Self-supervised Domain Adaptation for Forgery Localization of JPEG Compressed Images

Yuan Rao[1], Jiangqun Ni[2,3]
[1]School of Electronics and Information Technology, Sun Yat-Sen University, Guangzhou, China
[2]School of Computer Science and Engineering, Sun Yat-Sen University, Guangzhou, China
[3]Guangdong Provincial Key Laboratory of Information Security, Sun Yat-sen University, Guangzhou, China
raoy3@mail2.sysu.edu.cn, issjqni@mail.sysu.edu.cn

Abstract

With wide applications of image editing tools, forged images (splicing, copy-move, removal and etc.) have been becoming great public concerns. Although existing image forgery localization methods could achieve fairly good results on several public datasets, most of them perform poorly when the forged images are JPEG compressed as they are usually done in social networks. To tackle this issue, in this paper, a self-supervised domain adaptation network, which is composed of a backbone network with Siamese architecture and a compression approximation network (ComNet), is proposed for JPEG-resistant image forgery localization. To improve the performance against JPEG compression, ComNet is customized to approximate the JPEG compression operation through self-supervised learning, generating JPEG-agent images with general JPEG compression characteristics. The backbone network is then trained with domain adaptation strategy to localize the tampering boundary and region, and alleviate the domain shift between uncompressed and JPEG-agent images. Extensive experimental results on several public datasets show that the proposed method outperforms or rivals to other state-of-the-art methods in image forgery localization, especially for JPEG compression with unknown QFs.

1. Introduction

For image forgery forensics, a fundamental task consists in precisely localizing the forged regions, which is more challenging with the presence of post-processing operations, such as filtering, resampling and compression. Among these content-preserving manipulations, JPEG compression are most widely used in social networks to reduce transmission bandwidth or storage space. The subtle tampering artifacts, however, would be eliminated after strong JPEG compression, degrading the performance of forgery forensic methods. On the other hand, the variety of image tampering manipulations, e.g., splicing, copy-move, removal and etc., also seriously influences the generalization capability of forensic methods to unseen forgeries.

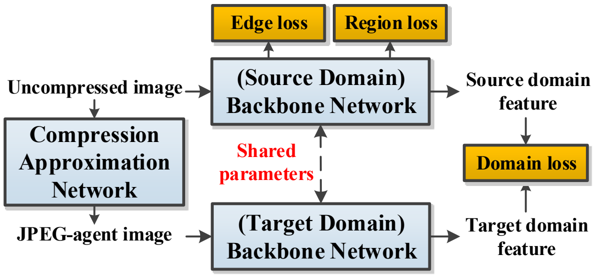


Figure 1. The architecture of the self-supervised domain adaptation network, which is composed of a backbone network with Siamese architecture and a compression approximation network.

Typically, conventional hand-crafted feature based methods rely on the hypothesis of a specific manipulation artifact, and aim to expose the forgeries by exploring local inconsistencies of color filter array (CFA) [14, 30], photo-response non-uniformity noise (PRNU) [23, 8], JPEG blocking artifacts [35, 19], texture units [25, 18], illumination [12, 6] and steganography-based local descriptors [32, 10, 22]. Alternatively, most of the state-of-the-art image forgery forensic methods [29, 27, 4, 34, 11, 24] resort to deep learning technique due to its powerful feature representation capability. Taking advantages of massive training samples with multiple forgeries, these deep learning based methods are able to perform much better in image forgery localization than the conventional ones. On the other hand, to achieve better robustness performance against JPEG compression, deep learning based methods commonly train the network models with targeted data augmentation strategy with tampered JPEG images of various quality factors (QFs). This, however, manifests deficiency in the following aspects: 1) undermining the boundary transition to some extents, which correspond to the high frequency components discarded by JPEG compression, therefore, increasing the difficulty to capture intrinsic feature representations of forgery manipulations; and 2) demanding diverse JPEG samples to relieve the mismatch of JPEG compression adopted between training and testing stages. These two issues degrade the performance of deep learning based method when conducting data augmentation in terms of JPEG compression, especially in the case of small training set.

In this paper, a self-supervised domain adaptation network, which is composed of a backbone network and a compression approximation network (ComNet), is proposed for JPEG-resistant image forgery localization as illustrated in Figure 1. The key idea consists in customizing the ComNet to approximate the JPEG compression operation through a self-supervised learning task, generating JPEG-agent images with more generalizable JPEG compression characteristics. Incorporated with ComNet, the backbone network with Siamese architecture is trained through domain adaptation strategy to improve performance against JPEG compression. The main contributions of this paper are summarized as follows:

- A conditional random field (CRF) based attention module is proposed to highlight the transition boundary of forged region. Unlike the simplified CRF model in [9] which is implemented with single spatial kernel and recursive convolution, we construct a standard CRF to better characterize local pattern correlation and implement only one iteration of mean field approximation [21] for CRF inference.

- An encoder-decoder based ComNet is proposed to approximate the JPEG compression operation through a self-supervised learning task, generating JPEG-agent images with general JPEG compression characteristics.

- In order to improve the performance against JPEG compression, domain adaptation strategy is applied to alleviate the domain shift between source (uncompressed images) and target (JPEG-agent images).

2. Related Works

2.1. Hand-crafted Feature Based Approaches

Conventional hand-crafted feature based approaches usually reveal the statistical dependency among pixels by modelling natural images, and capture the statistical deviation due to image tampering operations based on this statistical model. For example, in [38], a 2-D noncausal Markov model was proposed to characterize the underlying relationship of adjacent pixels for image forgery detection. In [32, 10], the spatial rich model (SRM) [15], which is widely used in image steganalysis, was generalized to extract residual-based feature for image forgery detection and localization through the support vector machine (SVM) classifier and multidimensional Gaussian model. Later, Li *et al.* [22] improved [32] by taking advantage of the possibility maps obtained with the statistical feature-based and copy-move detectors, where the spatial color rich model (SCRM) [16] was incorporated for splicing and erasing detection. On the other hand, tampering operations may inevitably induce the variations of visual elements in images, which can be effectively captured by local image descriptors for forgery detection. In this context, [6] combined the statistical characteristics extracted by various local descriptors which explore texture, illumination, shape and color features to detect the distortions caused by image splicing.

2.2. Deep Learning Based Approaches

Unlike the arduous process of feature engineering to construct the hand-crafted features in conventional approaches, deep learning based approaches can directly learn and optimize the feature representations for forgery forensics. In [26], a new initialization strategy was proposed to force the convolutional layer to learn residual features for forgery detection. As an extension of [26], [27] proposed an improved initialization strategy and adopted a Siamese network for splicing detection and localization. Siamese network was also utilized in Noiseprint [11] to capture camera model artifacts based on noise residual for forgery localization. Similarly, a CNN-based forensic similarity network [24] is proposed to determine whether a pair of image patches contains the same or different forensic traces, i.e., the source camera model and processing history. In general, all the above methods conduct forgery localization on a block-by-block basis, which is hard to generate fine-grained results. To tackle this issue, the network models perform well in semantic segmentation are generalized to conduct pixel-wise classification for forgery localization. In [29], a multi-task fully convolutional network (MFCN) was trained with ground truths of forged regions and boundaries for forgery localization. More recently, in [4], based on resampling and spatial features, a hybrid LSTM and encoder-decoder network was proposed for pixel-wise forgery localization, where the LSTM is utilized to capture the inconsistency in transitions between fake and authentic patches. ManTra-Net [34] formulated the forgery localization problem as local anomaly detection, and captured general image manipulation traces through a self-supervised learning task for classifying multiple manipulation types. In terms of data augmentation, in [39], a manipulated image generation process based on generative adversarial network (GAN) was proposed to produce tampered images through blending tam-

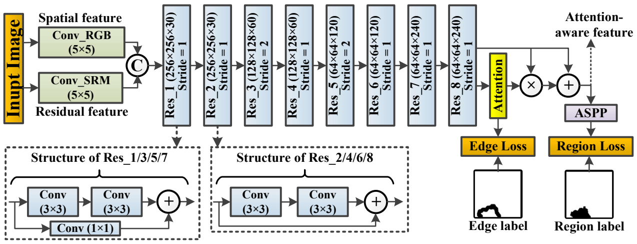


Figure 2. Architecture of the backbone network with the improved CRF-based attention module. The structure of each residual unit (Res) is shown in the dotted box, where each convolutional layer (Conv) is followed by a batch normalization layer and a ReLU activation function. The sizes of convolution kernels are denoted in Conv_RGB, Conv_SRM and each Conv. ⓒ, ⊕ and ⊗ represent concatenation, element-wise summation and element-wise multiplication, respectively. The size of output feature maps in each Res is specified as: height×width×channel. Res_3 and Res_5 adopt kernels of stride 2 to perform downsampling.

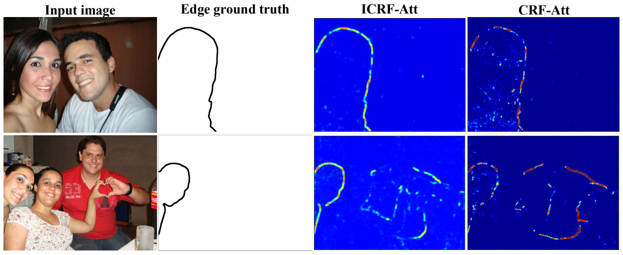


Figure 3. Samples of the generated attention maps. From left to right, each column shows the forged images, ground truths of forged boundary, attention maps generated with ICRF-Att and CRF-Att, respectively.

pered regions in existing datasets.

3. Self-supervised Domain Adaptation Network

In this Section, we elaborate three key designs of the proposed self-supervised domain adaptation network including: 1) backbone network; 2) compression approximation network (ComNet); and 3) domain adaptation strategy. As illustrated in Figure 1, the backbone network follows the Siamese network architecture, which consists of two parallel sub-networks with shared parameters. The uncompressed images and JPEG-agent images generated with ComNet constitute the source and target domains, respectively.

3.1. Backbone Network

The architecture of the sub-network is illustrated in Figure 2. In specific, to capture more tampering artifacts in spatial and residual domain, two parallel convolutional layers are adopted to extract dual-domain features, where the initialization strategy [26] is applied to the kernels in Conv_SRM. The next eight residual modules [17] are used to extract the hierarchical features at different scales, followed by an attention module to highlight the boundary of forged regions. Inspired by [9] which casts the CRF as attention model (CRF-Att) for characterizing local pattern correlation, we propose an improved CRF-based attention module (ICRF-Att) which incorporates with the differentiable implementation of mean field approximation (MFA) [21] to more fully exploits the local pattern correlation with three spatial kernels and only one iteration of MFA for reducing the computational cost.

We cast the attention map as a two-class problem. Denote x_i the random variable associated to pixel i in the attention map, and the domain of x_i is a set of labels $\mathcal{L} = \{0, 1\}$, where x_i = 1 and x_i = 0 represent the forged and authentic pixels, respectively. Let $\boldsymbol{x} = \{x_1, x_2, \cdots, x_N\}$ be a set of label assignments, where N is the number of pixels. To obtain the optimal $\boldsymbol{x}$, we minimize the energy function $E(\boldsymbol{x})$ in the fully connected CRF model [21] as follows:

$$\underset{x_i}{\mathrm{Min}}\, E(\boldsymbol{x}) = \underset{x_i}{\mathrm{Min}}(\sum_i \psi_u(x_i) + \sum_{i<j} \psi_p(x_i, x_j)), \quad (1)$$

where ψ_u is the unary potential that measures the cost of per-pixel label assignment. ψ_u is typically defined as $\psi_u(x_i) = -\ln(p(x_i))$, where $p(x_i)$ represents the probability of taking label x_i at pixel i. ψ_p is the pairwise potential that measures the penalty of assigning labels x_i, x_j to pixels i, j simultaneously.

Due to the optimization of Eq. 1 needs to estimate the exact Gibbs distribution of $\boldsymbol{x}$, which is intractable in practice, MFA is adopted to model a simpler distribution $Q(\boldsymbol{x})$, where $Q(\boldsymbol{x})$ can be expressed as a product of independent marginal distributions, i.e., $Q(\boldsymbol{x}) = \prod_i Q_i(x_i)$. Each $Q_i(x_i)$ is the variable we need to estimate, indicating the probability for assigning label x_i to pixel i. $Q_i(x_i = l)$ ($l \in \mathcal{L}$) is then computed iteratively with the update equation in MFA as follows:

$$\begin{aligned} Q_i(x_i = l) &= \frac{1}{Z_i} \exp(-\psi_u(x_i) \\ &- \sum_{l' \in \mathcal{L}} \mu(l, l') \sum_{m=1}^{K} w^{(m)} \sum_{j \neq i} k^{(m)}(\boldsymbol{f}_i, \boldsymbol{f}_j) Q_j(l')), \end{aligned} \quad (2)$$

where $k(\boldsymbol{f}_i, \boldsymbol{f_j})$ represents the Gaussian kernel applied on the feature vectors $\boldsymbol{f}_i$ and $\boldsymbol{f}_j$ corresponding to pixels i and j. $\mu(.,.)$ is the label compatibility function that captures the compatibility between different pairs of labels. Denote as F the input feature map of size 64×64×240, and $M = Q(x = l)$ the attention map of size 64×64×2, where each channel represents the probability of forgery and authenticity, respectively. M is generated by implementing Eq. 2 as follows:

$$\psi_u = \ln(\sigma(F * \boldsymbol{w}_u)), \quad (3)$$

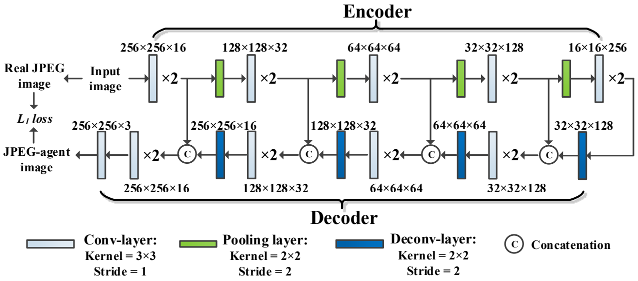


Figure 4. Architecture of the compression approximation network. The sizes of output feature maps for each convolution layer (conv-layer) and deconvolution layer (deconv-layer) is specified as: height×width×channel.

$$M = \mathrm{softmax}(\psi_u - \mathrm{lin}(\boldsymbol{w}_3 * \mathrm{lin}(\boldsymbol{w}_2 * \mathrm{lin}(\boldsymbol{w}_1 * \psi_u)))), \quad (4)$$

where $\boldsymbol{w}_u$, $\boldsymbol{w}_1$, $\boldsymbol{w}_2$ and $\boldsymbol{w}_3$ are convolution kernels of sizes 240×3×3×2, 2×5×5×8, 8×1×1×2 and 2×1×1×2, respectively. $*$ represents convolution operation. In this way, $\boldsymbol{w}_u$ summarizes the information in F across channels to generate ψ_u of size 64×64×2. $\boldsymbol{w}_1$ implements the message passing process in MFA, extracting local patterns in a 5×5 region through 8 filters. $\boldsymbol{w}_2$ generates the weighted sum of the filter outputs, and $\boldsymbol{w}_3$ learns the label compatibility function. In additon, $\sigma(x) = 1/(1+\exp(-x))$, $\mathrm{lin}(x) = x + b$ and $\mathrm{softmax}(\cdot)$ represent sigmoid, linear activation with bias b and channel-wise softmax functions, respectively.

Denoted as M^0 the first channel of M, due to training with edge ground truth for the attention module, element intensities surrounding forged boundaries are expected to be larger in M^0, representing higher probability of being forged. M^0 is then applied to refine F, leading to the attention-aware feature H as follows:

$$H_c = F_c \otimes (1 + M^0), \quad (5)$$

where c is the index of feature channel and $\otimes$ represents element-wise multiplication. Finally, taking as input H, atrous spatial pyramid pooling (ASPP) [7] is used to exploit multi-scale features to generate the fine-grained localization result. To illustrate the superiority of ICRF-Att, we visualize M^0 as shown in Figure 3. Due to better exploiting local correlation, ICRF-Att more accurately delineates the forged boundaries with rare false alarms comparing with CRF-Att. In terms of efficiency, ICRF-Att could decrease by two convolution operations per forward pass compared with CRF-Att.

3.2. Compression Approximation Network

To improve the performance against JPEG compression, we intend to construct a kind of JPEG-agent images with general JPEG compression characteristics rather than those related to a specific quality factor (QF). To this end, we propose a compression approximation network, i.e., ComNet, to approximate the JPEG compression operation, irrespective of QF. This can be effectively learned from a self-supervised learning task. In specific, as illustrated in Figure 4, based on the encoder-decoder network with skip connections [28] between mirrored layers, ComNet generates the JPEG-agent image I^a from input uncompressed image I. Note that I is a forged image without JPEG compression, instead of an authentic one. The target image, i.e., real JPEG image I^c, is compressed from I with a random QF using a standard JPEG compression algorithm. According to our experiment, $\mathbb{L}_2$ loss is less robust to outliers than $\mathbb{L}_1$ loss. Therefore, ComNet is then trained through minimizing $\mathbb{L}_1$ loss between I^a and I^c as follows:

$$L_{com} = \frac{1}{N}\sum_{i=1}^{N} |I_i^a - I_i^c|, \quad (6)$$

where i is the index of pixel and N is the number of pixels. By incorporating with JPEG features involving multiple QFs, the generated I^a with ComNet, i.e., the JPEG-agent image, is expected to exhibit more general JPEG compression characteristics than the real JPEG image itself, which is also able to generalize to JPEG images with unseen QFs.

3.3. Domain Adaptation Strategy

To perform JPEG-resistant image forgery localization, instead of directly mixing the training set with JPEG or JPEG-agent samples, domain adaptation [5] strategy is applied to the source and target domains, corresponding to the uncompressed and JPEG-agent images, respectively. Domain adaptation facilitates to capture intrinsic tampering artifacts in source domain, and simultaneously achieve better generalization ability to JPEG compression in target domain. The backbone network is trained with a combination of three losses: 1) region loss (L_r) for performing pixel-wise classification; 2) edge loss (L_e) for generating the attention map of forged boundary; and 3) domain loss (L_d) for reducing domain shift and transferring effective knowledge from source to target domain. Note that L_r and L_e are only computed within source domain. Denote as H and H_a the attention-aware features in Eq. 5 extracted from uncompressed and JPEG-agent images, respectively, the involved losses are computed as follows:

$$L_d = \frac{1}{N}\sum_{i=1}^{N} ||H_i - H_i^a||_2^2, \quad (7)$$

$$L_e = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=0}^{1}(\hat{w}^c \hat{y}_i^c \log(M_i^c)), \quad (8)$$

$$L_r = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=0}^{1}(w^c y_i^c \log(p_i^c)), \quad (9)$$

Dataset	Type	Num.	Size	Format
IFC [1]	training	441	800×600	PNG
DSO-1 [6]	testing	100	2048×1536	PNG
Korus [20]	testing	220	1920×1080	TIFF
Coverage [33]	testing	100	489×380	TIFF
CASIA1 [13]	testing	946	384×256	JPEG (QF=75,95,100)
Nim.16 [3]	testing	564	3561×2516	JPEG (QF=75,95,99,100)
Wild Web [36]	testing	9657	390×532	JPEG (QF=24∼100)
CASIA2 [13]	testing	5124	473×322	JPEG (40%, QF=95,97,100) +TIFF (60%)
MFC18dev1 [2]	testing	3886	3077×2034	JPEG (75%, QF=53∼100) +PNG (25%)

Table 1. The datasets involved in our experiments. The second and third columns list the number and average resolution of images, respectively.

where i is the index of spatial position, N is the number of elements. p^c and M^c represent the probabilities of class c for the softmax output of ASPP and the attention map, respectively. $\hat{y}$ and y represent the forged edge and region ground truths, respectively. $\hat{w}^c$ and w^c are the weights of class c corresponding to $\hat{y}$ and y for avoiding model bias, respectively. Finally, we minimize the joint loss function as follows:

$$L = L_e + L_r + \alpha \cdot L_d, \tag{10}$$

where the hyper-parameter α controls the balance between different tasks.

4. Experimental Results and Analysis

4.1. Dataset

To better show the generalization capability, we conduct cross-dataset evaluation to avoid dataset-related polarization on nine datasets, which are summarized in Table 1. In specific, we train the proposed approach on IEEE Forensics Challenge [1] (IFC) dataset containing only uncompressed images tampered with splicing and copy-move manipulations, while test the performance on other unseen datasets. For testing datasets, DSO-1 [6] dataset focuses on detecting splicing images containing people. Coverage [33] dataset is used to detect copy-move manipulation among multiple similar-but-genuine objects. Korus [20] dataset includes realistic forged images acquired by only four cameras. Wild Web dataset [36] comprises real-world splicing images collected from various social media, which is more challenging due to multiple post-processing operations, such as compression and rescaling. Other datasets cover various forgery scenarios involving splicing, copy-move and removal, sometimes even a series of forgery manipulations and random post-processing operations are included in one image. Five of these datasets include JPEG compressed images, where the involved compression operations are not made public.

4.2. Training Settings

We implement the proposed network in TensorFlow. The backbone network and ComNet are trained with the forged images (PNG format) on IFC dataset. We draw uncompressed patches I of size 256×256×3 which include 20% to 80% forged pixels, leading to 35,000 samples. For the ComNet, target JPEG patches (I^c) is compressed from I through $Matlab$ API function, using random QFs (QF=55, 65, 75, 85, 95 and 100). Firstly, we train the ComNet with Adam optimizer, setting mini-batch size to 64. Taking as input I, the pre-trained ComNet model generates the JPEG-agent counterparts I^a. The backbone network is then trained with I and I^a based on SGD optimization with a momentum value of 0.9. For both network models, the initial learning rates are set to 0.01 with 10% of decrement every 10 epochs, and the weight decays are fixed to $6{\times}10^{-3}$ during training stages. Hyper-parameter α in (10) is set to 1. The optimal α is derived from training data, where $\alpha = 1$ achieves the best performance among other settings.

4.3. Ablation Study

Before proceeding to comparative experiments, we conduct ablation study to investigate the effectiveness of the involved components in our proposed network, i.e., the proposed attention module (ICRF-Att), JPEG-agent image (I^a) and domain adaptation (DA) strategy as shown in Table 2. The localization performance is compared on Nim.16 [3], DSO-1 and Wild Web datasets in terms of F1-score (F1) and Matthews Correlation Coefficient (MCC) which are generally used in the previous studies.

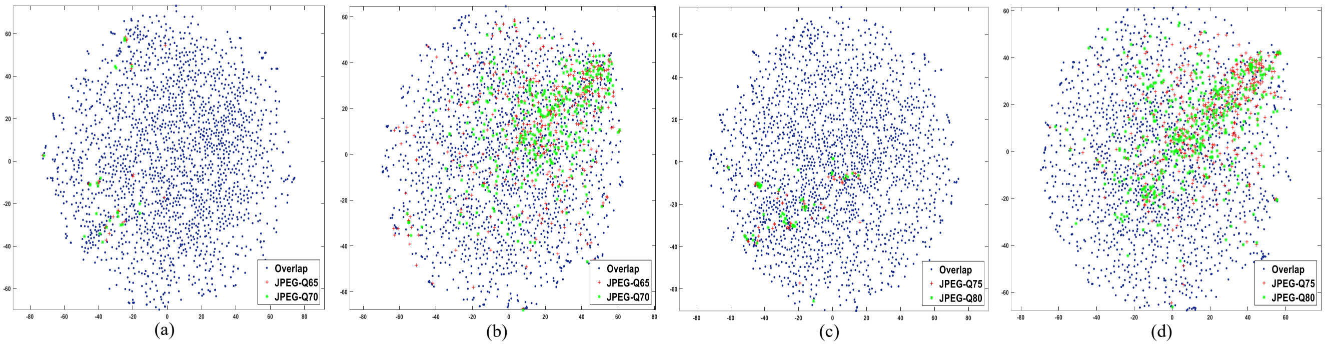
ICRF-Att. It is ready to see that the network incorporated with ICRF-Att consistently outperforms those without attention model and with CRF-Att [9] by a clear margin.

JPEG-agent image (I^a). The network trained with the uncompressed images (I) and I^a outperforms the one trained with I and the real JPEG images (I^c), however, it turns out to be unsatisfactory compared with the one training on I only. This indicates that improper data augmentation prevents the network from extracting more intrinsic feature representation for forgery localization.

Domain adaptation (DA) strategy. By adopting DA between the source domain I and target domain I^c (DA-I^c for short) or souce domain I and target domain I^a (DA-I^a for short), a consistent performance gain is achieved in comparison with the network trained on the same dataset without DA. This is due to DA facilitates transferring prior knowledge of capturing tampering artifacts from I to I^c or I^a. Taking advantage of DA, DA-I^a outperforms DA-I^c by 3.2%, 2.7% and 1.4% in terms of MCC on Nim.16, DSO-1 and Wild Web datasets, respectively, indicating more general JPEG compression characteristics embedded in the JPEG-agent image I^a. In addition, we compare the $\mathbb{L}_2$ domain loss with the ones computed via $\mathbb{L}_1$ (L_{d_L1}) and maxi-

Attention model	CRF-Att [9]		✓							
	ICRF-Att			✓	✓	✓	✓	✓	✓	✓
	W/o attention	✓								
Training data	Uncompressed image (I)	✓	✓	✓	✓	✓	✓	✓	✓	✓
	JPEG-agent image (I^a)					✓		✓	✓	✓
	Real JPEG image (I^c)				✓		✓			
Domain adaptation (DA)	$\mathbb{L}_2$ loss						✓	✓		
	$\mathbb{L}_1$ loss								✓	
	MMD loss									✓
	W/o DA	✓	✓	✓	✓	✓				
Nim. 16 dataset	MCC	0.254	0.285	0.351	0.256	0.333	0.359	**0.391**	0.362	0.372
	F1	0.295	0.315	0.378	0.287	0.362	0.387	**0.416**	0.387	0.396
DSO-1 dataset	MCC	0.712	0.742	0.805	0.590	0.774	0.795	0.822	0.792	**0.824**
	F1	0.743	0.767	0.826	0.642	0.801	0.794	**0.842**	0.816	0.841
Wild Web dataset	MCC	0.184	0.201	0.210	0.194	0.204	0.204	**0.218**	0.212	0.213
	F1	0.222	0.239	0.249	0.246	0.249	0.252	**0.264**	0.254	0.257

Table 2. The localization performance comparisons for ablation study.



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

Figure 5. Visualization of features extracted from JPEG compressed patches with different quality factors (QFs) through the network models DA-I^a with JPEG-agent images ((a) and (c)) and DA-I^c with real JPEG images ((b) and (d)), respectively. Overlap represents the overlapping feature points where the Euclidean distance between them is smaller than 0.4. The patches compressed with QF=65,70 and QF=75,80 share the same original patches, respectively.

mum mean discrepancy (MMD) [31] (L_{d_M}) loss functions as follows:

$$L_{d\_L1} = \frac{1}{N}\sum_{i=1}^{N}|H_i - H_i^a|, \tag{11}$$

$$L_{d\_M} = \frac{1}{N}\sum_{i=1}^{N}||\frac{1}{N_s}\sum_{j=1}^{N_s}H_{i,j} - \frac{1}{N_t}\sum_{j=1}^{N_t}H_{i,j}^a||_2^2, \tag{12}$$

where N_s and N_t represent the numbers of I and I^a within a mini-batch, respectively, and other variables take the same notations in Eq. 7. It is observed that $\mathbb{L}_2$ loss achieves the best performance, which is more effective to alleviate the domain shift for the proposed network.

To further demonstrate the generalization ability of the proposed network, we compare the feature distributions of patches (extracted from IFC dataset) compressed with different QFs for DA-I^a and DA-I^c, including seen (QF=65, 75) and unseen (QF=70, 80) QFs in the training set. In specific, among the I^c used to train the ComNet, we select two groups of JPEG compressed forgery patches, where each group consists of 2,000 patches compressed with QF65 and QF75, respectively. We compress the corresponding original patches of each group with QF70 and QF80, respectively. For each QF, the extracted attention-aware features H in Eq. 5 is projected to a 2-D space for visualization as shown in Figure 5. We then show the isolated feature points for each QF, and highlight the overlapping points if two feature points (i.e., two patches compressed with different QFs from the same uncompressed patch) are overlapped, which means the Euclidean distance is smaller than a threshold τ. Considering that τ in [0.1, 0.7] gives similar results, we take the median, i.e., $\tau = 0.4$ as the threshold. It is observed that much more overlapping points are achieved with the network model DA-I^a than DA-I^c, indicating less deviation of the extracted features between different QFs. Therefore, the effect of QFs is largely suppressed when adapting to the target domain of I^a, leading to better generalization ability to the compression operations of unseen QFs.

4.4. Comparisons With Other State-of-the-art Methods

To comprehensively evaluate the superiority of our method, as illustrated in Table 3, 4 and 5, we compare it

Dataset	DSO-1	Coverage	Korus	CASIA1	CASIA2	Nim.16	MFC18dev1	AVERAGE
CAGI [19]	0.490 (5)	0.266 (8)	0.202 (9)	0.248 (7)	0.222 (5)	0.279 (4)	0.250 (4)	0.280 (6)
CFA1 [14]	0.155 (11)	0.142 (11)	0.365 (2)	0.119 (11)	0.106 (11)	0.157 (10)	0.109 (11)	0.165 (11)
NOI5 [37]	0.409 (7)	0.291 (7)	0.234 (7)	0.258 (4)	0.204 (6)	0.227 (7)	0.220 (5)	0.263 (8)
ITPM [22]	0.721 (4)	**0.534 (1)**	0.312 (5)	0.251 (6)	0.197 (7)	0.259 (5)	0.202 (6)	0.354 (4)
MFCN [29]	0.213 (10)	0.212 (10)	0.162 (10)	0.215 (8)	0.180 (9)	0.143 (11)	0.130 (10)	0.179 (10)
HLED [4]	0.227 (9)	0.251 (9)	0.221 (8)	0.182 (10)	0.154 (10)	0.219 (8)	0.185 (8)	0.206 (9)
DLLD [27]	0.486 (6)	0.336 (4)	0.156 (11)	0.256 (5)	0.184 (8)	0.257 (6)	0.199 (7)	0.268 (7)
ManTra-Net [34]	0.374 (8)	0.440 (3)	0.242 (6)	**0.297 (1)**	0.268 (2)	0.165 (9)	0.176 (9)	0.280 (5)
FS [24]	0.763 (2)	0.299 (6)	**0.366 (1)**	0.267 (3)	0.254 (3)	0.361 (3)	0.314 (3)	0.375 (2)
Noiseprint [11]	0.758 (3)	0.306 (5)	0.345 (3)	0.205 (9)	0.237 (4)	0.387 (2)	**0.341 (1)**	0.368 (3)
Proposed	**0.822 (1)**	0.493 (2)	0.318 (4)	0.282 (2)	**0.311 (1)**	**0.391 (1)**	0.328 (2)	**0.421 (1)**

Table 3. The localization performance comparisons of the propose method with other state-of-the-art methods in terms of MCC.

Dataset	DSO-1	Coverage	Korus	CASIA1	CASIA2	Nim.16	MFC18dev1	AVERAGE
CAGI [19]	0.539 (5)	0.307 (8)	0.184 (9)	0.248 (5)	0.226 (5)	0.300 (4)	0.281 (4)	0.298 (6)
CFA1 [14]	0.290 (11)	0.223 (11)	**0.363 (1)**	0.157 (11)	0.154 (11)	0.158 (11)	0.191 (11)	0.219 (11)
NOI5 [37]	0.463 (8)	0.330 (5)	0.222 (7)	0.264 (3)	0.204 (7)	0.226 (8)	0.254 (6)	0.280 (7)
ITPM [22]	0.752 (4)	0.553 (1)	0.301 (5)	0.208 (7)	0.195 (8)	0.295 (5)	0.263 (5)	0.367 (4)
MFCN [29]	0.331 (10)	0.263 (10)	0.168 (10)	0.224 (6)	0.205 (6)	0.193 (10)	0.211 (10)	0.228 (10)
HLED [4]	0.339 (9)	0.303 (9)	0.218 (8)	0.200 (9)	0.187 (10)	0.245 (7)	0.240 (8)	0.247 (9)
DLLD [27]	0.535 (6)	0.312 (7)	0.137 (11)	0.252 (4)	0.188 (9)	0.265 (6)	0.222 (9)	0.273 (8)
ManTra-Net [34]	0.469 (7)	0.484 (3)	0.257 (6)	**0.316 (1)**	0.305 (2)	0.213 (9)	0.253 (7)	0.328 (5)
FS [24]	0.785 (2)	0.317 (6)	0.333 (3)	0.193 (10)	0.237 (3)	0.361 (3)	0.345 (3)	0.367 (3)
Noiseprint [11]	0.780 (3)	0.334 (4)	0.350 (2)	0.204 (8)	0.237 (3)	0.395 (2)	**0.373 (1)**	0.382 (2)
Proposed	**0.842 (1)**	0.516 (2)	0.327 (4)	0.275 (2)	**0.331 (1)**	**0.416 (1)**	0.359 (2)	**0.438 (1)**

Table 4. The localization performance comparisons of the propose method with other state-of-the-art methods in terms of F1.

with various state-of-the-art methods in terms of F1, MCC and average precision (AP, i.e, the area under the precision-recall curve), and also compute the average performance over all datasets in the last column and the corresponding rank of each method on each dataset in parenthesis. These competing methods which can be divided into 1) the traditional hand-crafted feature based and 2) deep learning based methods, are able to localize general image forgery manipulations. The former includes CAGI [19], CFA1 [14], NOI5 [37] and ITPM [22], exploiting features based on JPEG, CFA, noise level artifacts and the combination of SCRM [16] and the patch matching detector, respectively. The latter includes MFCN [29], HLED [4] and ManTra-Net [34] which generalizes networks used in semantic segmentation to perform pixel-wise forgery localization, and DLLD [27], Forensic Similarity (FS) [24] and Noiseprint [11] which capture statistical inconsistency through residual based and camera model based features. For FS, we report the results of 10 times of repeated tests, where we randomly select a reference patch each time. In the interest of a fair comparison, except for the unsupervised methods (CAGI, CFA1 and NOI5), FS, Noiseprint and ManTra-Net are tested with their released pre-trained models, and all the remaining data-driven methods are fine-tuned on the same dataset as our method.

It is observed that our proposed method always ranks the top two on each dataset, except for Korus [20] dataset, where comparable performance is also achieved. Korus dataset includes raw images acquired by only four specific camera models, which are not included in our training set. While those deep learning based methods, whose performances are superior to our proposed method on Korus dataset, e.g., Noiseprint and FS, all take advantage of the camera model based features. It is noted that our method outperforms Noiseprint and FS by a clear margin on Coverage (TIFF images), and CASIA1 (JPEG images) and CASIA2 (JPEG+TIFF images) datasets. Comparing with the others, these datasets involve relatively small-sized images with average sizes of 489×380, 384×256 and 473×322 pixels, respectively. Such low resolution images tend to be difficult to be detected by Noiseprint due to the extracted features within an image are too scarce to allow correct clustering by the expectation-maximization (EM) algorithm in Noiseprint. For small-sized images, FS could not generate fine-grained results due to the sliding-window size is fixed to 128×128 or 256×256 pixels. Our method performs better on CASIA2 than CASIA1 datasets, because CASIA2 includes 60% uncompressed images which are easier to be detected. For JPEG compressed images, Noiseprint deploys 46 networks, each of which is trained with JPEG images of a specific QF (QF55-QF100). In contrast, the proposed method is trained through a single Siamese network with uncompressed and JPEG-agent images, which greatly reduces the computational cost. In general, our method achieves the most superior average performance, outperforming the second best method Noiseprint by 5.3%, 5.6%

Dataset	DSO-1	Coverage	Korus	CASIA1	CASIA2	Nim.16	MFC18dev1	AVERAGE
CAGI [19]	0.517 (5)	0.216 (7)	0.131 (9)	0.170 (6)	0.185 (6)	0.272 (4)	0.263 (4)	0.251 (6)
CFA1 [14]	0.236 (11)	0.150 (1)	**0.318 (1)**	0.102 (11)	0.111 (11)	0.116 (11)	0.154 (11)	0.170 (11)
NOI5 [37]	0.384 (8)	0.241 (5)	0.164 (8)	0.205 (4)	0.159 (8)	0.227 (6)	0.238 (5)	0.231 (7)
ITPM [22]	0.764 (3)	**0.510 (1)**	0.238 (5)	**0.314 (1)**	0.292 (2)	0.228 (5)	0.221 (6)	0.330 (2)
MFCN [29]	0.280 (9)	0.171 (10)	0.127 (10)	0.164 (7)	0.166 (7)	0.140 (10)	0.166 (10)	0.173 (10)
HLED [4]	0.256 (10)	0.229 (6)	0.167 (7)	0.132 (8)	0.141 (9)	0.194 (8)	0.192 (8)	0.187 (9)
DLLD [27]	0.456 (6)	0.200 (9)	0.082 (11)	0.177 (5)	0.136 (10)	0.213 (7)	0.170 (9)	0.205 (8)
ManTra-Net [34]	0.432 (7)	0.463 (3)	0.210 (6)	0.271 (2)	0.266 (3)	0.161 (9)	0.211 (7)	0.288 (5)
FS [24]	0.799 (2)	0.206 (8)	0.278 (3)	0.128 (9)	0.187 (5)	0.319 (3)	0.304 (3)	0.317 (4)
Noiseprint [11]	0.728 (4)	0.247 (4)	0.288 (2)	0.128 (9)	0.195 (4)	0.332 (2)	**0.335 (1)**	0.322 (3)
Proposed	**0.851 (1)**	0.489 (2)	0.268 (4)	0.242 (3)	**0.302 (1)**	**0.381 (1)**	0.323 (2)	**0.408 (1)**

Table 5. The localization performance comparisons of the propose method with other state-of-the-art methods in terms of AP.

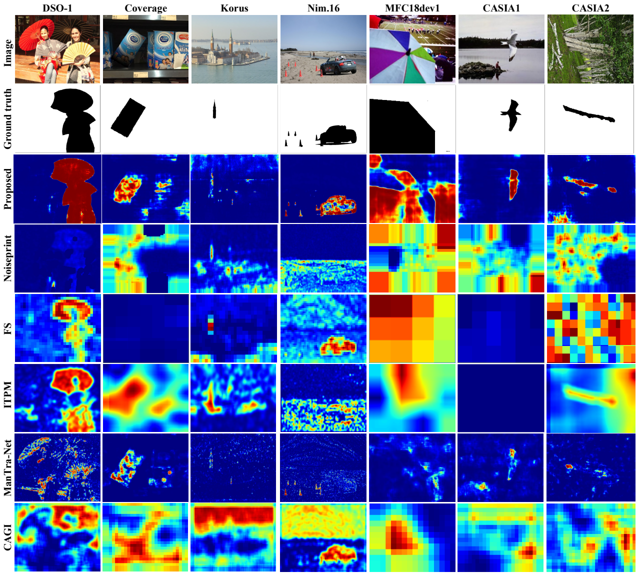


Figure 6. Forgery localization results on testing datasets. From top to bottom, each row shows the forged image, ground truth and heat maps from the competing methods: the proposed one, Noiseprint, FS, ITPM, ManTra-Net and CAGI. The black and white pixels in a ground truth correspond to the forged and authentic pixels, respectively.

and 8.6% in terms of MCC, F1 and AP, respectively. Finally, we compare the qualitative localization results of tampering heat maps among the best six methods in terms of average performance on all testing datasets in Figure 6, illustrating the cases where we can more precisely localize the forged regions.

4.5. Robustness on Extreme Compression Scenarios

To further verify the robustness performance of our method against JPEG compression, we compare the performance on extreme compression scenarios. Considering that images are usually compressed with QF≥70 in most social networks, e.g., Facebook (QF=71) and Wechat (QF=70), QF<70 can be regarded as the extreme compression scenarios in real-world cases. We compare our method with the top-3 deep learning based methods (ManTra-Net, FS and Noiseprint) in terms of F1 on Korus and Nim.16 datasets, where the images are compressed with QF=50, 60, 70. As illustrated in Table 6, our method still achieves the best robustness performance against JPEG compression with the tested QFs.

Dataset	Korus			Nim. 16		
Quality	QF=70	QF=60	QF=50	QF=70	QF=60	QF=50
ManTra-Net [34]	0.160	0.155	0.148	0.178	0.175	0.171
FS [24]	0.196	0.180	0.173	0.243	0.241	0.238
Noiseprint [11]	0.187	0.177	0.178	0.283	0.280	0.275
Proposed	**0.221**	**0.212**	**0.208**	**0.317**	**0.299**	**0.280**

Table 6. Robustness performance against JPEG compression in terms of F1 score.

5. Conclusion

In this paper, a self-supervised domain adaptation network, which incorporates a backbone network with a compression approximation network (ComNet), is proposed for JPEG-resistant image forgery localization. Instead of data augmentation with various real JPEG compressed images, we generate JPEG-agent images through ComNet which is trained with self-supervised learning to approximate the JPEG compression operation. The JPEG-agent images exhibit more generalizable characteristics of JPEG compression, and are applied to domain adaptation strategy for alleviating the domain shift between uncompressed and JPEG-agent images, leading to better robustness performance against JPEG compression. Extensive experiments are carried out on several public datasets, which demonstrates the superior generalization ability of the proposed method over other state-of-the-art methods.

Acknowledgment

This work was supported in part by the National Natural Science Foundation of China under Grants U1936212, U1726315, and 61772573.

References

[1] IEEE IFS-TC Image Forensics Challenge Dataset. http://ifc.recod.ic.unicamp.br/fc.website/index.py.

[2] The media forensics challenge 2018 (MFC2018) evaluation. https://www.nist.gov/itl/iad/mig/media-forensics-challenge-2018.

[3] Nimble challenge 2017 evaluation. https://www.nist.gov/itl/iad/mig/nimble-challenge-2017-evaluation.

[4] J. H. Bappy, C. Simons, L. Nataraj, B. S. Manjunath, and A. K. Roy-Chowdhury. Hybrid LSTM and encoderdecoder architecture for detection of image forgeries. *IEEE Transactions on Image Processing*, 28(7):3286–3300, 2019.

[5] G. Cai, Y. Wang, L. He, and M. Zhou. Unsupervised domain adaptation with adversarial residual transform networks. *IEEE Transactions on Neural Networks and Learning Systems*, 31(8):3073–3086, 2020.

[6] T. Carvalho, F. A. Faria, H. Pedrini, R. da S. Torres, and A. Rocha. Illuminant-based transformed spaces for image forensics. *IEEE Transactions on Information Forensics and Security*, 11(4):720–733, 2016.

[7] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(4):834–848, 2018.

[8] G. Chierchia, G. Poggi, C. Sansone, and L. Verdoliva. A bayesian-MRF approach for PRNU-based image forgery detection. *IEEE Transactions on Information Forensics and Security*, 9(4):554–567, 2014.

[9] X. Chu, W. Yang, W. Ouyang, C. Ma, A. L. Yuille, and X. Wang. Multi-context attention for human pose estimation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5669–5678, 2017.

[10] D. Cozzolino, D. Gragnaniello, and L. Verdoliva. Image forgery detection through residual-based local descriptors and block-matching. In *2014 IEEE International Conference on Image Processing (ICIP)*, pages 5297–5301, 2014.

[11] D. Cozzolino and L. Verdoliva. Noiseprint: A CNN-based camera model fingerprint. *IEEE Transactions on Information Forensics and Security*, 15:144–159, 2020.

[12] T. J. d. Carvalho, C. Riess, E. Angelopoulou, H. Pedrini, and A. d. R. Rocha. Exposing digital image forgeries by illumination color classification. *IEEE Transactions on Information Forensics and Security*, 8(7):1182–1194, 2013.

[13] J. Dong and W. Wang. CASIA tampered image detection evaluation (TIDE) database, v1.0 and v2.0. http://forensics.idealtest.org/.

[14] Pasquale Ferrara, Tiziano Bianchi, Alessia De Rosa, and Alessandro Piva. Image forgery localization via fine-grained analysis of CFA artifacts. *IEEE Transactions on Information Forensics and Security*, 7(5):1566–1577, 2012.

[15] Jessica Fridrich and Jan Kodovsky. Rich models for steganalysis of digital images. *IEEE Transactions on Information Forensics and Security*, 7(3):868–882, 2012.

[16] M. Goljan, J. Fridrich, and R. Cogranne. Rich model for steganalysis of color images. In *2014 IEEE International Workshop on Information Forensics and Security (WIFS)*, pages 185–190, 2014.

[17] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[18] M. Hussain, S. Q. Saleh, H. Aboalsamh, G. Muhammad, and G. Bebis. Comparison between WLD and LBP descriptors for non-intrusive image forgery detection. In *2014 IEEE International Symposium on Innovations in Intelligent Systems and Applications (INISTA) Proceedings*, pages 197–204, 2014.

[19] Chryssanthi Iakovidou, Markos Zampoglou, Symeon Papadopoulos, and Yiannis Kompatsiaris. Content-aware detection of JPEG grid inconsistencies for intuitive image forensics. *Journal of Visual Communication and Image Representation*, pages 155–170, 2018.

[20] P. Korus and J. Huang. Evaluation of random field models in multi-modal unsupervised tampering localization. In *2016 IEEE International Workshop on Information Forensics and Security (WIFS)*, pages 1–6, 2016.

[21] Philipp Krhenbhl and Vladlen Koltun. Efficient inference in fully connected CRFs with gaussian edge potentials. In *Advances in Neural Information Processing Systems*, pages 109–117, 2011.

[22] H. Li, W. Luo, X. Qiu, and J. Huang. Image forgery localization via integrating tampering possibility maps. *IEEE Transactions on Information Forensics and Security*, 12(5):1240–1252, 2017.

[23] Babak Mahdian and Stanislav Saic. Using noise inconsistencies for blind image forensics. *Image and Vision Computing*, 27(10):1497–1503, 2009.

[24] O. Mayer and M. C. Stamm. Forensic similarity for digital images. *IEEE Transactions on Information Forensics and Security*, 15:1331–1346, 2020.

[25] Ghulam Muhammad, Munner H. Al-Hammadi, Muhammad Hussain, and George Bebis. Image forgery detection using steerable pyramid transform and local binary pattern. *Machine Vision and Applications*, 25(4):1–11, 2014.

[26] Y. Rao and J. Ni. A deep learning approach to detection of splicing and copy-move forgeries in images. In *2016 IEEE International Workshop on Information Forensics and Security (WIFS)*, pages 1–6, 2016.

[27] Y. Rao, J. Ni, and H. Zhao. Deep learning local descriptor for image splicing detection and localization. *IEEE Access*, 8:25611–25625, 2020.

[28] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI) 2015*, pages 234–241, 2015.

[29] Ronald Salloum, Yuzhuo Ren, and C. C. Jay Kuo. Image splicing localization using a multi-task fully convolutional network (MFCN). *Journal of Visual Communication and Image Representation*, 51(feb.):201–209, 2017.

[30] Ashwin Swaminathan, Min Wu, and K. J. Ray Liu. Nonintrusive component forensics of visual sensors using output

images. *IEEE Transactions on Information Forensics and Security*, 2(1):91–106, 2007.

[31] Eric Tzeng, Judy Hoffman, Ning Zhang, Kate Saenko, and Trevor Darrell. Deep domain confusion: Maximizing for domain invariance. *Computer ence*, 2014.

[32] L. Verdoliva, D. Cozzolino, and G. Poggi. A feature-based approach for image tampering detection and localization. In *2014 IEEE International Workshop on Information Forensics and Security (WIFS)*, pages 149–154, 2014.

[33] B. Wen, Y. Zhu, R. Subramanian, T. Ng, X. Shen, and S. Winkler. COVERAGE – a novel database for copy-move forgery detection. In *2016 IEEE International Conference on Image Processing (ICIP)*, pages 161–165, 2016.

[34] Y. Wu, W. AbdAlmageed, and P. Natarajan. ManTra-Net: Manipulation tracing network for detection and localization of image forgeries with anomalous features. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9535–9544, 2019.

[35] S. Ye, Q. Sun, and E. Chang. Detecting digital image forgeries by measuring inconsistencies of blocking artifact. In *2007 IEEE International Conference on Multimedia and Expo*, pages 12–15, 2007.

[36] Markos Zampoglou, Symeon Papadopoulos, and Yiannis Kompatsiaris. Detecting image splicing in the wild (web). In *2015 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, pages 1–6, 2015.

[37] Hui Zeng, Yifeng Zhan, Xiangui Kang, and Xiaodan Lin. Image splicing localization using PCA-based noise level estimation. *Multimedia Tools and Applications*, 76(4):4783–4799, 2017.

[38] Xudong Zhao, Shilin Wang, Shenghong Li, and Jianhua Li. Passive image-splicing detection by a 2-D noncausal markov model. *IEEE Transactions on Circuits and Systems for Video Technology*, 25(2):185–199, 2015.

[39] Peng Zhou, Bor-Chun Chen, Xintong Han, Mahyar Najibi, Abhinav Shrivastava, Ser-Nam Lim, and Larry Davis. Generate, segment, and refine: Towards generic manipulation segmentation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34:13058–13065, 04 2020.