This ICCV paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Fine-grained Visible Watermark Removal

Li Niu*, Xing Zhao, Bo Zhang, Liqing Zhang
Department of Computer Science and Engineering, MoE Key Lab of Artificial Intelligence,
Shanghai Jiao Tong University
{ustcnewly,1033874657,bo-zhang}@sjtu.edu.cn, zhang-lq@cs.sjtu.edu.cn

Abstract

Visible watermark removal aims to erase the watermark from watermarked image and recover the background image, which is a challenging task due to the diverse watermarks. Previous works have designed dynamic network to handle various types of watermarks adaptively, but they ignore that even the watermarked region in a single image can be divided into multiple local parts with distinct visual appearances. In this work, we advance image-specific dynamic network towards part-specific dynamic network, which discovers multiple local parts within the watermarked region and handle them adaptively. Specifically, we propose a query-based multi-task framework, in which part query embeddings are jointly used in two branches to predict part masks and restore watermarked parts. Extensive experiments demonstrate the effectiveness of our fine-grained watermark removal network.

1. Introduction

Overlaying visible watermark on a digital image has been a prevalent way to claim copyright and declare ownership. As its reverse process, visual watermark removal [33, 56] has attracted more and more research interest, which can verify and promote the resilience of overlaid watermarks. Specifically, visible watermark removal aims to erase the watermark and reconstruct the watermark-free (background) image. One major challenge of watermark removal task is that the watermarks are very diversified, in terms of different locations, patterns, transparencies, and so on. Recent works [33, 56] realize this issue and develop dynamic networks to cope with diversified watermarks adaptively. For example, [33] extracts the feature of watermarked region and appends it to original feature map as extra guidance. [56] also extracts the feature of watermarked region, but uses it to generate dynamic convolution kernel to manipulate the original feature map. Although [33, 56] have achieved significant improvement over previous methods by adaptively coping with various types of watermarks, they still treat the watermarked region in an image as a whole, disregarding the fact that even the watermarked region in a single image could be divided into multiple parts with vastly distinct appearances. For example, the watermarked region in Figure 1(a) consists of one pattern and two lines of texts, and the watermarked region in Figure 1(b) consists of one symbol, one line of text, and extra curves. Moreover, when a watermark is overlaid on a background image with certain transparency, one part (*e.g.*, one line of text) in the watermarked region could also exhibit diverse colors due to the colorful background, which makes the appearance of watermarked region even more complicated.

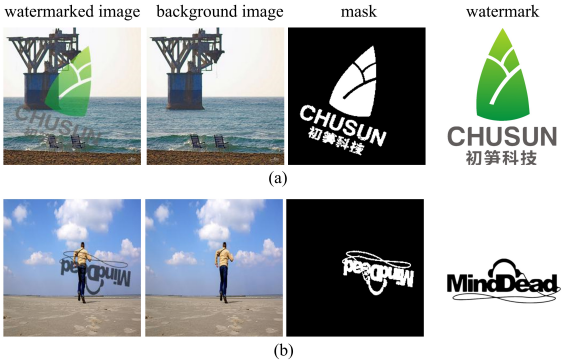

Figure 1. Two examples of watermarked images associated with their background images, watermark masks, and watermarks. Visible watermark removal aims to erase the watermark and reconstruct the background image.

In this work, inheriting the spirit of previous works [33, 56], we take a further step along the path of dynamic network. Specifically, we not only cope with the watermarked regions in different images adaptively, but also cope with different watermarked parts in one image adaptively, which advances image-specific dynamic network towards part-specific dynamic network. We adopt the multi-task framework used in previous works [33, 56], which contains one encoder and two encoders. The mask decoder accounts

*Corresponding author.

for localizing the watermark and the background decoder accounts for recovering the watermarked region. Following previous works [33, 56], we also add skip connections [42] linking one encoder with two decoders. To cope with different watermarked parts adaptively, we need to localize them in the mask decoder and repair them in the background decoder. Inspired by query-based network structures in previous works [3, 6], we introduce query embeddings into our multi-task framework, which establishes a connection between mask decoder and background decoder.

In particular, assuming that there are in total K fine-grained categories of watermarked parts, K part query embeddings and one watermark-free query embedding interact with the encoder feature map through a transformer [45] block to produce adapted query embeddings. $K+1$ adapted query embeddings are expected to contain the part-relevant information and watermark-free information of one specific image. Then, we apply $K+1$ adapted query embeddings to mask decoder and background decoder. In the mask decoder, similar to [6], we calculate the similarity between $K+1$ adapted query embeddings with the last feature map in the mask decoder to generate K part masks corresponding to K part categories and one watermark-free mask. Different from [6], we do not have ground-truth masks for K part categories. Hence, we enforce the union of K predicted part masks to approach the ground-truth watermark mask and employ low-entropy loss [17] to push apart K predicted part masks. In the background decoder, we use K adapted part query embeddings to produce K modulated convolution kernels for K part categories, which act upon K local parts specified by K predicted part masks. In this way, we can restore different watermarked parts adaptively and achieve better restoration effect.

We conduct experiments on two benchmark watermark removal datasets: CLWD [38] and LOGO30K [8], to verify the effectiveness of our proposed method. Our contributions can be summarized as follows. 1) We are the first to consider fine-grained part categories in the visible watermark removal task, which advances image-specific dynamic network towards part-specific dynamic network; 2) We design a query-based multi-task framework, in which the adapted query embeddings are jointly used to localize different part categories in the mask decoder and restore the corresponding local parts in the background decoder; 3) Extensive experiments on two benchmark datasets demonstrate the effectiveness of our proposed network.

2. Related Work

2.1. Visible Watermark Removal

Visual watermarking prevails in protecting the digital image copyright, by superimposing a watermark on the background image. As its adversarial task, visible watermark removal can enhance the resilience of watermark. Early methods [31, 2] proposed GAN [16]-based network structures for watermark removal. [38, 23] formulated watermark removal under a multi-task framework, which predicts watermark-free image, watermark mask, and watermark pattern collaboratively. [8] designed a two-stage multi-task framework with attention mechanism and refinement stage. [33] designed self-calibrated mask refinement for adaptive mask prediction and mask-guided background enhancement. [56] used semantic similarity to propagate useful information and applied dynamic convolution to handle diverse watermarks. [14] designed a novel architecture for better information extraction in the long range. [37] proposed watermark vaccine, which adds invisible perturbation to the watermarked images to attack against the existing watermark removal methods.

However, the above methods are still struggling to tackle diversified and complicated watermarked regions. Moreover, they ignore that even the watermarked region in a single watermarked image could be decomposed into multiple parts. In this work, we consider fine-grained part categories by localizing and restoring different local parts adaptively.

2.2. Image Content Removal

Image content removal or image restoration includes myriads of tasks such as image deraining [46, 51], image dehazing [7, 12, 55], shadow removal [9, 11], and so on. Besides the research works in each field, there are also some general methods [49, 20, 54, 53, 57, 40, 32] for universal image content removal or image restoration. To name a few, [49] proposed a transformer block with locally enhanced window and a learnable multi-scale restoration modulator. In [20], they proposed a general blind image decomposition network. In [54], they proposed a multi-stage progressive image restoration architecture. In visible watermark removal task, the removal target is watermark with diverse colors, patterns, locations, transparencies, *etc*, making it a challenging and unique task.

2.3. Vision Transformer

Transformer has been applied to a wide range of computer vision tasks like detection [3, 58], segmentation [48, 47, 6], and pose estimation [24, 35]. Transformer has also been used in some low-level computer vision tasks like image super-resolution [50, 34, 52, 1], image denoising and image deraining [4, 49], image harmonization [18], style transfer [10]. Our mask decoder is similar to [6], but we do not have ground-truth part masks. More importantly, our network is query-based multi-task framework, in which the adapted query embeddings are jointly used in the mask decoder and the background decoder.

2.4. Dynamic Network

Dynamic networks have been widely used in deep learning image processing tasks, which has been discussed and categorized in [21]. According to [21], dynamic parameters can be classified into parameter adjustment [22, 15], weight prediction [43, 25, 39, 19], and dynamic features [30, 5]. Our model uses adapted query embeddings to control the kernel weights, which belongs to weight prediction. This is the first work to handle different watermarked parts dynamically.

3. Our Method

3.1. Overall Network Structure

When superimposing one watermark on a watermark-free (background) image $\boldsymbol{I}^{bg}$, we can obtain the watermarked image $\boldsymbol{I}^{wm}$ and its corresponding watermark mask $\boldsymbol{M}$. Visible watermark removal task aims to localize and recover the watermarked region in $\boldsymbol{I}^{wm}$, producing the watermark-free image $\boldsymbol{I}^{bg}$. Since visible watermark removal task needs to handle two tasks simultaneously: watermark mask prediction and background image restoration, which falls into the realm of multi-task learning. Thus, previous works [33, 56] usually adopt the multi-task framework with one encoder shared by different tasks and multiple decoders accounting for different tasks. In this work, we also employ one encoder E and two decoders $\{D^{ms}, D^{bg}\}$, in which D^{ms} is the mask decoder used to predict the watermark mask $\boldsymbol{M}$ and D^{bg} is the background decoder used to recover the background image $\boldsymbol{I}^{bg}$. The encoder E consists of five encoder blocks. The decoder D^{bg}/D^{ms} consists of four decoder blocks, in which the first decoder block is shared by two decoders. The structures of encoder and decoder blocks are inherited from [23, 33]. Following previous works [33, 56], we also add skip connections [42] to link one encoder with two decoders.

To cope with diversified watermarked parts, we learn part query embeddings corresponding to different part categories, which are jointly used by mask decoder and background decoder. For the mask decoder, we calculate the similarity between query embeddings and mask decoder feature map, yielding part masks for different part categories. For the background decoder, the query embeddings are used to modulate convolution weights for different part categories, which act upon the background decoder feature map within the corresponding local parts specified by part masks. Next, we will introduce the details of part query embeddings in Section 3.2 and describe their usage in the mask (*resp.*, background) decoder in Section 3.3 (*resp.*, 3.4).

3.2. Part Query Embeddings

Assuming that the local parts of watermarked regions can be classified into K part categories based on distinct visual appearances, we use K part query embeddings $\{\boldsymbol{q}_k|_{k=1}^{K}\}$ to represent the prior knowledge (*e.g.*, color, scale, pattern) of K part categories. Additionally, we introduce a watermark-free query embedding $\boldsymbol{q}_{K+1}$ for the watermark-free category. These $K+1$ query embeddings can be used to predict the watermark mask and restore the watermarked region, establishing the connection between the mask decoder and the background decoder.

Even for the same part category, the local parts in different images are likely to have different visual appearances. Similarly, the watermark-free regions in different images also vary dramatically. To better cope with diverse images, we further adapt query embeddings to each specific image. Specifically, we use encoder E to extract the feature map $\boldsymbol{F}^{en}$ from watermarked image $\boldsymbol{I}^{wm}$. Each query embedding $\boldsymbol{q}_k$ interacts with the encoder feature map $\boldsymbol{F}^{en}$ through a transformer [45] block T, during which $\boldsymbol{q}_k$ serves as key and the pixel-wise feature vectors in $\boldsymbol{F}^{en}$ serve as keys/values, producing an adapted query embedding $\boldsymbol{q}'_k$. The adapted query embeddings are supposed to incorporate the relevant information from $\boldsymbol{I}^{wm}$ and become well-tailored to this image. Next, we will introduce how the adapted query embeddings $\{\boldsymbol{q}'_k|_{k=1}^{K+1}\}$ are used to predict fine-grained watermark part masks and restore local parts accordingly.

3.3. Part Mask Prediction

The mask decoder D^{ms} targets at predicting the masks for different part categories and watermark-free category. We denote the last feature map in D^{ms} as $\boldsymbol{F}^{ms}$. Analogous to [6], we obtain the masks for different part categories and watermark-free category by calculating the similarity between each adapted query embedding and pixel-wise feature vectors in $\boldsymbol{F}^{ms}$. Specifically, for the k-th adapted query embedding $\boldsymbol{q}'_k$, we project it to the same space as $\boldsymbol{F}^{ms}$ using a fully-connected (FC) layer and calculate its dot product with all pixel-wise feature vectors in $\boldsymbol{F}^{ms}$. We perform softmax normalization on the dot product results over $K+1$ query embeddings, leading to K part masks $\{\hat{\boldsymbol{M}}_k^{pt}|_{k=1}^{K}\}$ and one watermark-free mask $\hat{\boldsymbol{M}}_{K+1}^{pt}$. Note that we only have the ground-truth mask $\boldsymbol{M}$ for the entire watermark region, and there is no ground-truth supervision for each part mask. Thus, we calculate the sum of K part masks as $\hat{\boldsymbol{M}}^{wm} = \sum_{k=1}^{K} \hat{\boldsymbol{M}}_k^{pt}$, which is equivalent to $1 - \hat{\boldsymbol{M}}_{K+1}^{pt}$. $\hat{\boldsymbol{M}}^{wm}$ is enforced to be close to $\boldsymbol{M}$ using binary cross-entropy loss:

$$\begin{aligned}\mathcal{L}_{ms} = & -\sum_{i,j}[M(i,j)\log \hat{M}^{wm}(i,j) \\ & +(1-M(i,j))\log(1-\hat{M}^{wm}(i,j))],\end{aligned} \tag{1}$$

in which $M(i,j)$ (*resp.*, $\hat{M}^{wm}(i,j)$) is the (i,j)-th entry in $\boldsymbol{M}$ (*resp.*, $\hat{\boldsymbol{M}}^{wm}$).

When only using $\mathcal{L}_{ms}$, we observe that the predicted

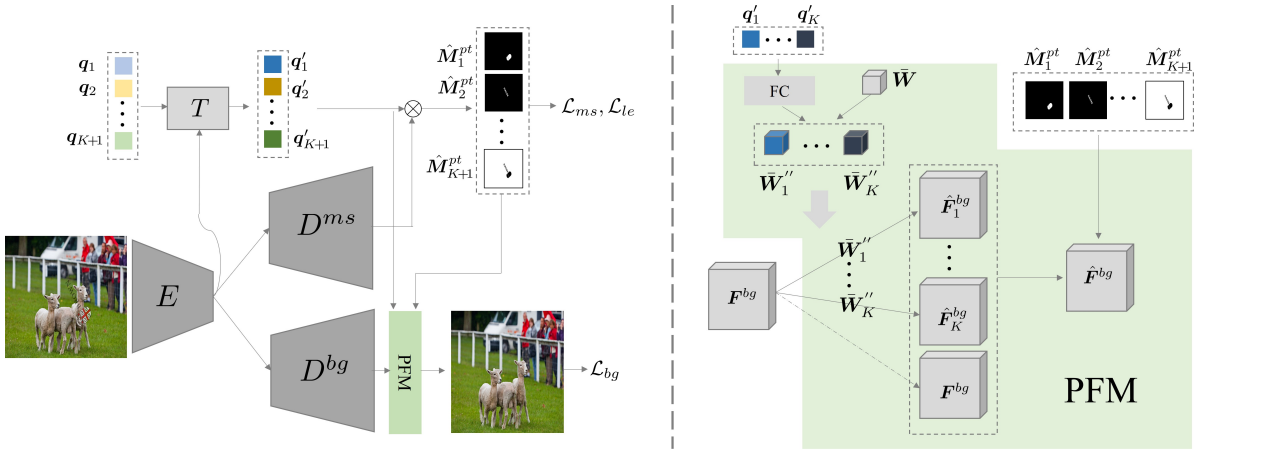

Figure 2. Our network consists of one encoder E and two decoders $\{D^{ms}, D^{bg}\}$. We omit the skip connections in this figure for brevity. The transformer block T outputs the adapted query embeddings, which interact with mask decoder feature to produce part masks and interact with background decoder feature through our PFM block to restore the background image. The detailed architecture of PFM block is shown in the right subfigure.

K part masks are all close to each other, failing to distinguish different parts. To address this issue, we additionally employ low-entropy loss [17] for the predicted part masks, which ensures that each watermarked pixel is classified into certain part categories instead of uniformly distributed over all part categories. Formally, we apply low-entropy loss to the watermarked pixels:

$$\mathcal{L}_{le} = -\sum_{i,j} M(i,j) \sum_{k=1}^{K+1} \hat{M}^{pt}_k(i,j) \log \hat{M}^{pt}_k(i,j), \qquad (2)$$

in which $M(i,j) = 1$ (*resp.*, $M(i,j) = 0$) indicates the watermarked (*resp.*, watermark-free) pixels and $\hat{M}^{pt}_k(i,j)$ is the (i,j)-th entry in $\hat{\boldsymbol{M}}^{pt}_k$. Note that lower entropy implies higher concentration. In addition, the probabilities that watermarked pixels belong to watermark-free category have been suppressed to zero based on Eqn. 1. Therefore, the probabilities that watermarked pixels belong to different part categories are pushed to be concentrated based on Eqn. 2.

Although the procedure of our query-based mask prediction resembles that in [6] to some extent, our method has several major differences: 1) [6] has ground-truth masks, which are inaccessible in our task. Hence, we design $\mathcal{L}_{ms}$ and $\mathcal{L}_{le}$ to supervise the predicted masks; 2) Our predicted part masks and watermark-free mask are also used in the background decoder to help restore the background, rendering a novel query-based multi-task framework in which two tasks jointly use the adapted query embeddings.

3.4. Part-aware Background Restoration

The background decoder D^{bg} targets at predicting the background image. Because different part categories have quite distinct visual appearances, we conjecture that restoring different local parts in the watermarked region adaptively could achieve better restoration effect. Therefore, we design a novel Part-aware Feature Modulation (PFM) block, which utilizes the adapted query embeddings and the masks predicted by mask decoder to restore different local parts adaptively. The adapted query embeddings and predicted masks solve the problem of *how* and *where* to restore, respectively.

We denote the last feature map in D^{bg} as $\boldsymbol{F}^{bg}$. We expect to transform the feature map in the watermarked region while keeping the watermark-free region unchanged, since only the watermarked region needs to be restored. To transform different local parts in the watermarked region adaptively, we opt for the dynamic feature transformation technique proposed in [28], which dynamically modulates convolution weights. In our work, we use K adapted part query embeddings to produce K modulated convolution weights, which are applied to $\boldsymbol{F}^{bg}$ separately to acquire K transformed feature maps $\{\hat{\boldsymbol{F}}^{bg}_k|^K_{k=1}\}$.

Specifically, we have learnable base convolution weights $\bar{\boldsymbol{W}}$, in which $\bar{W}(m,n,l)$ is the weight for the m-th input channel, n-th output channel, and the l-th spatial location. Then, we pass each adapted query embedding $\boldsymbol{q}'_k$ through one FC layer to predict the scale vector $\boldsymbol{s}_k$ corresponding to input channels, which are used to modulate the base convolution weights:

$$\bar{W}'_k(m,n,l) = \bar{W}(m,n,l) \cdot s_k(m), \qquad (3)$$

where $s_k(m)$ is the m-th entry in $\boldsymbol{s}_k$ (scale for the m-th input channel) and $\bar{W}'_k(m,n,l)$ is the modulated weight.

Next, we normalize the modulated weights following [28]:

$$\bar{W}_k''(m,n,l) = \frac{\bar{W}_k'(m,n,l)}{\sqrt{\sum_{m,l} \bar{W}_k'(m,n,l)^2 + \epsilon}}, \quad (4)$$

in which ϵ is a small constant to avoid numerical error. For more details of convolution weights modulation, please refer to [28]. We apply the modulated convolution weights $\bar{\boldsymbol{W}}_k''$ to $\boldsymbol{F}^{bg}$ and produce the transformed feature map $\hat{\boldsymbol{F}}_k^{bg}$. Then, we combine K transformed feature maps and the original feature map based on the predicted K part masks and one watermark-free mask, giving rise to the final feature map $\hat{\boldsymbol{F}}^{bg}$:

$$\hat{\boldsymbol{F}}^{bg} = \sum_{k=1}^{K} \hat{\boldsymbol{M}}_k^{pt} \circ \hat{\boldsymbol{F}}_k^{bg} + \hat{\boldsymbol{M}}_{K+1}^{pt} \circ \boldsymbol{F}^{bg}, \quad (5)$$

in which $\circ$ means element-wise product. Although similar dynamic kernel has been used in [56], our method is vastly different from theirs in the following aspects: 1) [56] uses the pooled watermark feature to predict the entire convolution kernel, while we use adapted query embeddings to modulate the base convolution weights; 2) [56] only predicts a single convolution kernel for the entire watermarked region, whereas we use K modulated convolution weights to cope with K part categories adaptively.

The final feature map $\hat{\boldsymbol{F}}^{bg}$ is used to predict the background image $\hat{\boldsymbol{I}}^{bg}$. We combine $\hat{\boldsymbol{I}}^{bg}$ with the input watermarked image $\boldsymbol{I}^{wm}$ using the predicted watermark mask $\hat{\boldsymbol{M}}^{wm}$, yielding the final output $\tilde{\boldsymbol{I}}^{bg} = \hat{\boldsymbol{M}}^{wm} \circ \hat{\boldsymbol{I}}^{bg} + (1 - \hat{\boldsymbol{M}}^{wm}) \circ \boldsymbol{I}^{wm}$. Following [33, 56], we regularize the final output with $\mathcal{L}_1$ loss and perceptual loss [26]:

$$\mathcal{L}_{bg} = \|\tilde{\boldsymbol{I}}^{bg} - \boldsymbol{I}^{bg}\|_1 + \sum_{d=1}^{3} \|\Phi_d(\tilde{\boldsymbol{I}}^{bg}) - \Phi_d(\boldsymbol{I}^{bg})\|_1, \quad (6)$$

where $\Phi_d(\cdot)$ represents the activation map of d-th layer in VGG16 [44].

So far, all the losses can be summarized as

$$\mathcal{L}_{all} = \mathcal{L}_{ms} + \mathcal{L}_{le} + \lambda \mathcal{L}_{bg}, \quad (7)$$

in which λ is a hyper-parameter.

4. Experiments

4.1. Datasets and Implementation Details

We conduct experiments on two benchmark datasets: Colored Large-scale Watermark Dataset (CLWD) [38] and LOGO30K Dataset [8]. Both datasets are constructed with colored complex watermarks, which are challenging for watermark removal task.

CLWD [38]: In CLWD, the training set has $60,000$ images with 160 different watermarks and the test set has $10,000$ images with 40 different watermarks. The watermarks are collected from public image websites. The background images in the training (*resp.*, test) set are randomly chosen from the training (*resp.*, test) set of PASCAL VOC2012 [13]. When superimposing the collected watermarks on the background images, the location, scale, rotation, and transparency of watermarks are randomly determined, with the watermark transparency in the range of [0.3, 0.7].

LOGO30K [8]: In LOGO30K, the training set has $28,352$ images and the test set has $4,051$ images. The watermarks collected from Internet include more than 1k famous logos. The background images come from the VAL2014 subset of MSCOCO [36] dataset. When creating watermarked images, the watermark transparency is set in the range of [0.35, 0.85].

Following previous methods [38, 33], we use PSNR, SSIM, RMSE, and RMSE_w to evaluate the restored background, in which RMSE_w means RMSE calculated within the watermarked region. We use IoU and F_1 to evaluate the predicted watermark mask.

Our method is implemented with Pytorch [41]. The input image size is set to 256×256. During training, we choose Adam [29] optimizer with the initial learning rate being 0.001. All methods are trained and evaluated on Ubuntu 4.18.0, with 512GB memory, Intel(R) Xeon(R) Platinum 8358 CPU, and one A100 SXM4 40GB GPU. The encoder E has five encoder blocks and the decoder D^{ms}/D^{bg} has four decoder blocks, in which the encoder and decoder block structures are borrowed from [33]. The transformer block T has two transformer layers [45]. We set $K = 8$ and $\lambda = 10$ by default.

4.2. Comparison with Baselines

We compare with two groups of baselines: 1) the general methods for image content removal; 2) the watermark removal methods.

For the first group of methods, we compare with recent methods MPRNet [54], UFormer [49], and BIDNet [20]. We also include UNet [42] for general image-to-image translation as a baseline. For the second group of methods, we compare with the following watermark removal methods: Li *et al*. [31], Cao *et al*. [2], WDNet [38], BVMR [23], SplitNet [8], SLBR [33], DKSP [56].

We first evaluate the restored backgrounds from different methods, which are summarized in Table 1. We observe that the methods specifically designed for watermark removal are generally better than general content removal methods [54, 49, 20]. Among the watermark removal methods, [8, 33, 56] achieve competitive performance due to the multi-task framework. Our method significantly outperforms the existing methods, including [56] which uses additional semantic information. The superiority of our method is attributed to the fined-grained watermark removal ability

Method	CLWD				LOGO30K			
	PSNR↑	SSIM ↑	RMSE↓	RMSE_w ↓	PSNR↑	SSIM ↑	RMSE↓	RMSE_w ↓
U-Net [42]	23.21*	0.8567*	19.35*	48.43*	24.64*	0.8816	17.84	43.29*
MPRNet [54]	35.19	0.9725	5.52	20.76	37.81	0.9866	4.26	20.67
UFormer [49]	34.07	0.9784	6.29	25.66	39.84	0.9936	3.34	18.68
BIDNet [20]	35.02	0.9709	5.66	21.75	36.63	0.9829	4.83	24.29
Li *et al*. [31]	27.96*	0.9161*	12.63*	46.80*	30.51*	0.9312	10.51	39.11*
Cao *et al*. [2]	29.04*	0.9363*	10.36*	41.21*	32.18*	0.9654	8.20	35.16*
WDNet [38]	35.53*	0.9738*	5.11*	17.27*	39.15*	0.9874	3.49	15.94*
BVMR [23]	35.89*	0.9734*	5.02*	18.71*	38.28*	0.9852*	3.52	16.72*
SplitNet [8]	37.41*	0.9787*	4.23*	15.25*	41.27*	0.9910*	3.12	14.85*
SLBR [33]	38.28*	0.9814*	3.76*	14.07*	41.50*	0.9914	2.98	14.69*
DKSP† [56]	38.84*	0.9813	3.53	12.16*	42.16*	0.9921	2.77	13.78*
Ours	**39.45**	**0.9823**	**3.19**	**11.28**	**42.57**	**0.9923**	**2.46**	**12.44**

Table 1. The results of restored backgrounds of different methods on CLWD [38] and LOGO30K [8]. The results marked with $*$ are directly copied from previous papers [8, 33, 56]. The method marked with † uses additional semantic information. The best results are denoted in boldface.

Method	CLWD		LOGO30K	
	IoU↑	F_1 ↑	IoU↑	F_1 ↑
WDNet [38]	0.6120*	0.7240*	0.6821*	0.8010*
BVMR [23]	0.7021*	0.7871*	0.7287*	0.8305*
SplitNet [8]	0.7196*	0.8027*	0.7414*	0.8411*
SLBR [33]	0.7463*	0.8234*	0.7858*	0.8647*
DKSP† [56]	0.7730*	0.8480*	0.8016*	0.8770*
Ours	**0.7909**	**0.8634**	**0.8185**	**0.8898**

Table 2. The results of predicted watermark masks of different methods on CLWD [38] and LOGO30K [8]. The results marked with $*$ are directly copied from previous papers [33, 56]. The method marked with † uses additional semantic information. The best results are denoted in boldface.

#	T	D^{ms}	D^{bg}	Evaluation Metrics			
				PSNR↑	RMSE_w ↓	IoU↑	F_1 ↑
1		- q	- PFM	40.08	16.60	0.7474	0.8325
2		- $\mathcal{L}_{le}$	- PFM	40.20	16.54	0.7576	0.8409
3		+	-PFM	40.23	16.48	0.7893	0.8674
4		+	+	41.67	13.85	0.8023	0.8772
5		+	SB	41.04	14.84	0.7928	0.8699
6		+	DK	40.82	15.51	0.7896	0.8679
7		+	-g	40.47	16.43	0.7895	0.8672
8	+	+	+	42.57	12.44	0.8185	0.8898

Table 3. The ablation study results on LOGO30K [8]. "SB" is short for scale/bias, "DK" is short for dynamic kernel, and "q" stands for query embeddings.

by considering fine-grained part categories.

We also compare the quality of watermark masks predicted by different methods in Table 2. We only compare with those methods [38, 23, 8, 33, 56] which predict watermark masks. Again, our method achieves the best results on both datasets.

We show the visual comparison between different methods in Figure 3. We compare with the competitive baselines WDNet [38], SplitNet [8], SLBR [33], DKSP [56]. We show both predicted watermark masks and restored backgrounds. Our method can predict accurate and complete watermark masks, while the baseline methods could only predict incomplete mask, when the watermark is composed of multiple parts. This is because we use adapted query embeddings corresponding to different part categories to produce multiple part masks, which can cover the complete watermarked region. For background restoration, since baselines fail to predict complete watermark masks, they cannot erase the whole watermark and the recovered backgrounds are far below expectation. With the predicted complete watermark masks, our method can successfully repair the entire watermarked region by handling different local parts adaptively, producing visually pleasant backgrounds closer to the ground-truth.

4.3. Ablation Studies

We conduct ablation studies to analyze the effectiveness of each component and loss term, which are summarized in Table 3. We first build a basic model without using query embeddings or PFM module in row 1, in which case $\boldsymbol{F}^{ms}$ directly outputs the watermark mask and $\boldsymbol{F}^{bg}$ directly outputs the restored background. Then, we use $K+1$ query embedding to interact with $\boldsymbol{F}^{ms}$ to predict mask without low-entropy loss $\mathcal{L}_{le}$. The obtained mask results in row 2 are better than those in row 1, which proves the effectiveness of query-based mask prediction. However, we observe that the predicted masks for different part categories are prone to resemble each other. Thus, we add $\mathcal{L}_{le}$ to enforce K part masks to be different from each other. By comparing row 3 with row 2, we observe that the mask results are significantly improved by discovering multiple part

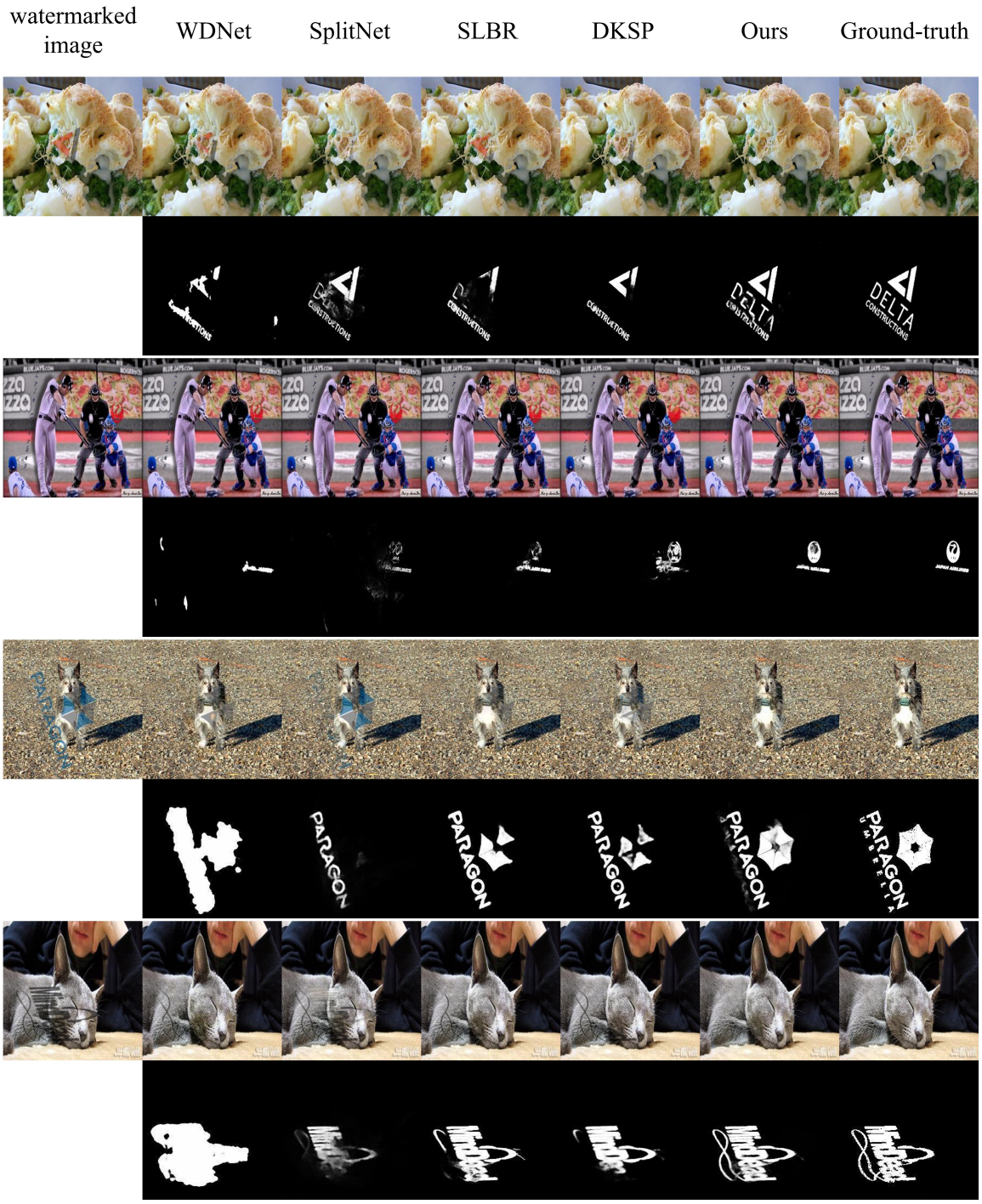

Figure 3. From left to right, we show the watermarked image, the results of WDNet [38], SplitNet [8], SLBR [33], DKSP [56], our method, and the ground-truth. In each group, the top row shows the restored background image and the bottom row shows the predicted watermark mask. The top two examples are from LOGO30K [8] and the bottom two examples are from CLWD [38].

categories. Based on row 3, we insert our PFM block into the background decoder, leading to row 4. We observe that the background results are greatly improved. In the meanwhile, the mask results are also improved, which might benefit from the information sharing between two tasks.

Based on row 4, we explore several variants of our PFM

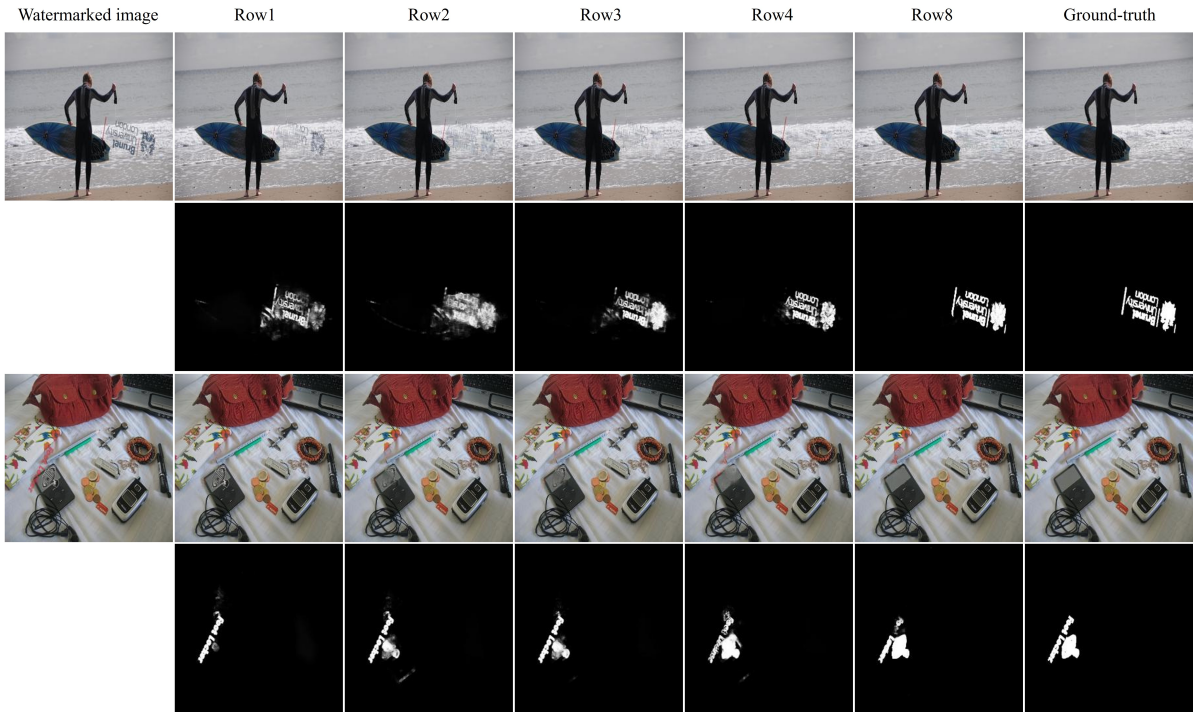

Figure 4. From left to right, we show the watermarked image, the results of row 1, 2, 3, 4, 8 in Table 3, and the ground-truth. In each group, the top row shows the restored background image and the bottom row shows the predicted watermark mask.

block. Recall that we use the dynamic strategy proposed in [28]. We also try some alternative ways of dynamic feature transformation. One alternative is dynamic convolution similar to [33]. Specifically, we use adapted query embeddings to predict dynamic convolution kernels for different part categories, which are applied to the corresponding regions in the feature map. Another alternative is predicting scales and biases similar to [27]. Specifically, we use adapted query embeddings to predict channel-wise scales and biases for different part categories, which are applied to the corresponding local regions in the feature map. The comparison among row 4, row 5, and row 6 shows that our choice in the method is the optimal one. Moreover, we remove the guidance from query embeddings and treat the input channel scales as learnable parameters. Without the guidance of query embeddings, the learnt dynamic convolution weights are less effective and thus the performance (row 7) becomes worse than those with guidance (row 4-6).

At last, we add the transformer block T to adapt query embeddings to each specific image. The obtained results in row 8 verify the validness of adapted query embeddings, because the local parts from different images belonging to the same part category may still have highly contrastive visual appearances.

In Figure 4, we show the results of row 1, 2, 3, 4, 8 in Table 3. From row 1 to row 4, although the quality of restored backgrounds are getting better, the watermarks are not completely erased and residual watermark traces can be obviously seen. When using our full method equipped with all modules, row 8 achieves the best results that are close to the ground-truth backgrounds.

5. Conclusion

In this work, we have proposed a query-based multi-task framework for visible watermark removal, in which query embeddings corresponding to different part categories are jointly used for fine-grained mask prediction and background reconstruction. Our network can split one watermarked region into multiple local parts and restore different local parts adaptively. By considering fine-grained watermarked parts, our network surpasses the existing methods on two benchmark datasets.

Acknowledgement

The work was supported by the National Natural Science Foundation of China (Grant No. 62076162), the Shanghai Municipal Science and Technology Major/Key Project, China (Grant No. 2021SHZDZX0102, Grant No. 20511100300).

References

[1] Jiezhang Cao, Jingyun Liang, Kai Zhang, Yawei Li, Yulun Zhang, Wenguan Wang, and Luc Van Gool. Reference-based image super-resolution with deformable attention transformer. In *ECCV*, 2022. 2

[2] Z. Cao, S. Niu, J. Zhang, and X. Wang. Generative adversarial networks model for visible watermark removal. *IET Image Processing*, 13(10):1783–1789, 2019. 2, 5, 6

[3] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020. 2

[4] Hanting Chen, Yunhe Wang, Tianyu Guo, Chang Xu, Yiping Deng, Zhenhua Liu, Siwei Ma, Chunjing Xu, Chao Xu, and Wen Gao. Pre-trained image processing transformer. In *CVPR*, 2021. 2

[5] Yinpeng Chen, Xiyang Dai, Mengchen Liu, Dongdong Chen, Lu Yuan, and Zicheng Liu. Dynamic relu. In *ECCV*, 2020. 3

[6] Bowen Cheng, Alex Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. *NeurIPS*, 2021. 2, 3, 4

[7] Xiaofeng Cong, Jie Gui, Kai-Chao Miao, Jun Zhang, Bing Wang, and Peng Chen. Discrete haze level dehazing network. In *ACMMM*, 2020. 2

[8] Xiaodong Cun and Chi-Man Pun. Split then refine: stacked attention-guided resunets for blind single image visible watermark removal. In *AAAI*, 2021. 2, 5, 6, 7

[9] Xiaodong Cun, Chi-Man Pun, and Cheng Shi. Towards ghost-free shadow removal via dual hierarchical aggregation network and shadow matting gan. In *AAAI*, 2020. 2

[10] Yingying Deng, Fan Tang, Weiming Dong, Chongyang Ma, Xingjia Pan, Lei Wang, and Changsheng Xu. Stytr2: Image style transfer with transformers. In *CVPR*, 2022. 2

[11] Bin Ding, Chengjiang Long, Ling Zhang, and Chunxia Xiao. Argan: Attentive recurrent generative adversarial network for shadow detection and removal. In *ICCV*, 2019. 2

[12] Hang Dong, Jinshan Pan, Lei Xiang, Zhe Hu, Xinyi Zhang, Fei Wang, and Ming-Hsuan Yang. Multi-scale boosted dehazing network with dense feature fusion. In *CVPR*, 2020. 2

[13] Mark Everingham, SM Eslami, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes challenge: A retrospective. *International journal of computer vision*, 111(1):98–136, 2015. 5

[14] Lijun Fu, Bei Shi, Ling Sun, Jiawen Zeng, Deyun Chen, Hongwei Zhao, and Chunwei Tian. An improved u-net for watermark removal. *Electronics*, 11(22):3760, 2022. 2

[15] Hang Gao, Xizhou Zhu, Steve Lin, and Jifeng Dai. Deformable kernels: Adapting effective receptive fields for object deformation. *arXiv preprint arXiv:1910.02940*, 2019. 3

[16] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020. 2

[17] Yves Grandvalet and Yoshua Bengio. Semi-supervised learning by entropy minimization. *NIPS*, 2004. 2, 4

[18] Zonghui Guo, Dongsheng Guo, Haiyong Zheng, Zhaorui Gu, Bing Zheng, and Junyu Dong. Image harmonization with transformer. In *ICCV*, 2021. 2

[19] David Ha, Andrew Dai, and Quoc V Le. Hypernetworks. *arXiv preprint arXiv:1609.09106*, 2016. 3

[20] Junlin Han, Weihao Li, Pengfei Fang, Chunyi Sun, Jie Hong, Mohammad Ali Armin, Lars Petersson, and Hongdong Li. Blind image decomposition. In *ECCV*, 2022. 2, 5, 6

[21] Yizeng Han, Gao Huang, Shiji Song, Le Yang, Honghui Wang, and Yulin Wang. Dynamic neural networks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. 3

[22] Adam W Harley, Konstantinos G Derpanis, and Iasonas Kokkinos. Segmentation-aware convolutional networks using local attention masks. In *ICCV*, 2017. 3

[23] Amir Hertz, Sharon Fogel, Rana Hanocka, Raja Giryes, and Daniel Cohen-Or. Blind visual motif removal from a single image. In *CVPR*, 2019. 2, 3, 5, 6

[24] Lin Huang, Jianchao Tan, Ji Liu, and Junsong Yuan. Hand-transformer: non-autoregressive structured modeling for 3d hand pose estimation. In *ECCV*, 2020. 2

[25] Xu Jia, Bert De Brabandere, Tinne Tuytelaars, and Luc V Gool. Dynamic filter networks. *NeurIPS*, 2016. 3

[26] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *ECCV*, 2016. 5

[27] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *CVPR*, 2019. 8

[28] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *CVPR*, 2020. 4, 5, 8

[29] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 5

[30] HyunJae Lee, Hyo-Eun Kim, and Hyeonseob Nam. Srm: A style-based recalibration module for convolutional neural networks. In *ICCV*, 2019. 3

[31] Xiang Li, Chan Lu, Danni Cheng, Wei-Hong Li, Mei Cao, Bo Liu, Jiechao Ma, and Wei-Shi Zheng. Towards photo-realistic visible watermark removal with conditional generative adversarial networks. In *ICIG*, 2019. 2, 5, 6

[32] Jingyun Liang, Jiezhang Cao, Guolei Sun, Kai Zhang, Luc Van Gool, and Radu Timofte. Swinir: Image restoration using swin transformer. In *ICCV*, 2021. 2

[33] Jing Liang, Li Niu, Fengjun Guo, Teng Long, and Liqing Zhang. Visible watermark removal via self-calibrated localization and background refinement. In *ACMMM*, 2021. 1, 2, 3, 5, 6, 7, 8

[34] Zhengyu Liang, Yingqian Wang, Longguang Wang, Jungang Yang, and Shilin Zhou. Light field image super-resolution with transformers. *IEEE Signal Processing Letters*, 29:563–567, 2022. 2

[35] Kevin Lin, Lijuan Wang, and Zicheng Liu. End-to-end human pose and mesh reconstruction with transformers. In *CVPR*, 2021. 2

[36] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 5

[37] Xinwei Liu, Jian Liu, Yang Bai, Jindong Gu, Tao Chen, Xiaojun Jia, and Xiaochun Cao. Watermark vaccine: Adversarial attacks to prevent watermark removal. In Shai Avidan, Gabriel J. Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner, editors, *ECCV*, 2022. 2

[38] Yang Liu, Zhen Zhu, and Xiang Bai. Wdnet: Watermark-decomposition network for visible watermark removal. In *WACV*, 2021. 2, 5, 6, 7

[39] Ningning Ma, Xiangyu Zhang, Jiawei Huang, and Jian Sun. Weightnet: Revisiting the design space of weight networks. In *ECCV*, 2020. 3

[40] Chong Mou, Qian Wang, and Jian Zhang. Deep generalized unfolding networks for image restoration. In *CVPR*, 2022. 2

[41] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *NeurIPS*, 2019. 5

[42] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015. 2, 3, 5, 6

[43] Martin Simonovsky and Nikos Komodakis. Dynamic edge-conditioned filters in convolutional neural networks on graphs. In *CVPR*, 2017. 3

[44] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 5

[45] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *NeurIPS*, 2017. 2, 3, 5

[46] Cong Wang, Yutong Wu, Zhixun Su, and Junyang Chen. Joint self-attention and scale-aggregation for self-calibrated deraining network. In *ACMMM*, 2020. 2

[47] Huiyu Wang, Yukun Zhu, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. Max-deeplab: End-to-end panoptic segmentation with mask transformers. In *CVPR*, 2021. 2

[48] Yuqing Wang, Zhaoliang Xu, Xinlong Wang, Chunhua Shen, Baoshan Cheng, Hao Shen, and Huaxia Xia. End-to-end video instance segmentation with transformers. In *CVPR*, 2021. 2

[49] Zhendong Wang, Xiaodong Cun, Jianmin Bao, Wengang Zhou, Jianzhuang Liu, and Houqiang Li. Uformer: A general u-shaped transformer for image restoration. In *CVPR*, 2022. 2, 5, 6

[50] Fuzhi Yang, Huan Yang, Jianlong Fu, Hongtao Lu, and Baining Guo. Learning texture transformer network for image super-resolution. In *CVPR*, 2020. 2

[51] Youzhao Yang and Hong Lu. Single image deraining via recurrent hierarchy enhancement network. In *ACMMM*, 2019. 2

[52] Jinsu Yoo, Taehoon Kim, Sihaeng Lee, Seung Hwan Kim, Honglak Lee, and Tae Hyun Kim. Enriched cnn-transformer feature aggregation networks for super-resolution. In *WACV*, 2023. 2

[53] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Ming-Hsuan Yang. Restormer: Efficient transformer for high-resolution image restoration. In *CVPR*, 2022. 2

[54] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, Ming-Hsuan Yang, and Ling Shao. Multi-stage progressive image restoration. In *CVPR*, 2021. 2, 5, 6

[55] Jing Zhang, Yang Cao, Zheng-Jun Zha, and Dacheng Tao. Nighttime dehazing with a synthetic benchmark. In *ACMMM*, 2020. 2

[56] Xing Zhao, Li Niu, and Liqing Zhang. Visible watermark removal with dynamic kernel and semantic-aware propagation. In *BMVC*, 2022. 1, 2, 3, 5, 6, 7

[57] Chen Zheng, Yulun Zhang, Jinjin Gu, Yongbing Zhang, Linghe Kong, and Xin Yuan. Cross aggregation transformer for image restoration. In *NeurIPS*, 2022. 2

[58] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. In *ICLR*, 2021. 2