the inpainting backbone model via gated feature fusion modules. Moreover, for relieving the dependence on high-quality watermark masks, we introduce a new training paradigm by utilizing coarse watermark masks to guide the inference process. This contributes to a visible image removal model which is insensitive to the quality of watermark mask during testing. Extensive experiments on both a large-scale synthesized dataset and a real-world dataset demonstrate that our approach significantly outperforms existing state-of-the-art methods. The source code is available in the supplementary materials.

Introduction

Visible watermarks serve a pivotal role in asserting image ownership and copyright. Yet, they can obscure vital content, especially during image editing or in situations where tampered information within images plays a crucial role. Studying on visible watermark removal contributes to evaluating the resilience of watermarks. This paper aims to develop a deep neural networks (DNN) based model which revolves around two objectives: watermark cleaning and background content restoration.

Prior visible watermark removal works, depicted in Figure 1 (a), embrace a multi-task framework that employs various decoder branches to implement sub-tasks such as watermark mask segmentation, watermark decomposition, and background content restoration. Notable models like (Hertz et al. 2019; Cun and Pun 2021; Liang et al. 2021; Sun, Su, and Wu 2023) are instrumental in this approach. However, existing methods still face two challenges: 1) In real-world images, areas of watermarks can be very large. Those large-area watermarks inevitably accentuate the complexity of background content recovery, especially when they are overlaid on regions with intricate visual content. Due to insufficient representation modeling ability, the performance of existing models still have substantial improvement room when coping with large-area watermarks. 2) Since DNN-based models can usually overfit the training data, they are able to generate high-quality watermark masks on training images. This makes the learned models dependent to quality of the predicted watermark masks on real-world images. Missed detection leaves discernible watermark traces, while too many false alarms cause confusion to the background content recovery process.

In light of these challenges, we are dedicated in borrowing the rich knowledge of the image inpainting model to address the visible watermark removal task. Recent advances in the image inpainting field demonstrate that DNN models are able to fill in missing regions of images with plausible visual content. Employing the knowledge of the image inpainting model to foster visible watermark removal model is a direction worth exploring. However, distinct to the image inpainting task, the residual background content beneath watermarks can provide valuable prompts for background content restoration. To bridge the knowledge gap between visible watermark removal and image inpainting, we propose a feature adapting framework which can effectively fuse the information of residual background content into intermediate features of a pre-trained image inpainting model, as depicted in Figure 1 (b). First, to capture the information of the residual background content residing in the watermarked input, we set up a watermark component cleaning branch which directly predicts an image precluding the watermark information from the input image. Moreover, we construct the other branch to further embed the background content. The intermediate features of the above two branches can provide valuable prompt information for

*Corresponding author.
Copyright © 2025, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

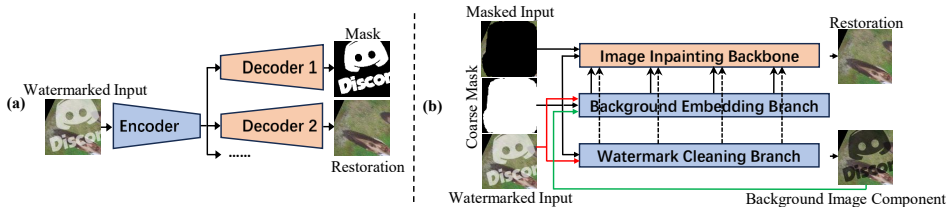


Figure 1: (a) Existing multi-task based methods such as (Cun and Pun 2021; Liang et al. 2021; Sun, Su, and Wu 2023) adopt a shared encoder and multi-branch decoder for implementing sub-tasks such as watermark segmentation, watermark decomposition, and background restoration; (b) We propose a novel solution through adapting an image inpainting backbone with prompt information extracted from a watermark component cleaning and a background content embedding branches. Moreover, we relieve the dependence on high-quality watermark masks by leveraging coarse masks to guide the inference process.

repairing the regions destroyed by watermarks. Hence, we utilize gated fusion modules to merge features extracted by the two branches into intermediate features of the image inpainting backbone. With help of the above feature adapting framework, we build up a novel watermark removal model which can combine the prior knowledge of image inpainting and prompt information of residual background content beneath transparent watermarks.

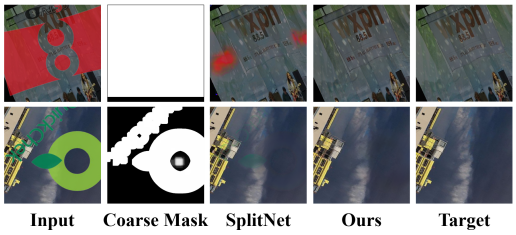


Figure 2: The first column showcases two input examples which are covered by large-area watermarks. Though coarse watermark masks (second column) are available, our method (fourth column) can effectively remove these watermarks and accurately recover the background, showing significant superiority over SplitNet (third column).

Awareness of the watermark region is critical to explore the knowledge of the image inpainting, while preventing the watermark component from affecting the background restoration process. However, the precise segmentation of watermark is also very challenging. In this study, we relieve the dependence on high-quality watermark detection performance by allowing coarse watermark masks to guide the image restoration process. Straightforwardly, we synthesize moderately corrupted watermark masks and integrate them with watermarked images as inputs. Such a training strategy helps foster a model insensitive to the quality of the segmented watermark masks. During practical usage, a rather coarse watermark mask is sufficient for the learned model to realize watermark removal. Figure 2 displays examples of watermark removal accomplished by our model. We conduct extensive experiments on a large-scale synthesized dataset and a real-world dataset, and the results demonstrate that our proposed method achieves state-of-the-art performance.

Our main contributions are summarized as follows:

- We propose a novel feature adapting framework for tackling the large-area watermark removal problem, capable of combing the prior knowledge of a pre-trained image inpainting model and prompt information of residual background content.
- A new training paradigm is devised for improving the robustness of watermark removal model against low-quality watermark segmentation masks, enhancing the stability in coping with real-world watermarked images.
- Through comprehensive evaluations on a large-scale synthesized dataset and a real-world dataset, our method sets a new state-of-the-art benchmark for large-area visible watermark removal.

Related Work

Visible Watermark Removal

Visible watermark removal involves restoring images that are covered by watermarks to their original watermark-free state. This task is challenging since watermarks have diverse shapes, areas, colors, and transparency levels. Typical methods in this field rely on a multi-task pipeline. Cheng et al. (2018) learn an object detection model (Redmon and Farhadi 2017) to locate watermarks and then construct a U-Net (Ronneberger, Fischer, and Brox 2015) model for transforming watermarked input to watermark-free output.

Hertz et al. (2019) propose a method that utilizes a shared encoder with separate decoders to predict the watermark image, watermark mask, and background image, enhancing watermark removal performance while maintaining low network complexity. Similarly, Cun and Pun (2021) leverage task-specific attention mechanisms to create multiple decoder branches within a shared parameter space, reducing parameter redundancy. They also introduce a refinement stage to further improve restoration quality. Liang et al. (2021) adopt dual decoder branches for watermark mask prediction and background restoration, using the predicted watermark mask to guide feature extraction in the background restoration branch, effectively enhancing features in regions affected by the watermark. Additionally, Sun, Su, and Wu (2023) utilize contrastive learning with multi-head attention (Vaswani et al. 2017; Dosovitskiy et al. 2020) to disentangle watermark and background information. Despite these advancements, existing methods struggle to re-

store images heavily corrupted by large-area watermarks and rely heavily on high-quality watermark masks. Given the strengths of image inpainting models in leveraging long-range context for image repair, our approach focuses on adapting pre-trained image inpainting models to tackle the challenges of large-area watermark removal.

Image Inpainting

Image inpainting techniques (Bertalmio et al. 2000) focus on filling missing parts of an image with content that matches the surrounding visual context, and many studies (Suvorov et al. 2022; Li et al. 2022; Liu et al. 2018; Dong, Cao, and Fu 2022; Zuo et al. 2023; Yang, Chen, and Liao 2023; Liu et al. 2023) have explored this area extensively. These methods can be applied to visible watermark removal by treating watermark regions as missing parts. However, they fail to leverage the residual background beneath transparent watermarks. To address this limitation, we introduce a dual-branch design that adapts the inpainting model's intermediate features by cleaning watermark components and embedding background content.

Methodology

This paper endeavors to develop a model with the capacity to convert a watermarked image into its watermark-free version, in which a coarse mask is given to specify the watermark region. We define the input watermarked image be $\mathbf{X} \in \mathbb{R}^{h\times w\times 3}$ and the coarse watermark mask be $\mathbf{M} \in \{0,1\}^{h\times w\times 1}$, where h and w represent the height and width of the input image, respectively.

Method Overview

The key challenges in visible watermark removal are the thorough elimination of watermark components and the seamless reconstruction of the damaged background. Global context information is particularly critical for accurately restoring backgrounds obscured by large-area watermarks. To address these challenges, we adopt the pre-trained image inpainting model LaMa (Suvorov et al. 2022) as our backbone due to its strong performance in reconstructing extensive masked regions and capturing global context through fast Fourier convolution (FFC) (Chi, Jiang, and Mu 2020). This capability is especially advantageous for removing visible watermarks that obscure significant background details, ensuring the naturalness and authenticity of the restored content. To further enhance LaMa's performance, as shown in Fig. 3, we introduce a watermark cleaning branch to remove watermark interferences from the input image and produce a cleaned background component image. Additionally, a background content embedding branch extracts features from both the original input and the cleaned background image to support the reconstruction of affected regions. The features from these branches are then fused with the LaMa backbone via gated fusion modules, ultimately generating the final restored background image.

Watermark Cleaning and Background Embedding

The image inpainting model effectively captures global context for reconstructing missing regions, but it overlooks crucial residual background information under the visible watermarks. To address this, we introduce a feature adapting framework that enhances the inpainting model's intermediate features using insights from watermark component cleaning and background content embedding branches.

Watermark Component Cleaning (WCC) Branch Considering that the watermark component is irrelevant to the background content, we first establish a branch to subtract it from the input image. Since the information from the whole image is needed for recognizing watermarks, the long-distance relations become crucial. Therefore, we design the watermark component cleaning branch to effectively capture global context information.

As depicted in the bottom section of Figure 3, regarding the concatenation of $\mathbf{X}$ and $\mathbf{M}$ as the input, an encoder with the same architecture of LaMa's encoder is utilized to extract a resolution-reduced feature map $\mathbf{F}_1^{wcc} \in \mathbb{R}^{h'\times w'\times d}$. Since the transposed attention module (Zamir et al. 2022) has outstanding advantages at extracting global context information, we stack three transposed attention modules to further enhance $\mathbf{F}_1^{wcc}$. Suppose the outcome of the i-th transposed attention module be $\mathbf{F}_{i+1}^{wcc}$. The calculation process of $\mathbf{F}_{i+1}^{wcc}$ is summarized as follows:

1) A 1×1 convolution layer and 3×3 depth-wise convolution layer (Chollet 2017) are employed to infer the query, key, and value variables, denoted by $\mathbf{Q}$, $\mathbf{K}$, and $\mathbf{V} \in \mathbb{R}^{h'\times w'\times d}$ respectively, from $\mathbf{F}_i^{wcc}$. The calculation process can be formulated as:
$$[\mathbf{Q},\mathbf{K},\mathbf{V}] = \texttt{DConv}_{\texttt{3x3}}(\texttt{Conv}_{\texttt{1x1}}(\mathbf{F}_i^{wcc})), \tag{1}$$
2) The horizontal and vertical dimensions of $\mathbf{Q}$ and $\mathbf{K}$ are unfolded into a single dimension, resulting in $\mathbf{q}$ and $\mathbf{k} \in \mathbb{R}^{(h'w')\times d}$, respectively. Namely, $\mathbf{q} = \texttt{unfold}(\mathbf{Q})$, and $\mathbf{k} = \texttt{unfold}(\mathbf{K})$, where $\texttt{unfold}(\cdot)$ represents the space-to-channel unfolding operation. Then, a cross-channel correlation map $\mathbf{S}$ is inferred by (α is a constant):
$$\mathbf{S} = \texttt{Softmax}(\mathbf{q}^{\mathrm{T}}\mathbf{k}/\alpha), \tag{2}$$
3) Upon the calculation of $\mathbf{S}$, $\mathbf{F}_{i+1}^{wcc}$ is generated by:
$$\mathbf{F}_{i+1}^{wcc} = \mathbf{F}_i^{wcc} + \texttt{Conv}_{\texttt{1x1}}(\texttt{fold}(\texttt{unfold}(\mathbf{V})\mathbf{S})), \tag{3}$$
where $\texttt{fold}(\cdot)$ denotes the channel-to-space folding.

At the end of this branch, a decoder expands the feature map resolution to generate a background component image, $\mathbf{C}^{bkg}$. This branch serves two key purposes: identifying residual background content and generating features essential for restoring the background image.

Background Content Embedding (BCE) Branch To more effectively leverage the prompt information contained in the generated residual background content, we introduce a Background Content Embedding (BCE) branch. To address potential loss of background information by the background component cleaning branch, $\mathbf{X}$ and $\mathbf{M}$ are reused to enrich the input of $\mathbf{C}^{bkg}$. As shown in the middle section of Figure 3, the BCE branch comprises an encoder followed by three transposed attention modules. The resulting feature maps, denoted as $\mathbf{F}_1^{bce}$, $\mathbf{F}_2^{bce}$, $\mathbf{F}_3^{bce}$, and $\mathbf{F}_4^{bce}$, explicitly capture the information of the background content, which are paramount for background reconstruction.

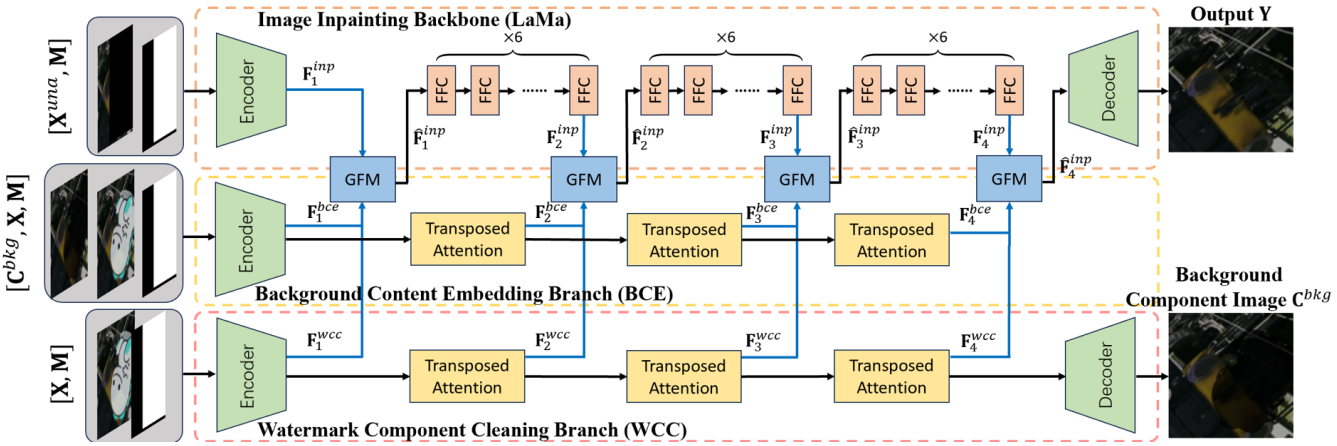


Figure 3: Overview of our framework which adapts an image inpainting backbone model, LaMa, to address the visible watermark removal task. Given an input image $\mathbf{X}$ and a coarse mask $\mathbf{M}$, the watermark component cleaning branch (WCC) is employed to preclude the interference information brought by watermarks from the input image. Then, a background content embedding branch (BCE) is used to extract prompt features from the background component image and the original input image. We enhance the intermediate features of LaMa with these feature extracted by WCC and BCE branches.

Backbone Model Adaptation

Introduction to Backbone Model The architecture of LaMa is composed of three stages: an encoder for extracting preliminary features; an intermediate feature enhancement module consisting of 18 FFC modules; and a decoder.

First, we remove the content inside the watermark mask $\mathbf{M}$ of $\mathbf{X}$, deriving $\mathbf{X}^{una} = (1-\mathbf{M}) \circ \mathbf{X}$, where $\circ$ denotes the broadcast Hadamard product. Regarding the concatenation of $\mathbf{X}^{una}$ and $\mathbf{M}$ as the input, the encoder produces a feature map $\mathbf{F}_1^{inp} \in \mathbb{R}^{h' \times w' \times d}$, where $h' = h/32$ and $w' = w/32$.

We divide the 18 FFC modules of the intermediate feature enhancement module into three groups with each containing six FFC modules. Define the outcome of the i-th group of FFC modules as $\mathbf{F}_{i+1}^{inp} \in \mathbb{R}^{h' \times w' \times d}$. The information of the residual background content provides valuable hints for recovering the regions destroyed by watermarks. To take advantage of such kind of information, we employ the features extracted by WCC and BCE branches to enhance each $\mathbf{F}_i^{inp}$ with help of gated fusion modules (GFM). As shown by Figure 3, every GFM module enhances one intermediate feature map with a pair features extracted by WCC and BCE. We denote the enhanced counterpart of $\mathbf{F}_i^{inp}$ be $\hat{\mathbf{F}}_i^{inp}$. These enhanced feature maps are regarded as the input for next FFC module groups. The final enhanced feature map $\hat{\mathbf{F}}_4^{inp}$ is fed into the decoder, deriving the final output $\mathbf{Y}$.

Gated Fusion Module The intermediate features from backbone lack modelling of the residual background information. Therefore, we devise a gated fusion module (GFM) to combine features from WCC and BCE with output features from FFC module groups, inspired by (Jin et al. 2023).

As shown in Figure 3, four GFM-s are incorporated to enhance the intermediate feature maps $\{\mathbf{F}_i^{inp}\}_{i=1}^4$ of the image inpainting backbone. The design of GFM is illustrated by Figure 4. The i-th GFM incorporates $\mathbf{F}_i^{wcc}$ and $\mathbf{F}_i^{bce}$ to

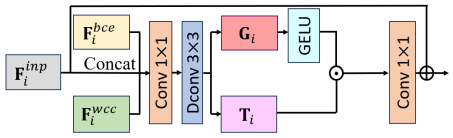


Figure 4: Illustration of the gated fusion module (GFM).

adapt $\mathbf{F}_i^{inp}$. First, a 1×1 convolution layer and a 3×3 depth-wise convolution layer are utilized to process the concatenation of $\mathbf{F}_i^{wcc}$, $\mathbf{F}_i^{bce}$, and $\mathbf{F}_i^{inp}$, resulting in a gate map $\mathbf{G}_i$ and a temporary feature map $\mathbf{T}_i$, namely,

$$[\mathbf{G}_i, \mathbf{T}_i] = \texttt{DConv}_{\texttt{3x3}}(\texttt{Conv}_{\texttt{1x1}}([\mathbf{F}_i^{wcc}, \mathbf{F}_i^{bce}, \mathbf{F}_i^{inp}])). \quad (4)$$

The final output of the i-th GFM is calculated through:

$$\hat{\mathbf{F}}_i^{inp} = \mathbf{F}_i^{inp} + \texttt{Conv}_{\texttt{1x1}}(\texttt{GELU}(\mathbf{G}_i) \circ \mathbf{T}_i), \quad (5)$$

where $\texttt{GELU}(\cdot)$ represents the Gaussian error linear unit function (Hendrycks and Gimpel 2016).

The GFM's effectiveness lies in its ability to highlight relevant features while suppressing less important ones. This is advantageous in our context, where different branches address complementary aspects of watermark removal. The GFM's gating mechanism enables the model to concentrate on features crucial for reconstructing the watermarked region while filtering out irrelevant ones.

Utilization of Coarse Watermark Mask

Given the diverse appearances and shapes of watermarks, pinpointing their pixel-wise locations is challenging. Existing methods like SLBR and SplitNet, which use specific modules for watermark segmentation, struggle with generalization to real-world images, as shown in Figure 5. To mitigate this issue, we use coarse watermark masks to guide the removal process. During training, we create these coarse

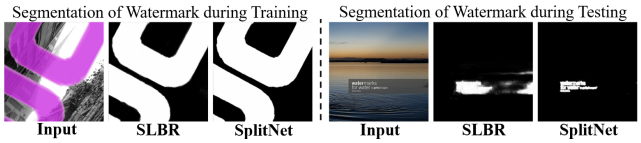


Figure 5: Watermark segmentation generated by blind visible watermark removal methods SLBR and SplitNet.

masks by applying random erosion or dilation to ground-truth masks. For testing, we either manually outline the watermark region or use a segmentation model to generate the mask, which then guides the background restoration.

Loss Function

To balance various objectives, including pixel-level accuracy, perceptual quality, and reality, the designed loss function for training our model encompasses three types of constraints, inspired by (Suvorov et al. 2022). First, we use the L_1 loss to improve the pixel-level accuracy of predicted background component image $\mathbf{C}^{bkg}$ and the final restored watermark-free image $\mathbf{Y}$:

$$L_{pixel} = ||\mathbf{Y} - \mathbf{G}^{wf}||_1 + ||\mathbf{C}^{bkg} - \mathbf{G}^{bkg}||_1, \quad (6)$$

where $\mathbf{G}^{wf}$ and $\mathbf{G}^{bkg}$ represent the ground-truth watermark-free image and background component image.

The perceptual loss (Johnson, Alahi, and Fei-Fei 2016; Zhang et al. 2018) is also incorporated to improve the perceptual quality of $\mathbf{C}^{bkg}$ and $\mathbf{Y}$ based on semantic features extracted via the pre-trained ResNet50 model (He et al. 2016). The calculation formulation of this loss is as follows,

$$L_{per} = \sum_{m=1}^{M} (||\texttt{ResNet}^{(m)}(\mathbf{Y}) - \texttt{ResNet}^{(m)}(\mathbf{G}^{wf})||_2 + ||\texttt{ResNet}^{(m)}(\mathbf{C}^{bkg}) - \texttt{ResNet}^{(m)}(\mathbf{G}^{bkg})||_2), \quad (7)$$

where M denotes the number of feature maps used for calculation, and $\texttt{ResNet}^{(m)}(\cdot)$ produces the m-th feature map.

Finally, the patch-wise adversarial training loss (Isola et al. 2017) is applied for improving the visual reality of $\mathbf{Y}$. Suppose the patch-wise discriminator model be $\mathcal{D}(\cdot)$. The training loss for the discriminator is defined as follows:

$$L_D = -\Gamma(\log(\mathcal{D}(\mathbf{G}^{wf}))) - \Gamma(\log(\mathcal{D}(\mathbf{Y}) \circ (1 - \mathbf{M}))) - \Gamma(\log(1 - \mathcal{D}(\mathbf{Y})) \circ \mathbf{M}), \quad (8)$$

where $\Gamma(\cdot)$ denotes the element summation operation. The adversarial regularization for the watermark removal model is formulated as follows,

$$L_G = -\Gamma(\log(\mathcal{D}(\mathbf{Y})) \circ \mathbf{M}). \quad (9)$$

To avoid the gradient fluctuation brought by the adversarial training, we introduce the gradient penalty $P = ||\nabla_{\theta_G} L_G||_2^2$ where θ_G represents the parameters of the visible watermark removal model. Besides, an additional perceptual loss L'_{per} is calculated using features extracted by the discriminator.

The total loss for training the watermark removal model is as follows:

$$L = \omega_1 L_{pixel} + \omega_2 L_{per} + \omega_3 L_G + \omega_4 L'_{per} + \omega_5 P, \quad (10)$$

where ω_1, ω_2, ω_3, ω_4, and ω_5 are weighting factors for the above loss terms.

Experiments

Datasets

• **ILAW.** Real-world images often feature large-area watermarks and undergo complex distortions such as compression or resampling, leading to the degradation of both watermark and background details. To advance research in large-area watermark removal, we introduce the *Images with Large-Area Watermarks* (ILAW) dataset. The training set includes 60,000 images of size 256×256 with 1,087 different watermarks, while the validation set contains 10,000 images of size 512×512 with 160 distinct watermarks, different from those in the training set. Background images are sourced from the Places365 Challenge dataset (Zhou et al. 2017), and watermarks are collected from the Internet. Given a clean background image $\mathbf{I}$ and a watermark image $\mathbf{W}$, we composite them into a watermarked image $\mathbf{X}$,

$$\mathbf{X} = \mathcal{T}((1 - \mathbf{A}) \circ \mathbf{I} + \mathbf{A} \circ \mathbf{W}), \quad (11)$$

where $\mathbf{A} \in [0, 1]^{h \times w \times 1}$ denotes alpha channel, and $\mathcal{T}(\cdot)$ represents random image distortion function consisting of image compression and resampling operations. The ground-truths of watermark-free background image and watermark-excluded background component image are obtained via $\mathbf{G}^{wf} = \mathcal{T}(\mathbf{I})$ and $\mathbf{G}^{bkg} = \mathcal{T}((1 - \mathbf{A}) \circ \mathbf{I})$. For synthesizing the coarse watermark mask, we first generate a precise mask $\mathbf{M}_0$ from $\mathbf{A}$, i.e., $\mathbf{M}_0 = \mathbf{A} > 0$. Then, the same resampling operations used for generating $\mathbf{X}$ and a random dilation operation are adopted to process $\mathbf{M}_0$, resulting in $\mathbf{M}$. Examples of our dataset are displayed in Figure 6, where the watermark area are much larger and more opaque than pictures in popular datasets used in watermark removal works.

• **Real-world Dataset.** We collect 27 high-resolution watermarked images from the Internet, and then employ LabelMe (Russell et al. 2008) to draw coarse masks. User study is conducted for assessment on this dataset.

Implementation Details & Evaluation Metrics

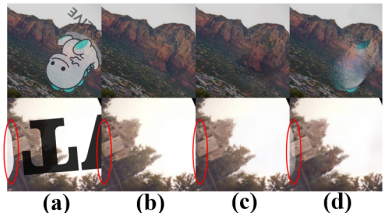
The codes are implemented by PyTorch (Paszke et al. 2019), PyTorch-Lightning (Falcon 2019) and Hydra (Yadan 2019). We employ Adam optimizer (Kingma and Ba 2014) with a learning rate of 0.0001 to train both generator and discriminator. The model is trained for 100 epochs with a batch size of 16. For the weights for individual sub-losses, we experiment with variation on weight factors, and observe subtle performance fluctuation. Finally, we set: $\omega_1 = 10, \omega_2 = 30, \omega_3 = 1, \omega_4 = 100, \omega_5 = 0.001$. For comparison, we train other models on ILAW with an additional input of coarse mask $\mathbf{M}$ concatenated to the original input.

To evaluate the removal efficacy, we use PSNR, SSIM, RMSE, RMSE_w and LPIPS (Zhang et al. 2018). The RMSE_w is RMSE averaged inside mask. LPIPS evaluates perceptually similarities in large-area content recovery.

Figure 6: Watermarked images from our constructed dataset.

Methods	PSNR↑	SSIM↑	RMSE↓	RMSE_w ↓	LPIPS↓
Performance with fixed mask per image					
WDNet	23.86	0.887	19.98	22.99	0.170
SplitNet	25.90	0.901	15.99	19.15	0.147
SLBR	26.13	0.908	15.16	18.07	0.139
LaMa	17.97	0.677	37.33	44.92	0.326
MAT	12.24	0.615	72.27	94.98	0.321
DENet	24.74	0.894	16.93	20.25	0.171
CoordFill	22.66	0.819	22.69	37.64	0.149
SCATCL	16.53	0.605	42.54	56.12	0.394
Ours # 1	26.38	0.922	15.34	18.30	0.097
Ours # 2	25.99	0.920	15.94	19.02	0.103
Ours	**26.81**	**0.924**	**15.11**	**18.01**	**0.094**
Performance with fixed and coarser mask per image					
WDNet	24.37	0.887	18.83	20.45	0.166
SplitNet	25.53	0.899	18.11	19.02	0.143
SLBR	26.09	0.907	15.16	15.91	0.141
LaMa	14.87	0.551	52.20	56.50	0.470
MAT	8.31	0.483	108.30	123.88	0.386
DENet	25.06	0.900	16.31	17.14	0.162
CoordFill	17.07	0.568	40.52	46.97	0.381
SCATCL	13.89	0.487	63.25	55.00	0.536
Ours	**26.66**	**0.924**	**15.09**	**16.48**	**0.094**

Table 1: Experimental results of different models on ILAW. # 1: our method using unaugmented masks during training. # 2: our method without using pretrained LaMa.



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

Figure 7: Comparisons of model without pre-training/model with pre-training. (a) Input, (b) Target, (c) Our output, (d) Output without pre-trained LaMa.

Comparison with Existing Methods

Visible Watermark Removal Guided by Coarse Mask First, we conduct experiments on the ILAW dataset using coarse masks, evaluating methods such as WDNet (Liu, Zhu, and Bai 2021), SplitNet (Cun and Pun 2021), SLBR (Liang et al. 2021), DENet (Sun, Su, and Wu 2023), LaMa (Suvorov et al. 2022), MAT (Li et al. 2022), CoordFill (Liu et al. 2023), and SCATCL (Zuo et al. 2023). We also test our approach in three scenarios: training with unaugmented masks (Ours # 1), training without pre-training LaMa (Ours # 2), and training with the original design (Ours). The results in the upper section of Table 1 show our method's superior performance, driven by the substantial prompt information extracted by WCC and BCE, which helps LaMa accurately recover background content. Ours # 1 highlights the benefits of coarse masks for restoration, while Ours # 2 confirms that pre-trained LaMa parameters assist in content generation. Figure 7 further illustrates that models without pre-training leave residual watermarks in the restored images.

Count	SplitNet	SLBR	**Ours**
Vote	154	206	**882**
Best-Restored	1	2	**24**

Table 2: User study on the real-world dataset.

Moreover, in the lower section of Table 1, we record some testing results using coarser masks generated from augmentation with more dilation and erosion, or rough outline with polygonal shape. The reuslts demonstrates our method's robustness to low-quality masks, maintaining superior performance compared to other models.

For real-world validation, we collect restoration results from the top 3 models (our model, SplitNet, SLBR) and conduct a survey with 46 participants. Each participant votes on the most effective model for each of 27 images. Table 2 shows the total votes and the number of images where each model was preferred. The results indicate a clear consensus of our model's superiority over the alternatives.

Figures 8 and 9 showcase visual comparisons between our method and existing approaches. It is evident that watermark removal methods like WDNet, SplitNet, SLBR, and DENet fail to completely remove watermarks, and LaMa's inpainting produces inaccurate content within masked areas. In contrast, our model achieves superior restoration.

	WDNet	SplitNet	SLBR	DENet	**Ours**
PSNR↑	24.37	25.72	25.02	19.66	**25.77**
SSIM↑	0.887	0.892	0.890	0.814	**0.916**
LPIPS↓	0.166	0.156	0.154	0.236	**0.100**

Table 3: Performance of methods using none/white masks.

Blind Watermark Removal We also test some methods on blind watermark removal, where coarse mask inputs are absent or replaced by white masks. As shown in Table 3, our method outperforms competitors, demonstrating its effective watermark locating even without assistance of mask input.

Ablation Studies

Effect of Key Structures To demonstrate the effectiveness of our inpainting model with dual-branch feature adaptation, we conduct ablation experiments with different branch configurations. First, we train with only LaMa, as shown in the first row of Table 4, highlighting the need for feature adaptation, as the model alone struggles to learn features in masked regions, limiting background recovery. Next, we test a single-branch adaptation (second row of Table 4), which performs suboptimally due to difficulty distinguishing low-opacity watermarks from the background. In contrast, dual-branch adaptation successfully separates the watermark, allowing for better restoration. We then use two branches with transposed attention modules (third row of Table 4), emphasizing the importance of LaMa's long-range feature capture for large-area restoration. Finally, we test on feature adapting network with different number of blocks, and found that out 3-block structure reaches the best results.

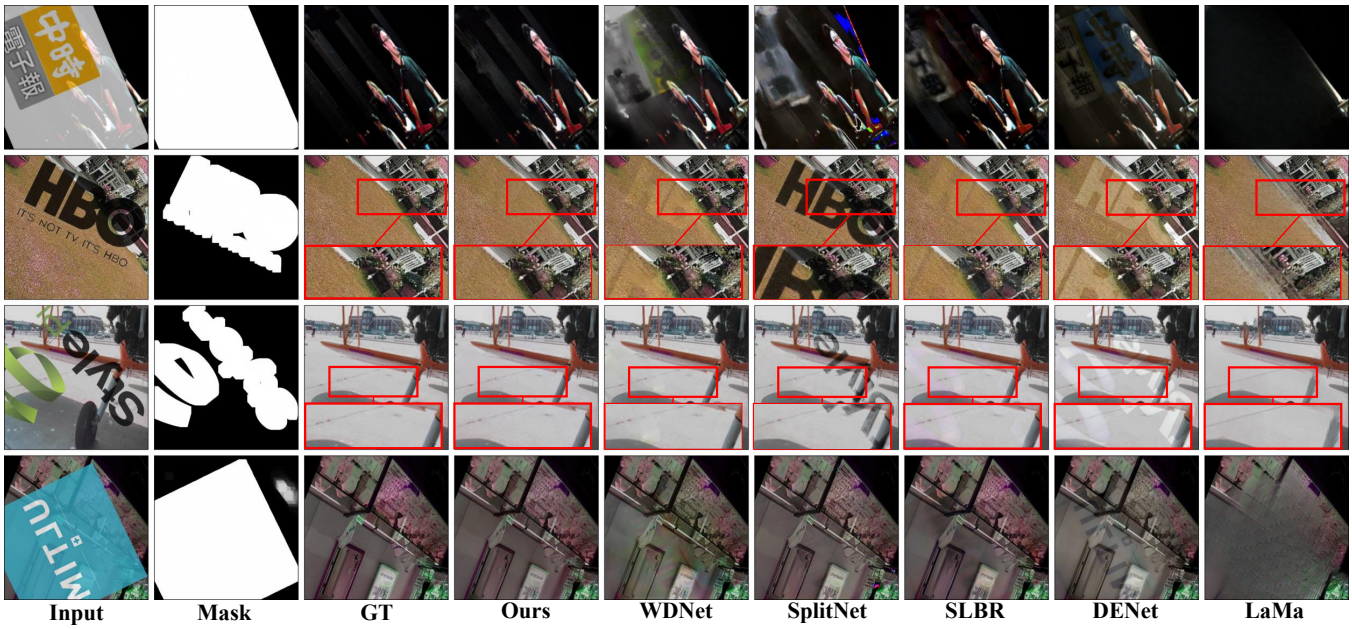


Figure 8: Visualization results of different methods on ILAW.

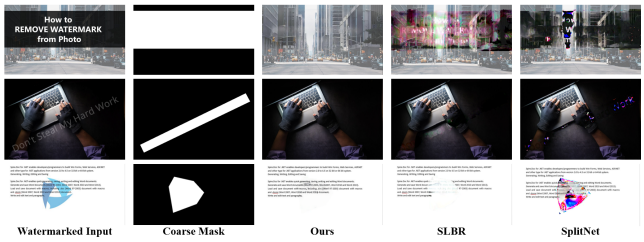


Figure 9: Visualization results of different methods on pictures from real-world dataset.

Backbone	Feature Adapting	PSNR	SSIM
LaMa	-	23.62	0.895
LaMa	3 TA blocks in BCE	25.67	0.916
WCC	3 TA blocks in BCE	25.82	0.917
LaMa	2 TA blocks in WCC+BCE	23.89	0.892
LaMa	6 TAblocks in WCC+BCE	22.82	0.874
LaMa	**3 TA blocks in WCC+BCE**	**26.81**	**0.924**

Table 4: Ablation study of network structure.

Effect of Key Modules To validate the effectiveness of the feature extraction and fusion modules in our design, we replaced the transposed attention and GFM modules with alternatives. The results in Table 5 show that our chosen modules are more effective, as they better capture global background and watermark features, while the GFM's gated mechanism efficiently filters useful information for restoration. The alternative modules perform suboptimally, as handling high-resolution images with large watermarks requires addressing both long-range dependencies between watermarked and unwatermarked areas and short-range dependencies within the watermarked area. Simple local or global approaches, such as convolution or dilated convolution, are insufficient. Conventional attention modules lead to excessive and meaningless calculations, resulting in minor performance and longer inference time.

Feature Extraction	Fusion	PSNR	SSIM
Conv (kernel = 3)	GFM	24.04	0.895
Conv (kernel = 7)	GFM	20.87	0.855
Conv (kernel = 5, dilation = 3)	GFM	22.25	0.864
Conventional Attention	GFM	23.50	0.887
Transposed Attention	Conv	26.02	0.920
Transposed Attention	**GFM**	**26.81**	**0.924**

Table 5: Ablation study of module selection.

Conclusion

In conclusion, this paper introduces an innovative feature adapting framework tailored for the challenging task of large-area visible watermark removal. The proposed framework leverages specialized components, including a watermark component cleaning branch and a background content embedding branch, both equipped with transposed attention modules for enhanced feature extraction. The integration of gated fusion modules further refines the image inpainting backbone, facilitating accurate reconstruction of watermarked regions by incorporating prompt information within the extracted features. Additionally, the model exhibits adaptability to imprecise watermark masks through the incorporation of a coarse segmentation mask. Empirical evaluations conducted on two datasets demonstrate the effectiveness of our method, showcasing its state-of-the-art performance in comparison to various existing approaches.

Acknowledgments

This work was supported by the National Natural Science Foundation of China under Grant 62376206 and 62322608.

References

Bertalmio, M.; Sapiro, G.; Caselles, V.; and Ballester, C. 2000. Image inpainting. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, 417–424.

Cheng, D.; Li, X.; Li, W.-H.; Lu, C.; Li, F.; Zhao, H.; and Zheng, W.-S. 2018. Large-scale visible watermark detection and removal with deep convolutional networks. In *Pattern Recognition and Computer Vision: First Chinese Conference, PRCV 2018, Guangzhou, China, November 23-26, 2018, Proceedings, Part III 1*, 27–40. Springer.

Chi, L.; Jiang, B.; and Mu, Y. 2020. Fast fourier convolution. *Advances in Neural Information Processing Systems*, 33: 4479–4488.

Chollet, F. 2017. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1251–1258.

Cun, X.; and Pun, C.-M. 2021. Split then refine: stacked attention-guided ResUNets for blind single image visible watermark removal. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 1184–1192.

Dong, Q.; Cao, C.; and Fu, Y. 2022. Incremental transformer structure enhanced image inpainting with masking positional encoding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11358–11368.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Falcon, W. A. 2019. Pytorch lightning. *GitHub*, 3.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Hendrycks, D.; and Gimpel, K. 2016. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*.

Hertz, A.; Fogel, S.; Hanocka, R.; Giryes, R.; and Cohen-Or, D. 2019. Blind visual motif removal from a single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6858–6867.

Isola, P.; Zhu, J.-Y.; Zhou, T.; and Efros, A. A. 2017. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1125–1134.

Jin, X.; Han, L.-H.; Li, Z.; Guo, C.-L.; Chai, Z.; and Li, C. 2023. DNF: Decouple and Feedback Network for Seeing in the Dark. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 18135–18144.

Johnson, J.; Alahi, A.; and Fei-Fei, L. 2016. Perceptual losses for real-time style transfer and super-resolution. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14*, 694–711. Springer.

Kingma, D. P.; and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Li, W.; Lin, Z.; Zhou, K.; Qi, L.; Wang, Y.; and Jia, J. 2022. Mat: Mask-aware transformer for large hole image inpainting. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 10758–10768.

Liang, J.; Niu, L.; Guo, F.; Long, T.; and Zhang, L. 2021. Visible watermark removal via self-calibrated localization and background refinement. In *Proceedings of the 29th ACM International Conference on Multimedia*, 4426–4434.

Liu, G.; Reda, F. A.; Shih, K. J.; Wang, T.-C.; Tao, A.; and Catanzaro, B. 2018. Image inpainting for irregular holes using partial convolutions. In *Proceedings of the European conference on computer vision (ECCV)*, 85–100.

Liu, W.; Cun, X.; Pun, C.-M.; Xia, M.; Zhang, Y.; and Wang, J. 2023. CoordFill: Efficient High-Resolution Image Inpainting via Parameterized Coordinate Querying. *arXiv preprint arXiv:2303.08524*.

Liu, Y.; Zhu, Z.; and Bai, X. 2021. Wdnet: Watermark-decomposition network for visible watermark removal. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 3685–3693.

Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32.

Redmon, J.; and Farhadi, A. 2017. YOLO9000: better, faster, stronger. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 7263–7271.

Ronneberger, O.; Fischer, P.; and Brox, T. 2015. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, 234–241. Springer.

Russell, B. C.; Torralba, A.; Murphy, K. P.; and Freeman, W. T. 2008. LabelMe: a database and web-based tool for image annotation. *International journal of computer vision*, 77: 157–173.

Sun, R.; Su, Y.; and Wu, Q. 2023. DENet: Disentangled Embedding Network for Visible Watermark Removal. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 2411–2419.

Suvorov, R.; Logacheva, E.; Mashikhin, A.; Remizova, A.; Ashukha, A.; Silvestrov, A.; Kong, N.; Goka, H.; Park, K.; and Lempitsky, V. 2022. Resolution-robust large mask inpainting with fourier convolutions. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, 2149–2159.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Yadan, O. 2019. Hydra-a framework for elegantly configuring complex applications. *Github*, 2: 5.

Yang, S.; Chen, X.; and Liao, J. 2023. Uni-paint: A Unified Framework for Multimodal Image Inpainting with Pretrained Diffusion Model. *arXiv preprint arXiv:2310.07222*.

Zamir, S. W.; Arora, A.; Khan, S.; Hayat, M.; Khan, F. S.; and Yang, M.-H. 2022. Restormer: Efficient transformer for high-resolution image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5728–5739.

Zhang, R.; Isola, P.; Efros, A. A.; Shechtman, E.; and Wang, O. 2018. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 586–595.

Zhou, B.; Lapedriza, A.; Khosla, A.; Oliva, A.; and Torralba, A. 2017. Places: A 10 million image database for scene recognition. *IEEE transactions on pattern analysis and machine intelligence*, 40(6): 1452–1464.

Zuo, Z.; Zhao, L.; Li, A.; Wang, Z.; Zhang, Z.; Chen, J.; Xing, W.; and Lu, D. 2023. Generative Image Inpainting with Segmentation Confusion Adversarial Training and Contrastive Learning. *arXiv preprint arXiv:2303.13133*.