Journal of Intelligent Computing and Networking
https://www.ffspub.com/index.php/jicn/index
ISSN 3079-9228 (print)
E-mail: jicn.office@ffspub.com

Article

Seam-carving Localization in Digital Images

Xiaoyi WANG[1], Bo LIU[2], Xiuli BI[2†], Bin XIAO[2†]
[1]*School of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing 400065, China*
[2]*School of Artificial Intelligence, Chongqing University of Posts and Telecommunications, Chongqing 400065, China*
[†]*E-mail: bixl@cqupt.edu.cn, xiaobin@cqupt.edu.cn*

Received: May 22, 2025 / Revised: June 30, 2025 / Accepted: July 7, 2025 / Published online: July 18, 2025

Abstract: Seam-carving is a relatively new image re-targeting technique. While it can be used for legitimate image re-targeting, it also provides a tool for malicious purposes, such as object removal. However, existing methods either classify images in blocks or try to learn faint seam traces from forgeries directly, with low accuracy in the former and inefficiency in the latter. To break these limitations, a new seam-carving localization method is proposed in this research, which can be used to solve the correlation forensic authentication tasks based on seam-carving. JPEG compression brings regular block artifacts to the image, which can be a suitable medium for seam-carving localization. Therefore, we design a multi-block network structure and propose an effective training strategy to localize seams in images. First, we extract the block artifacts hidden in the image self-supervised; second, we input the location map of the seams as guidance to localize the seams from the corrupted properties. As expected, the network can quickly localize the seams with a small amount of training data. By utilizing this prior, we achieve the detection of object removal based on seam-carving. Extensive experiments demonstrate the feasibility and effectiveness of the proposed method.

Keywords: Seam-carving localization block artifact; self-supervised; object removal detection
https://doi.org/10.64509/jicn.11.17

1 Introduction

In the 21st century, facing various electronic products, such as laptops, smartphones, smartwatches, and so on, re-targeting one image to fit different screen sizes is becoming more and more attention. The basic image re-targeting techniques mainly include linear scaling and center cropping, and they are widely utilized in various applications. However, because these basic methods pay attention to handling geometric transformations, they harm the image's visual effect. In 2007, Avidan [1] proposed a content-aware image re-targeting technique named seam-carving,and compared with basic image re-targeting methods, the seam-carving algorithm keeps the main content of the image unchanged and discards unimportant background information.To achieve better results, the seam-carving algorithm has been continuously optimized[2–6].

Due to the good performance of seam-carving, this technique has been integrated into many image editing applications (e.g., Adobe Photoshop and ImageMagic),and even in the medical field, where it has been utilized for optimizing mammogram images[7]. Similarly, it has been used for various malicious purposes, such as shrinking or even removing content unfavorable to the tamperer or highlighting areas favorable to the tamperer[8]. The forgery based on the seam-carving may not leave visual clues visible to human eyes [1, 9, 10]. Therefore, the detection and location of the seam-carving forgery have become a necessary topic in image forensics.

Although there has been extensive research on seam-carving, most of them cannot be used for seam-carving forgery detection. The idea of existing methods [8, 11–19] is the same, by extracting some features to classify whether an image has experienced seam-carving or not, they focus on the classification of the legitimate re-targeting image. Indeed, it is more meaningful to detect and localize the removal of malicious objects by seam-carving. However, object removal based on seam-carving is not easy to detect,since the general object removal algorithms utilize image inpainting algorithms to fill the blank regions caused by removing the object. Many detection methods locate the removed object by exploring the inpainting algorithm [20–22]. They are not applicable to detect the removed objects operated by seam-carving.

† Corresponding author: Xiuli Bi, Bin Xiao
*Academic Editor: Quan Zhou

Even though the seam-carving image re-targeting technique performs well, it still inevitably corrupts the correlation between adjacent pixels of an image, such as the consistency between pixels, *etc*. JPEG (Joint Photographic Experts Group) compression brings regular block artifact properties to the image, which can be a suitable medium for seam-carving localization. According to the summarization of requested images through the forensic website over two years [23], the JPEG format was found to be the most requested (77.95%), followed by PNG (20.67%). Since the blocking artifact is the natural property of JPEG images, it has been used in many forensic identification tasks [24–28]. Therefore, we utilize this property as a medium to detect and localize the seam operation in the images that undergo JPEG compression. Our main contributions are summarized as follows:

- A new pixel-level seam-carving localization method is proposed, which relies on the image's properties and can detect malicious operations detection, especially object removal based on seam-carving.
- A multi-block network structure was designed, composed of four blocks with different unique characteristics and purposes, which combine the residual structure and attention mechanisms.
- To capture the traces caused by seam-carving, we propose an effective training strategy: first, we extract the properties hidden in the image self-supervised; second, we input the location map of the seams as guidance to localize the seams from the corrupted properties.
- Compared with the existing methods, the proposed method achieves better localization precision and generality.

The rest of this paper is organized as follows: Section II devotes itself to the theoretical concepts of the seam-carving technique. Section III describes our proposed method, which includes the theoretical analysis of the JPEG blocks and detailed information about the network. Section IV demonstrates ablation studies and comparative experiments and discusses them in Section V. Finally, Section VI draws conclusions and suggestions for future work.

2 Related Work

2.1 Seam-carving

The definition of a seam is a set of connected pixels that traverses the image in the vertical (from top to bottom) or horizontal (from left to right) layout. Those pixels are located in the less important areas in the images selected by an energy function. Seam-carving is a content-aware image re-targeting approach that includes seam-removal and seam-insertion, which reduce and enlarge the size of an image by removing or replicating these low-energy seams.

Avidan first proposed a strategy for finding the seam with a minimum total energy value in 2007 [1] and defined the function of the energy value of a specific pixel in the seam as shown in Eq. 1.

$$e(I) = \left|\frac{\partial}{\partial x}I\right| + \left|\frac{\partial}{\partial y}I\right|. \tag{1}$$

A candidate vertical seam in the image I can be expressed as $\{I(s_i)\}_{i=1}^{N_1}$. The optimal one seam s^* is the seam with the lowest energy cost, which is defined by the following formula.

$$s^* = \min_s \{E(s)\} = \min_s \left\{\sum_{i=1}^{N_1} e(I(s_i))\right\}. \tag{2}$$

The optimal seam s^* can be found using a dynamic programming algorithm. For a particular pixel, there are only three pixels in the previous row that can be connected to it, whose coordinates are $(i-1, j-1)$, $(i-1, j)$ and $(i-1, j+1)$. We denote the cumulative minimum energy of the seam from the first row to (i, j) by $M(i, j)$. Then, the minimum cumulative energy $M(i, j)$ at (i, j) can be calculated by Eq. 3.

$$\begin{aligned} M(i,j) &= e(i,j) + \arg\min \\ &(M(i-1,j-1), M(i-1,j), M(i-1,j+1)). \end{aligned} \tag{3}$$

Using Eq. 3. to calculate downwards from the second row of the image, when calculating the cumulative minimum energy for each pixel position in the last row is complete, we select the smallest value from the last row and record its coordinates. Then, we backtrack to find the low-energy pixel connected to this pixel, and finally, we can get the seam with minimum total energy. After removing a seam, the cumulative energy should recalculate, and we repeat the stages until the image reaches the targeted size.

2.2 Seam-carving Detection

Seam-carving has been used for a variety of malicious purposes with widespread use of it, and the trend of such illegal operations is growing [8]. Therefore, researchers have proposed many relevant methods, which can be divided into traditional and neural network-based methods.

Traditional methods determine whether an image has undergone a seam operation through the human-designed features, and its classification accuracy also depends on the quality of the feature extraction process. The laws that humans formulate often have many limitations. Among traditional methods, Lu *et al.* verified whether pre-embedded SIFT (Scale Invariant Feature Transform) features changed [11]. Ryu *et al.* extracted three features for classification based on energy, seams, and noise [12]. Based on [12], Yin *et al.* proposed to use the information of the difference of LBP values on both sides of the seam [13]. Lu *et al.* proposed a Local Neighborhood Magnitude Occurrence Pattern (LNMOP) and used its feature histogram for detection [14]. Liu *et al.* analyzed the domain joint density of the coefficients by Discrete Cosine Transformation (DCT) [8]. Wattanachote *et al.* extracted blocks Characteristics Matrix (BACM) intra-block and inter-block histograms to measure JPEG image blocks symmetry changes [15]. Liu *et al.* combined information from spatial and frequency domains to find detectable features based on a feature mining approach [16, 17]. Some methods use steganography to determine whether the information in the image has been tampered with [29–32].

With the rapid development of deep learning, many neural network methods have been proposed. Nataraj *et al.* built one detector for classification, the other to detect patches that

have been seam carved for patch-level localization [33]. Gudavalli *et al.* proposed a method to learn the faint seam traces from seam-carving images directly [34]. [18] utilized the neural network to find the weak seam information in an image, which is the classification of forged images, however, the proposed method can localize the seams in the image, which use for the detection of malicious operations, especially object removal.

3 The Proposed Method

Although there are many studies on seam-carving, most of them just classify whether an image has undergone seam-carving or not [14–17], which can not be used for the detection of malicious operations, such as object removal. To tackle complex tamper detection problems, we need to realize seam-carving localization. Finding the location of the seam-carving directly from the image is a difficult task. Fortunately, some regular properties of the image can help us capture these corruptions. JPEG compression is a widely used image technique with unique properties and distributions, as shown in Figure 1. (a), we compress the original image using different quality factors (QF) and observe the regions in the red box, we can see that the block artifacts (BA) appeared, and the smaller the compression quality factor is, the more obvious the block artifacts are. The properties in JPEG images provide a good opportunity for us to localize seam-carving. As shown in Figure 1. (b), if we randomly remove or insert a seam from the JPEG image, all the 8×8 distribution that the seam passes through will be corrupted, resulting in obvious misalignment. So we do a further investigation of the JPEG compression block artifacts and design an effective training strategy to realize the localization of seam-carving. By utilizing this prior, we simulate the change of block artifacts when the JPEG image undergoes seam-carving, as shown in Figure 1. (c-d), it is difficult to find the traces of seams on the seam-removal (SR) or seam-insertion (SI) images, but the seams seriously corrupt the 8×8 distribution, we can even see the outline of the seams from the position of the corrupted block artifacts, which can be a medium for seam-carving localization. However, it is almost impossible to observe the change of this property directly from the image. If we can extract it from the image, as in Figure 1. (d), seam-carving localization will no longer be difficult.

3.1 Problem Formulation

JPEG compression is a lossy compression where the image is divided into 8×8 non-overlapping blocks and compressed separately, which brings block artifacts to the compressed image and degrades the visual quality. Suppose $f(i,j)$ is the pixel matrix of an 8×8 image block. $f(i,j)$ will be transformed by

$$f^{'}(i,j) = iDCT\left(\left\lfloor \frac{DCT\left(f\left(i,j\right)\right)}{Q\left(i,j\right)} \right\rfloor \cdot Q\left(i,j\right)\right), \qquad (4)$$

where $DCT()$ is a two-dimensional Discrete Cosine Transform (DCT), $\lfloor \cdot \rfloor$ means rounding down, and Q is the JPEG quantization matrix given by Joint Photographic Experts Group (JPEG). If the rounded-off part is regarded as the truncation error $e(i,j) \in (0, Q(i,j))$ caused by quantization procedure, Eq. (4) can be rewritten as

$$f^{'}(i,j) = iDCT(DCT(f(i,j)) - e(i,j)). \qquad (5)$$

Since $iDCT()$ transformation satisfies the linear invariant property, the compressed block $f'(i,j) = f(i,j) + iDCT(-e(i,j))$, and we can denote the compressed image as

$$I_{QF} = I + E_{QF}, \qquad (6)$$

where E_{QF} represents $iDCT(-e)$ of all the 8×8 blocks, which causes the block artifacts of JPEG image.

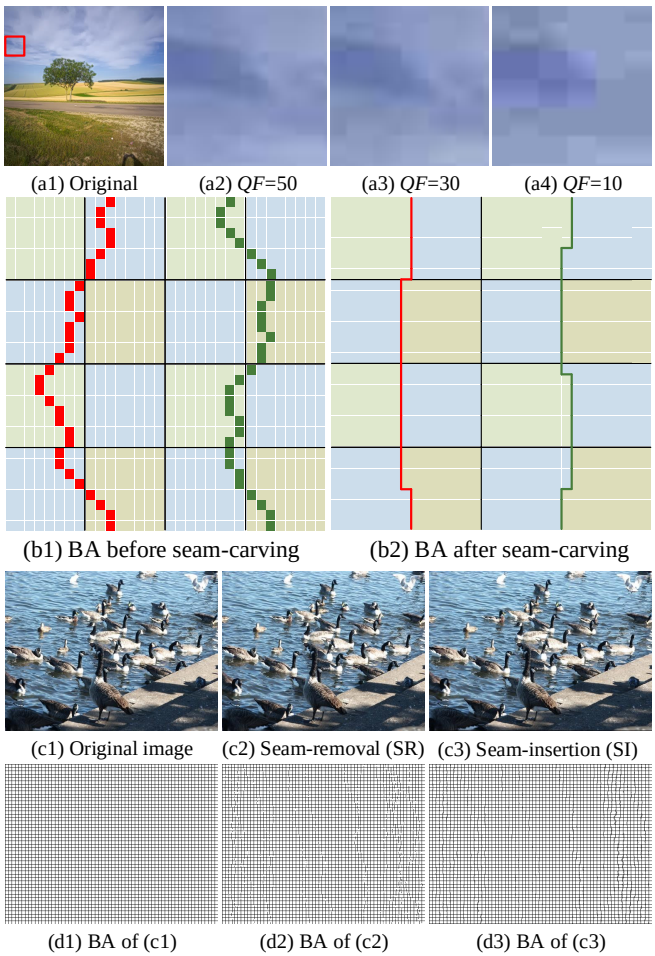


Figure 1: The block artifacts (BA) distribution of JPEG images and the influence of seam-carving on this property. (a2-a4) The original image was compressed with $QF = 90, 70, 50$. (b1-b2) are the distribution of BA before and after seam-carving, respectively; the red line indicates seam-removal (SR), and the green line indicates seam-insertion (SI). (c1-c2) are the original JPEG images, SR and SI images, (d1) is the block artifacts we simulate from (c1), and when we perform seam-removal and seam-insertion on (c1), we do the same operation at the corresponding position of (d1) to get (d2-d3).

In recent years, many block artifact removal methods based on convolutional neural networks have been developed with great success [35–37], their goal is to remove unwanted artifacts in the compressed image I_{QF}. In contrast to these

methods, our goal is to remove the image content I and extract the block artifacts E_{QF} by

$$\arg\min \|G(I_{QF};\theta_G)-(I_{QF}-I)\|_F^2, \tag{7}$$

where θ_G represents the set of all neural network parameters G. The network continuously narrows the gap between its output $G(I_{QF};\theta_G)$ and block artifacts E_{QF} by inputting the original image I and the compressed image I_{QF} in pairs, finally, it can extract E_{QF} directly from the JPEG image. Then we input the seam-carving image I_S and its corresponding seams location map M in pairs to guide the network:

$$\arg\min \|G(I_S;\theta_G)-M\|_F^2, \tag{8}$$

which will localize the seams by the corrupted 8×8 distributions of block artifacts E_{QF}, and then output the localization map of the seams $G(I_S;\theta_G)$.

3.2 Network Architecture

To enable the network for the task in Eq.7 and Eq.8, we meticulously design a multi-block network structure, which combines the residual structure [38] and attention mechanisms [39], and we prevented the weak features from disappearing by excluding the pooling layer, inspired by the insight and approaches in [40, 41]. The proposed network can be expressed as

$$\begin{aligned}\mathbf{y}&=G(x;\theta_G)\\&=B_4^{(n_4)}\left(RA^{(n_3)}\left(B_2^{(n_2)}\left(B_1^{(n_1)}(\mathbf{x})\right)\right)\right),\end{aligned} \tag{9}$$

where $\mathbf{x},\mathbf{y}$ denote the input and output of the network respectively, $B_1(\cdot)$, $B_2(\cdot)$, $RA(\cdot)$ and $B_4(\cdot)$ denote four kinds of blocks, and n_{th} is the number of n-th block.

The first block $B_1(\cdot)$ is composed of 64 convolutional (Conv) filters of size $\mathbb{R}^{3\times3\times1}$ with stride 1, followed by the rectified linear unit (ReLU) as an activation function for non-linearity, and the number n_1 was set to 1. It has been demonstrated in [42] that batch normalization (BN) [43] can speed up and stabilize the training process. For block $B_2(\cdot)$, 64 Conv filters of size $\mathbb{R}^{3\times3\times64}$ with stride 1 are used, the BN layer is added between Conv and ReLU, and the number n_2 was set to 2. $B_1(\cdot)$ and $B_2(\cdot)$ blocks extract shallow features from the input data.

To improve the ability to extract forgery-related features, we further enhance the structure by combining the residual structure and the attention mechanisms (RA), rather than simply using ordinary convolutional layers. The residual structure $Res(\cdot)$ can be formulated as:

$$\mathbf{r}=Res(\mathbf{m}_{in})=\mathbf{m}_{in}+B_2^{(2)}(\mathbf{m}_{in}), \tag{10}$$

where $\mathbf{m}_{in}$ and $\mathbf{r}$ denote the input and output of the $Res(\cdot)$, the feature map is reused by using the skip-connection, which can alleviate the vanishing gradient problem that adversely affects the convergence of deep-structured CNNs. The attention mechanism we use is the Spatial Channel Squeeze-and-Excitation (SCSE) [39], which consists of two attention branches, the spatial squeeze and channel excitation (SSE) and the channel squeeze and spatial excitation (CSE). The core idea of squeeze and excitation is that the network learns the weights $\mathbf{w}_s,\mathbf{w}_c$ of the features according to the loss, motivates the important features, and suppresses the unimportant ones so that the network model achieves better results,

$$\begin{aligned}\mathbf{m}_{out}&=RA(\mathbf{m}_{in})\\&=SCSE(Res(\mathbf{m}_{in}))\\&=Conc(SSE(\mathbf{r}),CSE(\mathbf{r}))\\&=Conc(\mathbf{r}\odot_s\mathbf{w}_s,\mathbf{r}\odot_c\mathbf{w}_c),\end{aligned} \tag{11}$$

where $\mathbf{m}_{out}$ denote the output of the $RA(\cdot)$, $\odot_s$ and $\odot_c$ indicate the spatial-wise and channel-wise multiplication respectively, $Conc$ means the concatenate operation.

The output of SSE is the weight matrix $\mathbf{w}_s\in\mathbb{R}^{H\times W}$ that signifies the network which features at which position on the $H\times W$ dimension is more important, and the network uses the weights at different positions to reward or penalize the corresponding feature, which can be implemented by the convolution layer, and the parameters in it are updated as the gradient is back-propagated.

The output of CSE is a weight vector $\mathbf{w}_c\in\mathbb{R}^{1\times1\times C}$ that signifies the network which features at which channel on the C dimension are more important. The generation of the weight vector $\mathbf{w}_c$ is relatively complicated. First, we need to convert $\mathbf{r}$ into a $\mathbb{R}^{1\times1\times C}$ vector $\mathbf{v}$ by a global average pooling layer, the k_{th} value in the vector $\mathbf{v}$ is calculated as follows

$$\mathbf{v}_k=\frac{1}{H\times W}\sum_{i=1}^{H}\sum_{j=1}^{W}\mathbf{r}(i,j,k). \tag{12}$$

and for vectors, we usually use fully-connected layers (FC) to learn the relationships between the features,

$$\mathbf{w}_c=ReLU(FC_2*Sigmoid(FC_1*\mathbf{v})), \tag{13}$$

where $FC_1\in\mathbb{R}^{\frac{C}{h}\times C}$ and $FC_2\in\mathbb{R}^{C\times\frac{C}{h}}$ are the weight matrix of two fully connected layers respectively, h is the number of hidden nodes in the middle layer, and we set it to 16 in our experiments. When we concatenate the output of SSE and CSE, the channels of the feature map increase from 64 to 128, then we use 64 Conv filters of size $\mathbb{R}^{1\times1\times128}$ to downscale the number of channels to 64.

The last block $B_4(\cdot)$ is composed of one Conv filter of size $\mathbb{R}^{3\times3\times64}$ with stride 1, which is used to reconstruct the output, and the number n_4 was set to 1. The four blocks have different unique characteristics and purposes, resulting from our optimal combination under multiple attempts. Extensive experiments have proved that the proposed network structure can effectively achieve the learning task and reach the final goal: seam-carving localization. As shown in Figure 2, we average the multi-channel feature maps output from each block and then visualize them to observe their functions. As we expected, $B_1(\cdot)$ and $B_2(\cdot)$ extract the shallow features of the image, such as the edges and details. $RA(\cdot)$ block is the kernel of the proposed network structure. In the feature maps of $RA^{(1)}$ and $RA^{(2)}$, we can see noticeable block artifacts as well as the outline of seams, and after $RA^{(3-5)}$ network finds more precise seam locations from the corrupted blocks and finally outputs localization results through $B_4(\cdot)$.

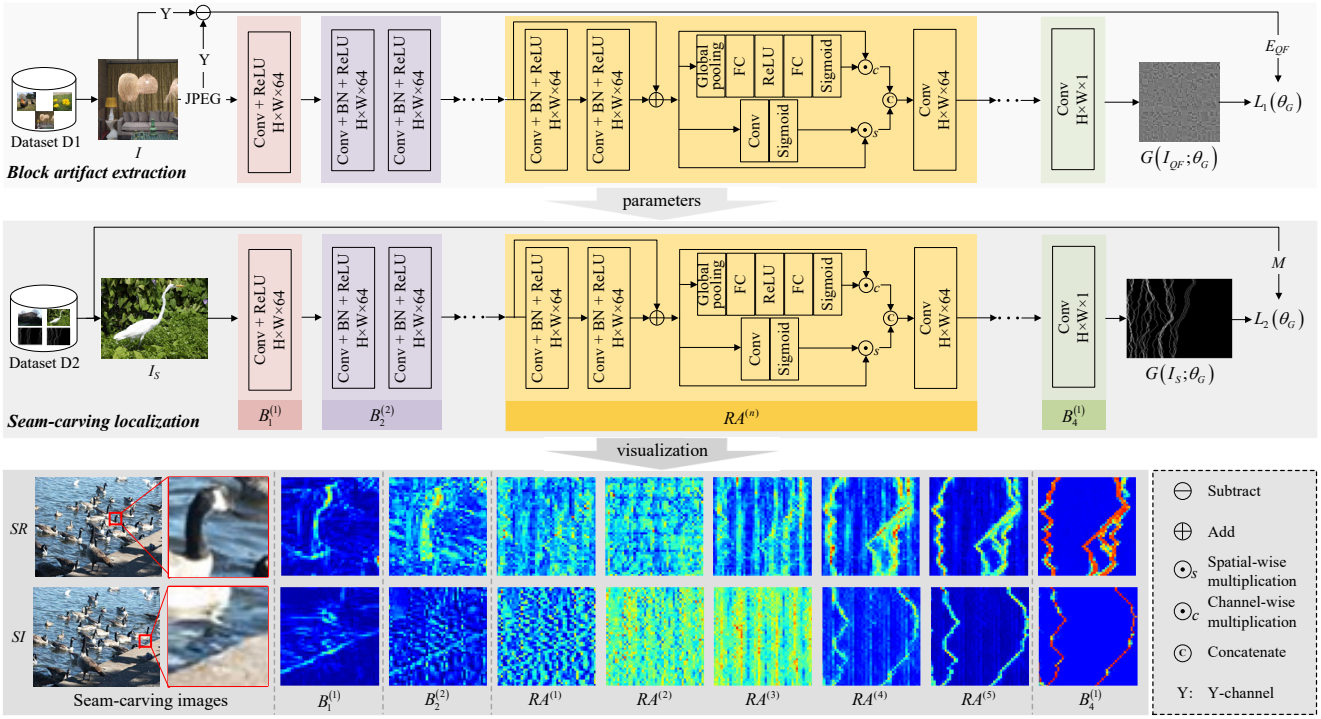


Figure 2: Structure of the proposed method. The training strategy consists of two steps: self-supervised training for the network to extract JPEG block artifacts,and end-to-end training for the network to realize pixel-level localization of seams. The visualization of the four blocks in the proposed network structure, $B_1(\cdot)$, $B_2(\cdot)$, $RA(\cdot)$, and $B_4(\cdot)$.

3.3 Seam Carving Localization

Although the JPEG block artifact is a suitable property for detecting and localizing the seams, it is difficult to estimate the error E_{QF} directly from a picture due to the lack of the original image. Deep learning is a powerful feature learning method for learning the relationships and laws implicit in the data. So, we use deep learning-based methods to learn the truncation error E_{QF} in images and then analyze the effect of different quality factors on them.

According to Eq. 10, error E_{QF} is the fundamental cause of the block artifact of JPEG compression, so we hope to remove the influence of image content through deep learning to learn the distribution of the truncation error E_{QF}. For making the network learn the error E_{QF} directly from the input compressed image I_{QF}, we design the loss function as fellow

$$L_1(\theta_G) = \sum_{i=1}^{N} \|G(I_{QF}[i];\theta_G) - (I_{QF}[i] - I[i])\|_F^2, \qquad (14)$$

where N indicates the batch size of the input. In JPEG compression, since the RGB image is converted to YCbCr color space and then compressed in separate channels, and the Y luminance channel contains the most information, we select the Y channel for learning the compression error E_{QF}.

Finally, the network can learn the distribution of the error E_{QF} with different QF and extract JPEG compression block artifacts directly from the images. In Figure 3. (a2-a4), we compress an image I by using different QF and subtracting the compressed images I_{QF} in the green boxes from the uncompressed image in the red box. Then we extract block artifacts from the compressed images I_{QF} shown in Figure 3. (b2-b4), we can see that $G(I_{QF})$ owns the same distribution as E_{QF}.

By analyzing Figure 1, we know know that the block artifacts extracted from the seam-carving images have the corrupted 8×8 blocks distribution; however, we can't observe such a weak difference by the human eye, so we have to utilize some feature extraction tools. The blocks characteristics matrix (BACM) [44] calculated the differences within a block and spanning across a block boundary, which can measure the symmetrical property of the 8×8 block artifacts introduced by JPEG compression. To verify that seam-carving does have serious corruption to block artifact properties, we randomly select 500 images from the Alaska [45] dataset, and for every ten images of them, we use a quality factor $QF = \{90, 70, 50\}$ for JPEG compression. Next, we perform seam-removal and seam-insertion with a tampering rate (TR) of 1%, which means reducing or enlarging the width of the image size by 1%, then extract the block artifacts in the JPEG images through the network and use BACM to extract the characteristics matrix. From Figure 3. (a), we can see that the characteristics matrix of the JPEG images shows a diagonal symmetry, and when the QF value is smaller, the symmetrical property is more obvious. In contrast, the properties of the seam-removal and seam-insertion images disappeared. The results demonstrated that seam-carving is corrupt to 8×8 blocks on the JPEG image, and the proposed network structure and training strategy can capture this property very well.

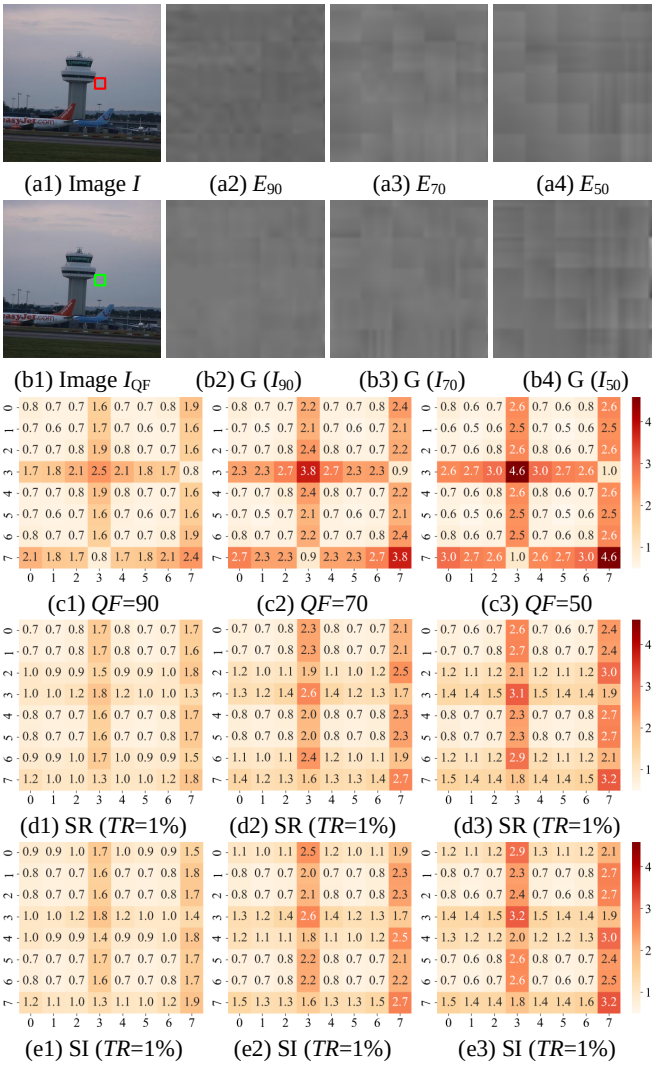


Figure 3: The true and learned JPEG compression traces, and the heat maps of the extracted 8×8 BACM characteristics matrix. The image I was compressed to obtain the compressed image $I_{QF=90,70,50}$. (a2-a4) are the truncation errors $E_{QF=90,70,50}$ generated by subtracting the red box in the image I from the green box in I_{QF}. (b2-b4) are the learned JPEG compression trace $J_{QF=90,70,50}$ from the corresponding JPEG images $I_{QF=90,70,50}$. (c1-c3) are the heat maps for images with QF=90, 70, and 50, respectively. (d1-d3) are the heat maps for QF=90, 70, and 50 at the tampering rate of 1% seam-removal. (e1-e3) are the heat maps for QF=90, 70, and 50 at the tampering rate of 1% seam-insertion.

As shown in Figure 1. (a), if we randomly remove and insert a seam from the JPEG image, all the 8×8 blocks distributions of the blocks that the seam passes through will corrupt, resulting in misalignment. If we give a little guidance to the network, it will learn the regularity of the seam positions quickly. Therefore, we record the coordinates of each seam when we use the seam-carving technique [1] to scale the images, then mark these coordinates. But there is a problem: for seam insertion, we can mark the inserted seam coordinates one by one, while for seam removal, we cannot mark the removed seams, so we mark both the left and right neighbor coordinates. In this way, we get the seam-carving forgery and its corresponding seams location map in pairs, and we input both images to guide the network in localizing the seams in the corrupted blocks. Our training goal is to minimize the loss between the network output location map and the ground truth M, which is defined as follows:

$$L_2(\theta_G) = \sum_{i=1}^{N} \|G(I_S[i]; \theta_G) - M[i]\|_F^2, \tag{15}$$

where N is the number of input pairs; $G(I_S[i]; \theta_G)$ and $M[i]$ are the i_{th} network output localization map and ground truth map, respectively.

4 Experiments

We propose a seam-carving localization method based on the block artifacts and conduct extensive experiments to evaluate the feasibility and effectiveness. In this section, we provide implementation details, experimental analysis, and object removal detection based on seam-carving.

4.1 Dataset

In the experiment, we use several datasets for the three different tasks. To extract JPEG compression block artifacts, we randomly select 200 TIFF-formated images from the ALASKA [45] dataset and randomly crop 12800 48×48 patches. We combine horizontal flips and rotations (90, 180, and 270 degrees) for data augmentation, increasing them to 76,800 patches as our training set.

For the seam-carving localization, Uncompressed Color Image Database (UCID) [46] and UCUS [15] datasets are used for training and testing. We randomly selected 1024 images from the UCID data set as the training set, and the remaining 314 images combined with the UCUS dataset to form the test set, a total of 1323 images. In the experiment, we mention different tampering rates (TR) of seam-removal and seam-insertion. Take 1% as an example, suppose the image size is 512×384 pixels, 1% vertical seam removal (insertion) rate means the width size was reduced (enlarged) by 1% resulting in the image size becomes 507×384 (517×384) pixels.

For object removal forgery localization, since we use seam-carving to remove the object from the pristine image requires the mask corresponding to the object, we create an object removal dataset (OR-COCO) using the COCO dataset [47], which is a dataset with masks for multiple tasks, such as segmentation, localization captioning, etc., the forged images automatically created by removing the object corresponding to the masks based on seam-removal.

4.2 Implementation Details

For the extraction of JPEG compression block artifacts, the network's batch size is set to 128, and patches within a batch randomly select the quality factor QF in [50, 100] for compression. The Adam optimizer was used with a learning rate of 0.001.

The batch size was set to 4 for the seam-carving localization, and the optimizer was still Adam with the seam learning rate. Because the network has already learned the block artifacts property, which provides good features for seam-carving

localization, the network can achieve good localization results by several epochs.

4.3 Evaluation Metrics

We select several evaluation metrics to illustrate the proposed method's effectiveness better. To extract JPEG compression block artifacts, we used peak signal-to-noise ratio (*PSNR*) to measure whether the network can extract correct block artifacts. If JPEG compression errors E_{QF} can be effectively extracted, according to Eq. 10, we can improve the quality of the image by removing E_{QF} and improving *PSNR*. The compressed image I_{QF} becomes I' after removing the blocks, and then the PSNR value of I' and the original image I can be calculated by Eq. 16 and Eq. 17.

$$MSE = \frac{1}{h \times w} \sum_{i=0}^{h-1} \sum_{j=0}^{w-1} [I(i,j) - I'(i,j)]^2, \tag{16}$$

where *MSE* means Mean square error, h, w denotes the height and width of the image, respectively.

$$PSNR = 20 \cdot \log_{10} \left(\frac{255}{\sqrt{MSE}} \right). \tag{17}$$

We can regard the seam-carving localization output as a binary classification. Pixels belong to the pristine (negative) or seams (positive) class. All performance metrics rely on four basic values, TP (true positive): positive pixels predicted positive; TN (true negative): negative pixels predicted negative; FP (false positive): negative pixels predicted positive; FN (false negative): positive pixels predicted negative. However, there are many more negative than positive pixels for seam-carving localization, and the incorrectly predicted positive pixels affect very little on accuracy. To emphasize the positive class, we used *Precision* (P) and *Recall* (R) to avoid false alarms, *Precision* is the ratio of the correctly detected areas to all detected regions, and *Recall* is the ratio of the correctly detected areas to the tampered regions in the ground truth, defined as:

$$Precision = \frac{TP}{TP+FP}, \; Recall = \frac{TP}{TP+FN}. \tag{18}$$

Precision and *Recall* are both indicators for evaluating model performance, but their focus is different, so the two quantities are summarized in a single index by their harmonic mean, the $F1$ measure

$$F1 = \frac{2 \times Precision \times Recall}{Precison + Recall} \times 100\%, \tag{19}$$

To assess performance effectively, we consider all three measures: *Precision*, *Recall*, and $F1$.

4.4 Performance of Seam-carving Localization

This section presents the localization results of seam-removal and seam-insertion in JPEG format. The proposed method performs pixel-level seam localization and achieves satisfactory results. We train and test the seam-removal and seam-insertion with tampering rates of 1%, 2%, 5%, 10%, and their mixtures (M%) under the quality factors of 90, 70, and 50. The localization results of seam-removal and seam-insertion are recorded in Figure 4. In Figure 4, we give the P, R, and $F1$ values for seam-removal and seam-insertion localization. The colors indicate training on different tampering rate datasets, and the horizontal coordinates indicate the tampering rate of the test set.

By comparing the localization results of seam-removal and seam-insertion, we can conclude that the precision of seam-removal localization is higher and more stable than seam-insertion, and the training set greatly influences the localization result. On this issue, seam-carving and seam-insertion have their characteristics. For the seam-removal, training with the tampering rate of 10% (black) data will achieve better localization results, while the tampering rate of 2% (red) is more appropriate for the seam-insertion. For different tampering rates, when the tampering rate is low, the network learns less knowledge from the samples, so the metric of localization is relatively low. However, when the number of seams is too large, the 8×8 block distribution of the JPEG images is severely broken, and some basic knowledge cannot be learned.

From the experimental results, we can see that the proposed method can obtain effective pixel-level localization results using a small amount of train data. For different compression quality factors, even when the quality factor is 90, the proposed method can still accurately detect the vast majority of the area of the seams, which shows that the network can effectively extract block artifacts from JPEG images. The seams can be easily localized when the distribution of the 8×8 block is corrupted. Although the proposed method is not a classification task, the high-precision localization results allow us to determine whether an image has been seam carved, even if there are only a few seams. A better insight into the actual quality of results can be gathered by the visual inspection in Figure 5.

Table 1: The cross-validation $F1$ values across different quality factors.

test \ train	90	70	50	QF mixed
Seam-removal				
90	74.26	54.17	40.81	74.62
70	77.64	85.58	82.04	82.21
50	77.73	85.07	85.89	82.81
QF mixed	76.55	74.94	69.62	79.87
Seam-insertion				
90	73.59	69.51	59.84	77.13
70	70.18	77.13	73.88	75.22
50	67.11	73.98	74.54	71.76
QF mixed	70.29	73.54	69.42	74.71

To verify the generalization of the proposed seam-carving localization method, we perform cross-validation on different quality factors. For seam-removal, we use data with the tampering rate of 10% as the training set, then we fix the quality factor and test on datasets containing different quality factors,

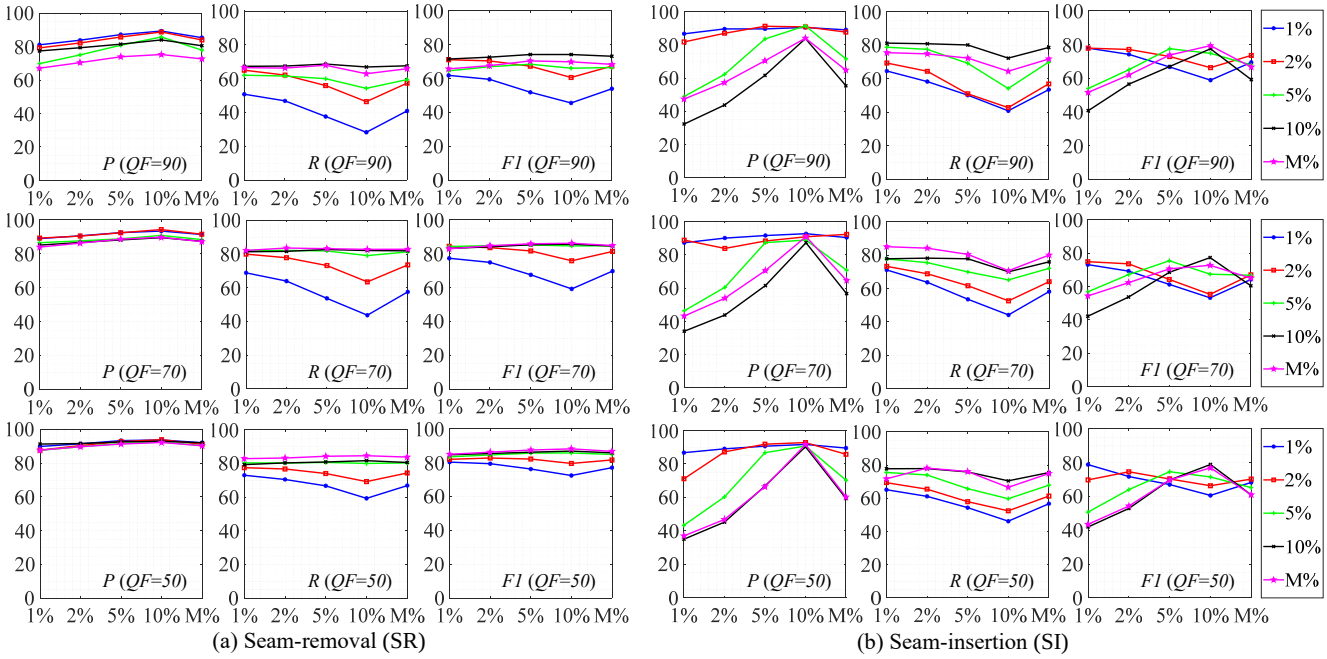


Figure 4: The *Precision*, *recall*, and $F1$ values trained on seam-removal and seam-insertion datasets with different tampering rates. We trained and tested with three different compression quality factors: 90, 70, and 50. Each line indicates the training using the dataset with that tampering rate, the horizontal coordinates indicate the test results on tampering rates of 1%, 2%, 5%, and 10%, and M% indicates the mixture of these four tampering rates.

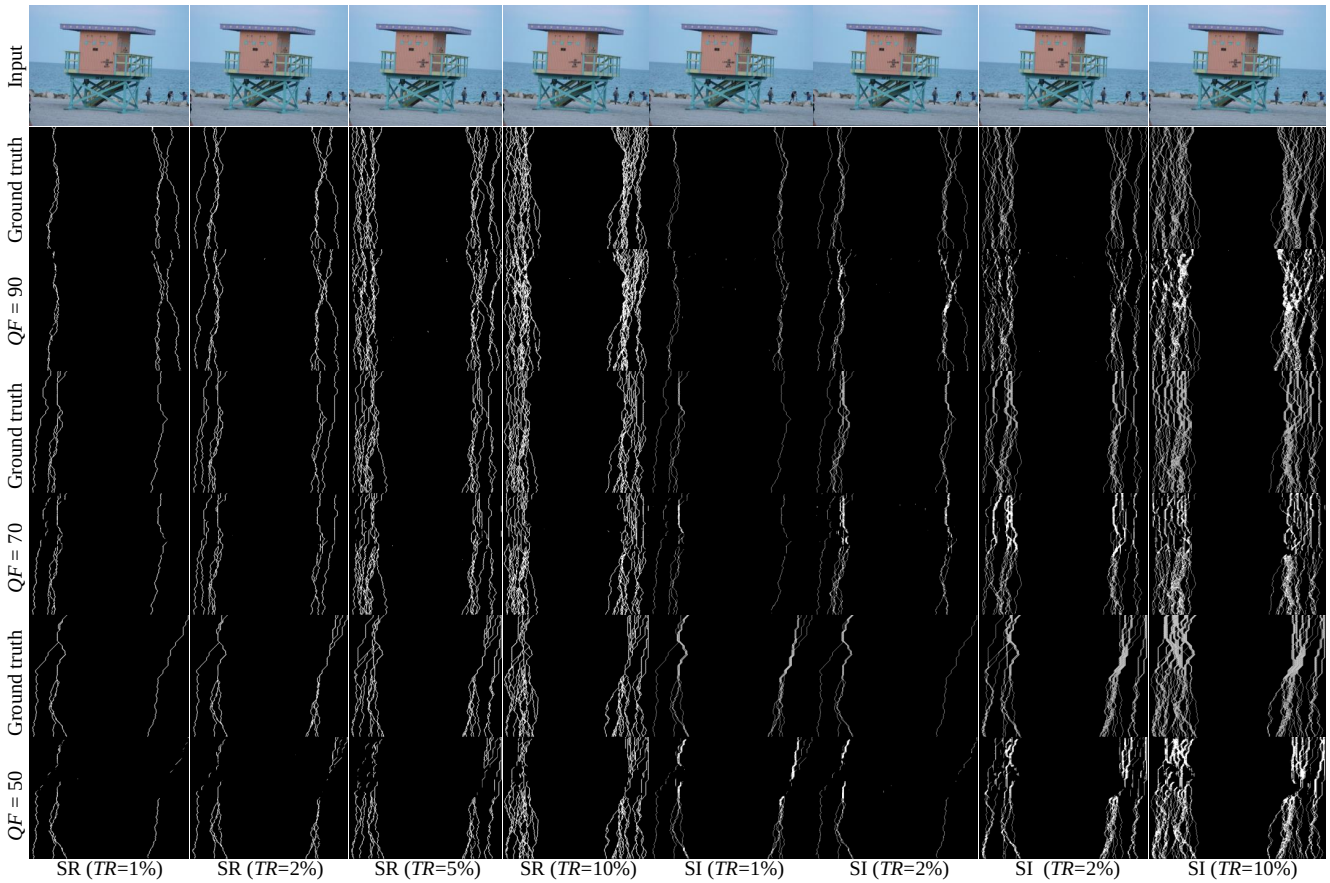


Figure 5: Localization results for seam-removal and seam-insertion with four tampering rates of 1%, 2%, 5%, and 10% under three compression quality factors of 90, 70, and 50, we test the situation with different tampering rates and different quality factors.

while for seam-insertion, we use the dataset with the tampering rate of 2% for training to perform the same test, the experimental results are recorded in Table 1. We can conclude from the crossover experimental results that the 8×8 block artifact distribution of JPEG images extracted by the network has a strong generalization property. Even if only one quality factor of the block artifacts is learned, the network can apply to other situations.

4.5 Ablation Study

To verify the effectiveness of the proposed network structure and training strategy, we perform many ablation experiments for the extraction of block artifacts and the localization of seam-carving, respectively. For JPEG block artifact extraction, according to Eq. 10 and Eq. 7, a JPEG image I_{QF} can improve image quality by removing blocking artifacts (Image restoration) if the network can achieve the extraction task effectively.

We randomly select 500 images from the Alaska [45] dataset. For every ten images, we use a quality factor $QF \in [50, 100]$ for JPEG compression; then we calculate the *PSNR* value of the compressed image and the original image, respectively. Next, we input the compressed images into the network to extract their blocking artifacts. After that, we subtract the corresponding blocking artifacts from the compressed image for image restoration. Finally, we calculate the *PSNR* value of the restore and the original images. The experimental results are shown in Figure 6. (a), we give an example in Figure 6 for better understanding. (c), where (c2) was the JPEG compressed image, (c3) was the extract block artifacts of the network, and (c4) was the restored result of (c2). We can see that after image restoration, the influence of block artifacts is effectively eliminated, and the image quality improves, demonstrating that the network can extract the JPEG compression error E_{QF} well.

We also demonstrate the effectiveness of the network structure, as shown in Figure 6. (b), where we test the performance of the network after removing different structures. The baseline means that we only use the convolution operation in the $RA(\cdot)$ block, which guarantees the network's computational power and allows for a fairer comparison. Compared with the baseline, we can conclude that residual structure and attention mechanisms can help the network learn better features, which facilitates a decrease in the loss function and achieves higher *PSNR* values. In Figure 7. (a-d), we show a seam-insertion image with a tampering rate of 2% and its ground truth, in Figure 7. (e), we directly input the seam-carving image and its corresponding seams location map in pairs to train the network for localization in Figure 7. (f), we perform seam-carving localization based on the extraction of the block artifacts. We can see that without the extraction of block artifacts, it is difficult for the network to find the location of the seams because the network is more likely to pay attention to the semantic information rather than the corrupted distribution of the 8×8 blocks. By contrast, after extracting the JPEG compression block artifacts, the network easily finds the corrupted blocks guided by the seams location map, and just after one epoch, the network quickly localized the position of the seams, which demonstrates the correctness and feasibility of our theory. As the epoch increases, seam localization becomes more precise. Compared with the baseline, adding skip-connection can reuse the feature map of the shallow layers, which makes the network training more stable; adding the attention mechanism can learn the different weights of the features according to the loss, motivate the seams' features, suppress unimportant background information, and improve the performance of seam localization. In Figure 8.(b), we increased the number of $RA(\cdot)$ blocks from four to seven, and the number to six provided a significant localization result on the seam-removal and seam-insertion datasets and got $F1$ values of 79.87 and 74.71, respectively. Still, there was no subsequent significant improvement for more numbers. Therefore, we choose six $RA(\cdot)$ blocks as our final network structure.

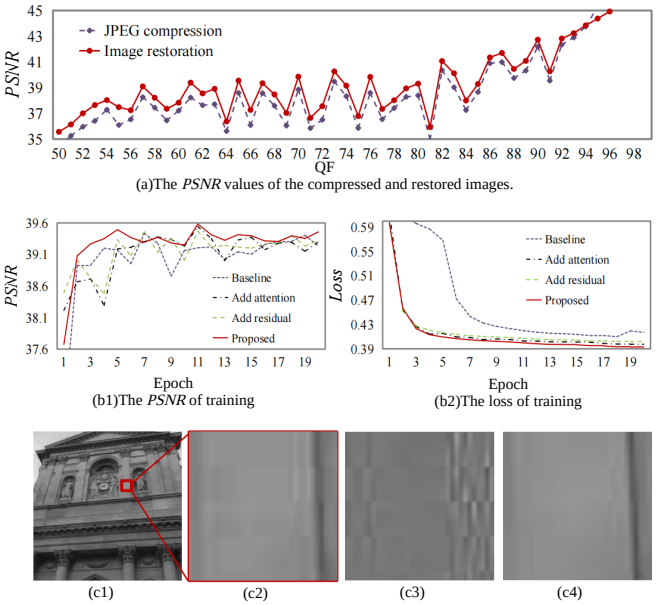
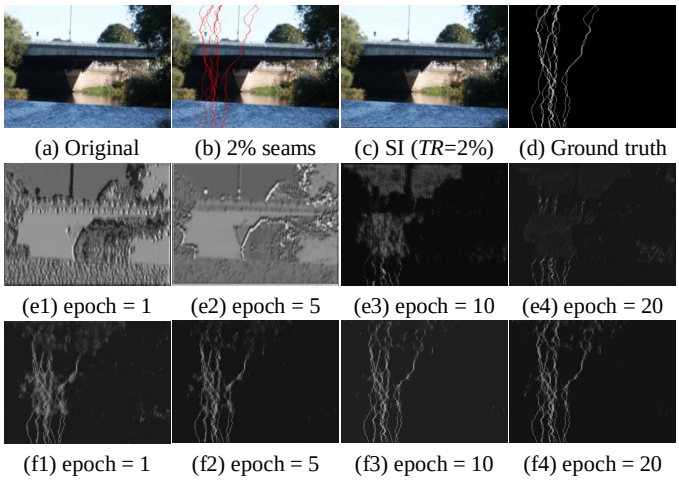


Figure 6: The results of image restoration. (a) The *PSNR* values of the compressed images (purple dots) and restored images (red dots) with different QF*s*. (b) The network's performance after removing different structures (b1-b2) are the *PSNR* values and losses on the test set when the epoch is [0, 20], respectively. (c1) The image after JPEG compression. (c2) The regions in the red box of (c1). (c3) The output of extracted block artifacts. (c4) The restored result of (c2).

4.6 Performance for Unseen Cases

When we perform seam-carving localization in reality, there are various unseen cases, such as unseen compression and tampering rates. It is necessary to maintain good performance for those forensics. To demonstrate the generality of the proposed method, we performed extensive experiments under several scenarios. We used the model train on the different QF_s datasets in the cross-validation experiment for the unseen cases test.



(a) Original (b) 2% seams (c) SI (*TR*=2%) (d) Ground truth

(e1) epoch = 1 (e2) epoch = 5 (e3) epoch = 10 (e4) epoch = 20

(f1) epoch = 1 (f2) epoch = 5 (f3) epoch = 10 (f4) epoch = 20

Figure 7: The ablation study of the extraction of the block artifacts (w/o BA). (a) The original image. (b) The image with 2% seams before removal. (c) The image of 2% seam-removal. (d) The ground truth is the location map of the removed seams. (e) We only performed seam-carving localization without the extraction of the block artifacts. (f) We performed seam-carving localization based on the extraction of the block artifacts.

4.6.1 Unseen Compression

To simulate the unseen compression cases, we consider the cases of single and double JPEG compression. For the former, we randomly select a quality factor $QF \in [50, 100]$ to compress the test dataset; for the latter, because there are tens of thousands of combinations, it is impossible to simulate every case, so we perform a random compression again based on the single one. The experimental results are recorded in Table 2.

Table 2: The $F1$ values evaluation for seam-carving localization under unseen compression cases.

test \ train	90	70	50	QF mixed
Seam-removal				
Single compression	73.62	70.39	69.81	76.22
Double compression	77.61	82.46	79.68	81.39
Seam-insertion				
Single compression	70.58	73.08	69.59	74.84
Double compression	70.98	75.68	72.75	75.51

The experimental results show that the proposed method has good localization ability for unseen compression cases, is not only applicable to single compression, but also maintains stable localization ability for double cases. Because seam-carving corrupting the 8×8 block artifacts distribution in JPEG images is a common phenomenon, the knowledge and capabilities learned by the network generalize well across different compressions. Moreover, we can see from Table 2 that the localization results of double cases are better than single compression. Because we perform JPEG compression again based on the single one, the compression degree is greater, and the image's block artifacts are more obvious. This phenomenon demonstrates the proposed seam-carving localization method based on corrupted block artifacts.

4.6.2 Unseen Tampering Rates

We also test the localization performance of the proposed method for the seam-removal and seam-insertion at compression quality factors of 90, 70, and 50 with different tampering rates, which included smaller tamper rates of 2% and 4%, and larger tampering rates of 8% and 15%, the experimental results are recorded in Table 3.

The experimental results show that the proposed method can achieve good localization results in the cases of different tampering rates. Unlike unseen compression cases, unseen tampering rates have a more significant impact on seam localization performance, especially seam-insertion. The number of seams in the image increases as the tampering rate increases, however, compared to the seam-removal, the number of seams in the seam-insertion is high and more difficult to localize, so there is a decrease in the performance.

Table 3: The $F1$ values evaluation for seam-carving on unseen tampering rates.

TR \ QF	90	70	50	QF mixed
Seam-removal				
2%	73.06	83.91	85.37	76.64
4%	73.96	85.21	86.09	78.05
8%	74.96	85.68	86.62	79.57
15%	73.31	85.21	87.05	79.67
Seam-insertion				
2%	73.85	76.97	74.74	74.87
4%	70.84	74.11	71.48	69.01
8%	65.45	68.74	67.85	64.09
15%	60.31	61.24	63.66	60.47

4.7 Comparison Experiments

Seam-carving localization has been studied extensively. Previous work [33] divides images into patches with stride 64 for patch-level classification, achieving low localization accuracy. [34] proposes localization using 512×512 patches, improving precision but requiring large training datasets and long training time. Recent work [19] introduces SPNet with spatial-phase learning for seam carving detection, showing improved performance but producing band-like localization results rather than precise seam-level detection. We train and test seam-removal and seam-insertion with tampering rates of 1%, 2%, 5%, 10%, and mixtures (M%) under quality factors 90, 70, and 50, with results in Table 4.

Experimental results show [33] localization is unreliable, only vaguely marking seam regions. [34] performs better but has low precision at negligible tampering rates and significantly decreased precision when QF becomes large. [19] achieves moderate performance but does not provide precise seam localization, producing band-like regions.These methods require extensive training data and must crop images into patches due to different sizes between original, seam-removal, and seam-insertion images, increasing computation and limiting global information access.

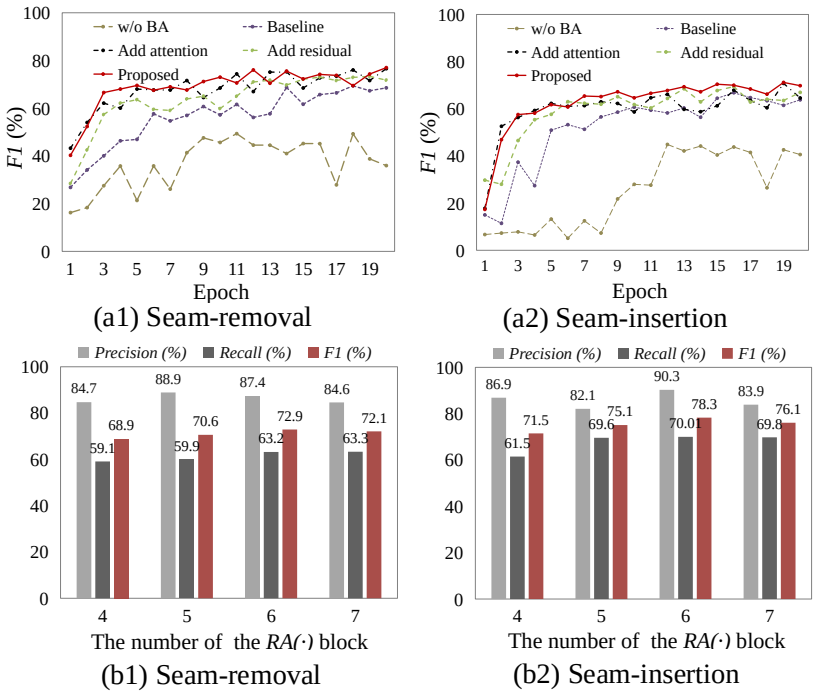


Figure 8: The seam-carving forgery localization performance of several ablated versions. (a) The variation of $F1$ values for the training process of the network with different ablated versions. (b) The effect of different numbers of $RA(\cdot)$ blocks on the *Precision*, *Recall*, and $F1$ values of seam-carving localization

Table 4: Comparative $F1$ values of three seam-carving localization methods.

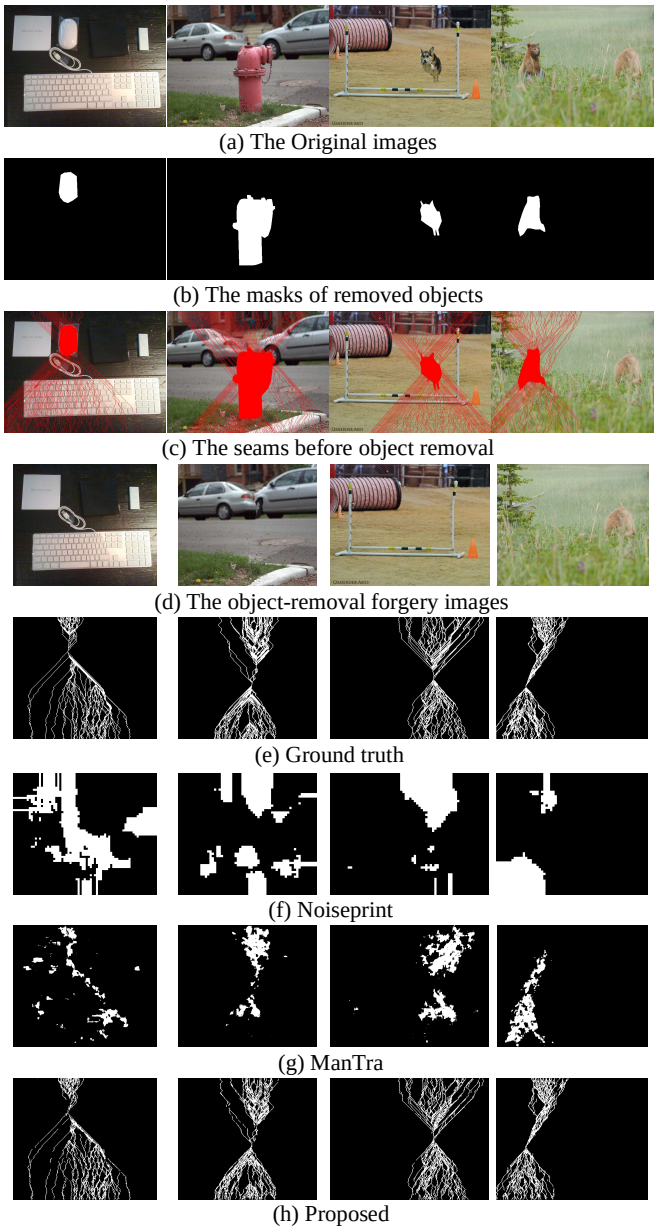
QF	QF=50				QF=70				QF=90			
Methods	[33]	[34]	[19]	Proposed	[33]	[34]	[19]	Proposed	[33]	[34]	[19]	Proposed
SR 1%	6.9	20.43	8.78	**80.31**	3.42	4.02	11.01	**77.26**	3.31	3.16	12.91	**61.93**
SR 2%	8.4	60.62	14.43	**82.81**	5.59	16.34	17.67	**83.56**	5.76	5.74	18.30	**70.44**
SR 5%	10.74	69.17	22.40	**85.67**	11.14	44.79	24.68	**85.14**	11.82	22.08	27.33	**68.49**
SR 10%	20.97	71.76	32.74	**86.89**	18.71	62.54	35.79	**85.58**	19.78	47.38	36.84	**74.26**
SR M%	15.99	57.24	16.68	**86.73**	10.16	37.62	21.54	**84.83**	10.16	42.73	22.98	**68.42**
SI 1%	3.86	66.68	12.77	**78.84**	2.22	59.58	10.75	**77.91**	2.08	21.85	11.31	**73.13**
SI 2%	6.99	69.67	13.94	**74.54**	4.43	66.97	15.73	**77.13**	4.64	43.08	15.65	**73.59**
SI 5%	14.61	71.64	19.94	**74.64**	11.16	71.09	20.43	**77.56**	10.42	58.21	20.02	**75.46**
SI 10%	27.06	73.04	25.71	**78.93**	21.85	69.31	25.93	**77.55**	20.11	63.35	25.09	**77.33**
SI M%	15.41	35.47	19.28	**61.01**	9.72	58.87	15.05	**66.74**	9.21	26.93	19.29	**65.03**

4.8 Object removal Localization

Seam-carving is a feature in popular image editing software such as ImageMagic and Photoshop. The ease of access to these programs makes it easy to forge images. Seam-carving can easily remove entire objects and leave a large percentage of pixel values untampered, which poses a challenge to image forgery detection. When seam-carving is applied to remove the object from the image, it may lead to the distortion of the information transmitted by the image, misleading the public.

Object removal based on seam-carving requires us to provide the corresponding mask for the object, so we create the object removal dataset OR-COCO based on the COCO dataset. The COCO dataset is available for multiple tasks, such as segmentation, localization captioning, etc. The COCO dataset contains 80 categories, such as a person, a bicycle, a car, and so on, and the size of most objects is suitable for removal operations. We select ten images and their corresponding masks from each class and then artificially remove some targets that were too large, so we obtain 625 unknown JPEG compressed images and the mask of an object in the figure. Because the seam-carving algorithm determines the seams by finding pixels with low energy, we could interfere with the energy function to determine the seams by assigning the energy of object pixels to a low value, such that the seams forcibly pass through the object marked for removal. To facilitate the observation, we record the locations of

these seams. Finally, we complete the production of the OR-COCO dataset, which includes the original images, the object removal images, and the seam location images.



(a) The Original images

(b) The masks of removed objects

(c) The seams before object removal

(d) The object-removal forgery images

(e) Ground truth

(f) Noiseprint

(g) ManTra

(h) Proposed

Figure 9: The detection results in object removal operated by seam-carving. Noiseprint and Mantra are two state-of-the-art forgery detection methods that have good detection ability for traditional object removal forgery.

By observing the distribution and position of the seams in the OR-COCO dataset, we find some rules for object removal operated by seam-carving. As shown in Figure 9, when we illegally remove the object from the image, the seams are forced to pass through the object, which forms an apparent crossover region, even if the object is removed, the crossover region formed by the seam still exists in the forgery image. By contrast, if we reduce the image to the same size by the ordinary seam-removal operation, the seams selected by minimizing the energy function show a relatively uniform distribution that avoids damaging the image content as much as possible. The particular crossover region in the object removal image provides clues for our detection.

For the traditional object removal method, after removing an object, an inpainting algorithm will fill the left blank region. Based on the research of the inpainting algorithm, many effective detection methods have been proposed [21, 22, 48]. By contrast, object removal operated by seam-carving does not leave a significant padding area on the image. In comparison experiments, we select two methods that represent the state-of-the-art in traditional forgery detection and have good object removal detection ability. Noistprint [49] believes that different cameras leave inconsistent noise on output images, and ManTra [50] uses semantic segmentation to locate various local forgeries. Through the detection results of the two methods, we can demonstrate the difficulty of object removal detection operated by seam-carving, the experimental results shown in Figure 9.

We can see that general local forgery detection methods cannot achieve good results for object removal forgeries operated by seam-carving; they can notice some regions of removed seams, but this is not enough to detect and localize the position of the removed object. Compared with these methods, the proposed method can localize the position of the removed seams in the forgery image at the pixel level. Then, we can determine whether the image has undergone illegal object removal and localize the location of the removed object.

5 Discussion

In this paper, we design a multi-block network structure and propose an effective training strategy to achieve pixel-level seam-carving localization by using a small amount of training data. Compared with the existing studies, the proposed method is no longer satisfied with the classification of seam-carving images but finds the location of seams, which has more practical value. The proposed method also has the limitation that we relied on the block artifact property, but there are still some images not experience JPEG compression, for this case, we need to explore other image properties to assist us in achieving seam-carving localization. In addition, the proposed training strategy makes it easier for the network to capture the weak trace of seams rather than the semantics or colors in the image. The training strategy also applies to other image forensic tasks, such as image local tampering detection, where we only need to change the input and its corresponding ground truth of the network. Because if we can find the innate character of the local forgery, it will be easy to localize the tampered regions quickly and accurately.

6 Conclusion

Seam-carving is an image re-targeting technique with good performance, but it also provides a tool for criminals to create fake images. Although the seam-carving image does not appear to change to the human eye, it still inevitably destroys some image properties. Through theoretical analysis of seam-caring and JPEG compression, we find that block artifact of the JPEG compression provides a good opportunity for us to localize seam-caring. To capture the trace of the seams,

we design a multi-block network structure to enhance the learning ability, which combines the residual structure and attention mechanisms, and we propose an effective training strategy to localize the seams in images. First, we extract the properties hidden in the image self-supervised; second, we input the location map of the seams as guidance to localize the seams from the corrupted properties. As we expected, the network can quickly localize the seams from the corrupted properties without a large amount of training data. Extensive experiments verify the feasibility and effectiveness of the proposed method. Based on seam-carving localization, we detect object removal operated by seam-carving, turning the theoretical assumptions into reality.

In future work, we will mainly focus on two aspects: to mine more image properties in digital images and use multiple properties to localize seam-carving; the other is to extend the proposed training strategy to solve more image forensic tasks.

Funding

This work is supported by the Natural Science Foundation of Chongqing under Grant CSTB2023NSCQ-MSX0341.

Author Contributions

Writing-original draft, Xiaoyi Wang; data curation, Bo Liu;writing-review and editing, Xiuli Bi; data curation, Bin Xiao. All authors have read and agreed to the published version of the manuscript.

Conflict of Interest

All the authors declare that they have no conflict of interest.

Data Available

The data and materials used in this study are available upon request from the corresponding author.

References

[1] Avidan, S., Shamir, A.: Seam Carving for Content-Aware Image Resizing, in Seminal Graphics Papers: Pushing the Boundaries, Volume 2, 1st edn. Association for Computing Machinery, New York, NY, USA (2023)

[2] Mujahid, M., Khan, A., Hossain, M.S., Laulai, A., Javeed, M.A.: Reinforcement learning based improved seam carving using special point and alpha value for optimal content preservation. In Magnenat-Thalmann, N., Kim, J., Sheng, B., Deng, Z., Thalmann, D., Li, P. (eds.) Advances in Computer Graphics, pp. 30–41. Springer, Cham (2025). https://doi.org/10.1007/978-3-031-82024-3_3

[3] Mujahid, M., Hossain, M.S., Khan, A., Huang, Z.: Seam carving empowered by reinforcement learning for optimal content preservation. In Magnenat Thalmann, N., Hu, X., Sheng, B., Thalmann, D., Peng, T., Meng, W., Huang, J., Zhu, L., Wei, X. (eds.) Computer Animation and Social Agents, pp. 441–455. Springer, Singapore (2025). https://doi.org/10.1007/978-981-96-2684-7_31

[4] Su, H., Ye, Z., Liu, Y., Yu, S.: Seam carving based on dynamic energy regulation. Multimedia Tools and Applications **82**(17), 25795–25810 (2023) https://doi.org/10.1007/s11042-023-14516-9

[5] Ayubi, J., Amirani, M.C., Valizadeh, M.: A new seam carving method for image resizing based on entropy energy and lyapunov exponent. Multimedia Tools and Applications **82**(13), 19417–19440 (2023) https://doi.org/10.1007/s11042-022-13823-x

[6] Liu, D., Yu, Y., Tan, L., Wu, W., Zhao, B., Li, Z., Lu, B., Wen, Y.: Seamcarver: Llm-enhanced content-aware image resizing. Preprints (2024) https://doi.org/10.20944/preprints202412.1897.v1

[7] Shehu, A., Mati, K., Natowicz, R.: A hybrid method for robust malignancy-benignancy prediction in mammograms: Seam carving and u-net for cnns. In 2024 International Conference on Computing, Networking, Telecommunications Engineering Sciences Applications (CoNTESA), pp. 68–73 (2024). https://doi.org/10.1109/CoNTESA64738.2024.10891287

[8] Liu, Q., Chen, Z.: Improved approaches with calibrated neighboring joint density to steganalysis and seam-carved forgery detection in jpeg images. ACM Trans. Intell. Syst. Technol. **5**(4) (2015) https://doi.org/10.1145/2560365

[9] Rubinstein, M., Shamir, A., Avidan, S.: Improved seam carving for video retargeting. ACM Trans. Graph. **27**(3), 1–9 (2008) https://doi.org/10.1145/1360612.1360615

[10] Achanta, R., Süsstrunk, S.: Saliency detection for content-aware image resizing. In 2009 16th IEEE International Conference on Image Processing (ICIP), pp. 1005–1008 (2009). https://doi.org/10.1109/ICIP.2009.5413815

[11] Lu, W., Wu, M.: Seam carving estimation using forensic hash. In Proceedings of the Thirteenth ACM Multimedia Workshop on Multimedia and Security. MMSec '11, pp. 9–14. Association for Computing Machinery, New York, NY, USA (2011). https://doi.org/10.1145/2037252.2037255 . https://doi.org/10.1145/2037252.2037255

[12] Seung-Jin RYU, H.-K.L. Hae-Yeoun LEE: Detecting trace of seam carving for forensic analysis. IEICE TRANSACTIONS on Information **E97-D**(5), 1304–1311 (2014) https://doi.org/10.1587/transinf.E97.D.1304

[13] Yin, T., Yang, G., Li, L., Zhang, D., Sun, X.: Detecting seam carving based image resizing using local binary patterns. Computers Security **55**, 130–141 (2015) https:

//doi.org/10.1016/j.cose.2015.09.003

[14] Lu, M., Niu, S.: Detection of image seam carving using a novel pattern. Computational Intelligence and Neuroscience **2019**(1), 9492358 (2019) https://doi.org/10.1155/2019/9492358

[15] Wattanachote, K., Shih, T.K., Chang, W.-L., Chang, H.-H.: Tamper detection of jpeg image due to seam modifications. IEEE Transactions on Information Forensics and Security **10**(12), 2477–2491 (2015) https://doi.org/10.1109/TIFS.2015.2464776

[16] Liu, Q.: Exposing seam carving forgery under recompression attacks by hybrid large feature mining. In 2016 23rd International Conference on Pattern Recognition (ICPR), pp. 1041–1046 (2016). https://doi.org/10.1109/ICPR.2016.7899773

[17] Liu, Q.: An approach to detecting jpeg down-recompression and seam carving forgery under recompression anti-forensics. Pattern Recognition **65**, 35–46 (2017) https://doi.org/10.1016/j.patcog.2016.12.010

[18] Nam, S.-H., Ahn, W., Yu, I.-J., Kwon, M.-J., Son, M., Lee, H.-K.: Deep convolutional neural network for identifying seam-carving forgery. IEEE Transactions on Circuits and Systems for Video Technology **31**(8), 3308–3326 (2021) https://doi.org/10.1109/TCSVT.2020.3037662

[19] Chen, J., Lv, Z., Jiao, G., Xia, M., Yang, G.: Sp-net: Seam carving detection via spatial-phase learning. Journal of Information Security and Applications **89**, 103963 (2025) https://doi.org/10.1016/j.jisa.2025.103963

[20] Wu, H., Zhou, J.: Iid-net: Image inpainting detection network via neural architecture search and attention. IEEE Transactions on Circuits and Systems for Video Technology **32**(3), 1172–1185 (2022) https://doi.org/10.1109/TCSVT.2021.3075039

[21] Li, H., Huang, J.: Localization of deep inpainting using high-pass fully convolutional network. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 8301–8310 (2019)

[22] Wang, X., Niu, S., Wang, H.: Image inpainting detection based on multi-task deep learning network. IETE Technical Review **38**(1), 149–157 (2021) https://doi.org/10.1080/02564602.2020.1782274 https://doi.org/10.1080/02564602.2020.1782274

[23] Park, J., Cho, D., Ahn, W., Lee, H.-K.: Double jpeg detection in mixed jpeg quality factors using deep convolutional neural network. In Proceedings of the European Conference on Computer Vision (ECCV) (2018)

[24] Tang, W., Li, B., Barni, M., Li, J., Huang, J.: Improving cost learning for jpeg steganography by exploiting jpeg domain knowledge. IEEE Transactions on Circuits and Systems for Video Technology **32**(6), 4081–4095 (2022) https://doi.org/10.1109/TCSVT.2021.3115600

[25] Li, W., Li, X., Ni, R., Zhao, Y.: Quantization step estimation for jpeg image forensics. IEEE Transactions on Circuits and Systems for Video Technology **32**(7), 4816–4827 (2022) https://doi.org/10.1109/TCSVT.2021.3123477

[26] Kwon, M.-J., Yu, I.-J., Nam, S.-H., Lee, H.-K.: Catnet: Compression artifact tracing network for detection and localization of image splicing. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), pp. 375–384 (2021)

[27] Rao, Y., Ni, J.: Self-supervised domain adaptation for forgery localization of jpeg compressed images. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp. 15034–15043 (2021)

[28] Ma, L., Zhao, Y., Peng, P., Tian, Y.: Sensitivity decouple learning for image compression artifacts reduction. IEEE Transactions on Image Processing (2024) https://doi.org/10.1109/TIP.2024.3403034

[29] Liao, X., Yin, J., Chen, M., Qin, Z.: Adaptive payload distribution in multiple images steganography based on image texture features. IEEE Transactions on Dependable and Secure Computing **19**(2), 897–911 (2022) https://doi.org/10.1109/TDSC.2020.3004708

[30] Butora, J., Puteaux, P., Bas, P.: Errorless robust jpeg steganography using outputs of jpeg coders. IEEE Transactions on Dependable and Secure Computing **21**(4), 2394–2406 (2024) https://doi.org/10.1109/TDSC.2023.3306379

[31] Zhou, Z., Su, Y., Li, J., Yu, K., Wu, Q.M.J., Fu, Z., Shi, Y.: Secret-to-image reversible transformation for generative steganography. IEEE Transactions on Dependable and Secure Computing **20**(5), 4118–4134 (2023) https://doi.org/10.1109/TDSC.2022.3217661

[32] Zhang, J., Chen, K., Qin, C., Zhang, W., Yu, N.: Aas: Automatic virtual data augmentation for deep image steganalysis. IEEE Transactions on Dependable and Secure Computing **21**(4), 3515–3527 (2024) https://doi.org/10.1109/TDSC.2023.3333913

[33] Nataraj, L., Gudavalli, C., Manhar Mohammed, T., Chandrasekaran, S., Manjunath, B.S.: Seam carving detection and localization using two-stage deep neural networks. In Gopi, E.S. (ed.) Machine Learning, Deep Learning and Computational Intelligence for Wireless Communication, pp. 381–394. Springer, Singapore (2021). https://doi.org/10.1007/978-981-16-0289-4_29

[34] Gudavalli, C., Rosten, E., Nataraj, L., Chandrasekaran, S., Manjunath, B.S.: Seetheseams: Localized detection

of seam carving based image forgery in satellite imagery. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, pp. 1–11 (2022)

[35] Liang, J., Cao, J., Sun, G., Zhang, K., Van Gool, L., Timofte, R.: Swinir: Image restoration using swin transformer. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops, pp. 1833–1844 (2021)

[36] Jiang, J., Zhang, K., Timofte, R.: Towards flexible blind jpeg artifacts removal. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp. 4997–5006 (2021)

[37] Li, J., Wang, Y., Xie, H., Ma, K.-K.: Learning a single model with a wide range of quality factors for jpeg image artifacts removal. IEEE Transactions on Image Processing **29**, 8842–8854 (2020) https://doi.org/10.1109/TIP.2020.3020389

[38] He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016)

[39] Roy, A.G., Navab, N., Wachinger, C.: Recalibrating fully convolutional networks with spatial and channel "squeeze and excitation" blocks. IEEE Transactions on Medical Imaging **38**(2), 540–549 (2019) https://doi.org/10.1109/TMI.2018.2867261

[40] Nam, S.-H., Ahn, W., Mun, S.-M., Park, J., Kim, D., Yu, I.-J., Lee, H.-K.: Content-aware image resizing detection using deep neural network. In 2019 IEEE International Conference on Image Processing (ICIP), pp. 106–110 (2019). https://doi.org/10.1109/ICIP.2019.8802946

[41] Boroumand, M., Chen, M., Fridrich, J.: Deep residual network for steganalysis of digital images. IEEE Transactions on Information Forensics and Security **14**(5), 1181–1193 (2019) https://doi.org/10.1109/TIFS.2018.2871749

[42] Zhang, K., Zuo, W., Chen, Y., Meng, D., Zhang, L.: Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. IEEE Transactions on Image Processing **26**(7), 3142–3155 (2017) https://doi.org/10.1109/TIP.2017.2662206

[43] Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Bach, F., Blei, D. (eds.) Proceedings of the 32nd International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 37, pp. 448–456. PMLR, Lille, France (2015). https://proceedings.mlr.press/v37/ioffe15.html

[44] Luo, W., Qu, Z., Huang, J., Qiu, G.: A novel method for detecting cropped and recompressed image block. In 2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07, vol. 2, pp. 217–220 (2007). https://doi.org/10.1109/ICASSP.2007.366211

[45] Ruiz, H., Chaumont, M., Yedroudj, M., Amara, A.O., Comby, F., Subsol, G.: Analysis of the scalability of a deep-learning network for steganography "into the wild". In Del Bimbo, A., Cucchiara, R., Sclaroff, S., Farinella, G.M., Mei, T., Bertini, M., Escalante, H.J., Vezzani, R. (eds.) Pattern Recognition. ICPR International Workshops and Challenges, pp. 439–452. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-68780-9_36

[46] Schaefer, G., Stich, M.: Ucid: An uncompressed color image database. In Storage and Retrieval Methods and Applications for Multimedia 2004, vol. 5307, pp. 472–480 (2003). SPIE

[47] Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) Computer Vision – ECCV 2014, pp. 740–755. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10602-1_48

[48] Li, H., Luo, W., Huang, J.: Localization of diffusion-based inpainting in digital images. IEEE Transactions on Information Forensics and Security **12**(12), 3050–3064 (2017) https://doi.org/10.1109/TIFS.2017.2730822

[49] Cozzolino, D., Verdoliva, L.: Noiseprint: A cnn-based camera model fingerprint. IEEE Transactions on Information Forensics and Security **15**, 144–159 (2020) https://doi.org/10.1109/TIFS.2019.2916364

[50] Wu, Y., AbdAlmageed, W., Natarajan, P.: Mantra-net: Manipulation tracing network for detection and localization of image forgeries with anomalous features. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9543–9552 (2019)