DOA-GAN: Dual-Order Attentive Generative Adversarial Network for Image Copy-move Forgery Detection and Localization

Ashraful Islam\(^1\), Chengjiang Long\(^2\)*, Arslan Basharat\(^2\), Anthony Hoogs\(^2\),
\(^1\)Rensselaer Polytechnic Institute, Troy, NY
\(^2\)Kitware Inc., Clifton Park, NY
islama6@rpi.edu, \{chengjiang.long, arslan.basharat, anthony.hoogs\}@kitware.com

Abstract

Images can be manipulated for nefarious purposes to hide content or to duplicate certain objects through copy-move operations. Discovering a well-crafted copy-move forgery in images can be very challenging for both a human or a machine, for instance, when an image patch with uniform background is copied to hide an object. In this paper, we propose a Generative Adversarial Network with a dual-order attention model to detect and localize copy-move forgery. In the generator, the 1st-order attention is designed to capture the copy-move aware location information, and the 2nd-order attention exploits more discriminative features for the patch co-occurrence. Both attention maps are extracted from the affinity matrix and are used to location-aware and co-occurrence features to form the fused representation for the final detection and localization branches of the network. The discriminator network is designed to further ensure more accurate localization results. To the best of our knowledge, we are the first to propose such a network architecture with the 1st-order attention mechanism from the affinity matrix. We have performed extensive experimental validation and the results strongly demonstrate efficacy of the proposed approach.

1. Introduction

The content of digital images can be easily manipulated or forged now-a-days, as there are many image editing tools like GIMP or Adobe Photoshop. Such manipulations can be done for nefarious purposes to either hide or duplicate an object or similar content in the original images. A copy-move image forgery refers to a type of image manipulation where a source region is copied to another location within the same image. As a real-world example in Figure 1, copy-move image forgery could be used to hide an individual appearing a digital image, leading to a different interpretation. If such a manipulated image was part of a criminal investigation, without effective forensics tools the investigators could be misled. Therefore, it is crucial to develop a robust image forensic tool for copy-move detection and localization.

A number of copy-move detection approaches are already available including various traditional patch/block-based methods [8, 22, 14], keypoint-based methods [39, 23], irregular region-based methods [16, 26], and a few recent deep learning approaches [34, 19, 36]. Although some copy-move detection methods have been able to generate accurate localization result, but the results of these approaches are still far from perfect on some of the more challenging scenarios. For instance, in the case of when a uniform background region is copied to hide a foreground object, like the scenario shown in Figure 1.

In this paper, we propose a dual-order attentive Generative Adversarial Network (DOA-GAN) for copy-move forgery detection and localization. As illustrated in Figure 2, our generator is an end-to-end unified framework based on a deep convolutional neural network. Given an input image, we calculate an affinity matrix based on the extracted feature vectors. We design a dual-order attention module to produce the 1st-order attention map \(A_1\), which is able to explore the copy-move aware location information, and the 2nd-order attention map \(A_2\) to capture the patch

*This work was supervised by Chengjiang Long when Ashraful Islam was a summer intern at Kitware Inc.
inter-dependency. The final feature representation is formulated with these two attention maps, and then fed into a detection branch to output a detection confidence score and a localization branch to produce a prediction mask in which the source region and forged region are distinguishable. Meanwhile, the discriminator is designed to check whether the predicted mask is real or fake.

It is worth noting that our dual-order attention module is calculated based on the affinity matrix, which covers second-order statistics of features and plays a critical role for more discriminative representation [17, 9]. This motivates us to explore the second-order co-occurrence attention map $A_2$ for more discriminative feature representation. Also, we observe the maximum values in off-diagonal elements indicate high likelihood for copy-move spatial relations existing between patches. This observation inspires us to explore the 1st-order attention map $A_1$ to focus on the copy-move region aware feature representation. In this paper, we refine and normalize the affinity matrix, take top-$k$ values for each column and reshape to form a 3D tensor with $k$ channels. The tensor is then fed into simple convolutions to formulate our final 1st-order attention map $A_1$ which is able to give more attention to the source and forged region. To our knowledge, we are the first to extract the 1st-order attention map from the affinity matrix.

We adopt the adversarial training process [12] between the generator and the discriminator to generate a more accurate localization mask. As the number of epochs increases, both the generator and the discriminator improve their functionality so that the predicted mask iteratively becomes just like the ground-truth mask. Therefore, after a sufficiently large number of epochs leading to convergence in training, we can use the learned parameters in the generator to output a detection confidence score and to generate the prediction localization mask.

To summarize, our contributions are three-fold. (1) We propose a dual-order attentive Generative Adversarial Network (DOA-GAN) for image copy-move forgery detection and localization. (2) Our 1st-order attention module is able to extract the copy-move location aware attention map and the 2nd-order attention module explores pixel-to-pixel inter-dependence. These two attention maps ensure more discriminative feature representation for copy-move detection and localization. (3) The extensive experiments strongly demonstrate that our proposed DOA-GAN clearly outperforms state-of-the-art approaches in terms of both detection and localization quality.

2. Related work

Copy-move forgery detection and localization. A typical copy-move forgery detection approach [8] is composed of three stages: feature vector extraction, correspondence matching from the feature representation, and post-processing to reduce false alarms and improve detection rates. Patch/block-based methods include chroma features [3, 8], PCA feature [14], Zernike moments [29], blur moments [21], DCT [22]; keypoint-based methods such as SIFT [1, 7, 39], ORB [40], triangles [2], SURF [23, 30], and irregular region-based methods [16, 26]. Many traditional copy-move detection algorithms rely on strong assumptions about specific image characteristics like edge sharpness and local features. However, such assumptions are not always satisfied in the forged images, since other transformations like compression, resampling, or geometric transformations may hide traces of the manipulation.

Recently, deep neural networks have been applied to image forgery detection research [19, 13, 34, 36, 37, 4]. Especially, Wu et al. [36] introduced an end-to-end DNN solution to detect copy-move forged images with source/target localization with two separate branches. Unlike these DNN methods, our proposed DOA-GAN formulates both detection and localization as an end-to-end unified framework in the Generator network, where the 1st-order attention and the 2nd-order attention significantly improve the detection and localization performance.

Attentive Generative Adversarial Networks. Attention mechanisms have been successfully used in Generative Adversarial Networks [10, 38, 27]. Different from the existing attentive GANs, the dual-order attention module in our DOA-GAN is dependent on the affinity matrix calculated from contextual feature representation.

3. Method

The framework of the proposed approach is illustrated in Figure 2. The generator is an end-to-end unified framework to conduct both copy-move manipulation detection and localization tasks. Given an input image $I$, we first apply the first four blocks of a VGG-19 network to extract hierarchical features and resize them to the same size to form a concatenated feature $F_{cat}$. Then an affinity matrix is calculated, and the 1st-order attention map $A_1$ and the 2nd-order attention map $A_2$ are obtained via a dual-order attention module. Two atrous spatial pyramid pooling (ASPP) operation, i.e., ASPP-1 and ASPP-2, with different parameters, are applied to extract contextual features $F^{\text{aspp}}_{1}$ and $F^{\text{aspp}}_{2}$ which are multiplied element-wise with $A_1$ to get the possible copy-move regions attentive features $F^{\text{attn}}_{1}$ and $F^{\text{attn}}_{2}$. $A_2$ is used to obtain co-occurrence features $F^{\text{cooc}}_{1}$ and $F^{\text{cooc}}_{2}$. Both region attentive features and co-occurrence features are fused for the detection branch to produce a detection output score and for the localization branch to generate a mask. The discriminator is designed to check whether the predicted mask is real or fake. The alternative training between the generator and the discriminator enables more accurate results.
operation the image with itself. To resolve this issue, we define an
nal values indicate the correlation between the same part of
S
will have higher values along the diagonal, as the diago-
However, as we are calculating self-correlation of an image,
kernel. After that, we get the new affinity matrix
score between the same parts of the image using a Gaussian

Figure 3: The dual-order attention module to obtain the
copy-move region attention map \( A_1 \) and the co-occurrence
attention map \( A_2 \).

The Dual-Order Attention Module is designed as shown in Figure 3 to extract the copy-move aware region attention map \( A_1 \) and the co-occurrence attention map \( A_2 \). However, as we are calculating self-correlation of an image, \( S \) will have higher values along the diagonal, as the diagonal values indicate the correlation between the same part of the image with itself. To resolve this issue, we define an operation \( G \)
\[
G(i, j, i', j') = 1 - \exp\left( \frac{(i - i')^2 + (j - j')^2}{2\sigma^2} \right) \tag{2}
\]
and reshape it into \( hw \times hw \). \( G \) reduces the correlation score between the same parts of the image using a Gaussian kernel. After that, we get the new affinity matrix \( S' = S \odot G \), where \( \odot \) denotes the element-wise product.

We leverage the patch-matching strategy from [6]. We calculate the likelihood that a patch in the \( i \)-th column in \( S' \) by
\[
L^r(i, j) = \frac{\exp(\alpha S'[i, j])}{\sum_{j'=1}^{hw} \exp(\alpha S'[i, j'])}, \tag{3}
\]
\[
L^c(i, j) = \frac{\exp(\alpha S'[i, j])}{\sum_{j'=1}^{hw} \exp(\alpha S'[i', j])}, \tag{4}
\]
\[
L(i, j) = L^r(i, j)L^c(i, j) \tag{5}
\]
where \( \alpha \) is a trainable parameter, which is initialized as 3. \( L \) is the final affinity matrix.

From \( L \in \mathbb{R}^{hw \times hw} \), we extract the top-\( k \) values for each row, and reshape into \( T \in \mathbb{R}^{h \times w \times k} \). We feed \( T \) into an attention module. The attention module consists of three convolution blocks. The first two blocks contain convolution layers with 16 output channels and kernel size 3, followed by BatchNorm and ReLU. The final block contains two consecutive convolution layers with 16 output channels and kernel size 3, and 1 output channel and kernel size 1, respectively. We finally apply a sigmoid function to obtain the spatial copy-move aware attention map \( A_1 \in \mathbb{R}^{h \times w} \). As illustrated in Figure 4, the possible copy-move regions show higher values than other regions.

Figure 4: Visualization of \( A_1 \) on copy-move forgery images.

To make full use of the patch-to-patch inter-dependence, we normalize the affinity matrix in Equation 5 to obtain co-occurrence attention map \( A_2 \in \mathbb{R}^{hw \times hw} \),
\[
A_2(i, j) = \frac{L(i, j)}{\sum_{j'=1}^{hw} L(i, j')} \tag{6}
\]
This approach is inspired by [33].
Atrous Spatial Pyramid Pooling (ASPP) Block is used to extract contextual feature from the extracted features $F_{cat}$. ASPP block is utilized in DeepLab V3 [5] to capture context at several ranges for image segmentation. We found through experiments that two ASPP blocks are useful to learn two different tasks, namely source detection and target detection. The first ASPP block has atrous rates 12, 24, and 36, and the second block has atrous rates 6, 12, and 24. After the ASPP modules, we obtain two feature representations $F_{aspp}^1 \in \mathbb{R}^{h \times w \times d_s}$ and $F_{aspp}^2 \in \mathbb{R}^{h \times w \times d_s}$.

**Feature Fusion** is designed to merge both copy-move region aware attentive features and co-occurrence features.

The input to the discriminator network is the concatenation of $F_{aspp}^1$ and $F_{aspp}^2$ with the spatial copy-move region aware attention map $A_1$, and get

\[
F_{attn}^1 = F_{aspp} \odot A_1, \quad (7)
\]
\[
F_{attn}^2 = F_{aspp} \odot A_1 \quad (8)
\]

where $\odot$ is the element-wise product operation. We also obtain the co-occurrence features

\[
F_{cooc}^1 = A_2 \odot F_{attn}^1, \quad (9)
\]
\[
F_{cooc}^2 = A_2 \odot F_{attn}^2 \quad (10)
\]

where $\odot$ is the matrix product operation. Such a treatment fully explores the inter-dependence between patches, and distant pixels are able to contribute to the feature response at a location based on similarity metrics.

The final feature presentation is merged based on the above four features,

\[
F_{final} = \text{Merge}(F_{attn}^1, F_{attn}^2, F_{cooc}^1, F_{cooc}^2) \quad (11)
\]

where Merge is merge operation. In principle, any kinds of merge operation can be used. We use concatenation in this paper.

**Detection Branch and Localization Branch.** With the final presentation $F_{final}$, we design two convolution layers followed by two fully connected layers as the detection branch to output a detection score. At the same time, $F_{final}$ is fed into the localization branch, which consists of three convolution blocks, each followed by BatchNorm and ReLU, and a final convolution block of 3 channels to output the segmentation mask of pristine (background), source and target regions.

### 3.2. Discriminator Network

The structure of the discriminator is based on the PatchGAN discriminator [15]. Specifically, the discriminator is designed to predict whether each $N \times N$ patch in the image is real or fake. The discriminator is fully convolutional. It consists of five convolution blocks, each followed by BatchNorm and LeakyReLU, and a final convolution layer. The output channels of the convolution layers are 32, 64, 128, 256, 512, and 1, respectively, and the kernel size for all the convolution layers is 4 × 4. The stride of the convolution layers is 2 except the last one, which has a stride of 1. Therefore, as the input image is passed through each convolution block, the spatial dimension is decreased by a factor of two, and finally we get an output feature of size $H/16 \times W/16 \times 1$, where the spatial size of the input is $H \times W$.

The input to the discriminator network is the concatenation of the image $I \in \mathbb{R}^{H \times W \times 3}$ and mask $M \in \mathbb{R}^{H \times W \times 3}$. The discriminator is trained to discern the ground-truth mask from the predicted mask, while the generator tries to fool the discriminator.

### 3.3. Loss Functions

The loss function consists of adversarial loss, cross-entropy loss, boundary loss, and detection loss formulated as:

\[
L = L_{adv} + \alpha L_{ce} + \beta L_{bound} + \gamma L_{det} \quad (12)
\]

**Adversarial Loss.** The adversarial loss $L_{adv}$ is defined as:

\[
L_{adv}(G, D) = E(I,M)[\log(D(I, M))] + \log(1 - D(I, G(I))] \quad (13)
\]

where the discriminator $D$ tries to maximize the objective, and the generator $G$ tries to minimize it, i.e.,

\[
G^* = \arg\min_G \max_D L_{adv}(G, D) \quad (14)
\]

**Cross-Entropy Loss.** The cross-entropy loss $L_{ce}$ is expressed as:

\[
L_{ce} = \frac{1}{H \times W \times 3} \sum_{k=1}^{3} \sum_{i=1}^{H} \sum_{j=1}^{W} M(i, j, k) \log \hat{M}(i, j, k) \quad (15)
\]

where $\hat{M} = G(I)$ is the predicted mask of the generator network, and $M$ is the ground-truth mask.

**Boundary Loss.** Generally, the target copies contain boundary artifacts that can help in localizing the target copies easily from the background. To amplify this property of the target copies, we add a boundary loss $L_{bound}$ defined as:

\[
L_{bound} = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} M_{tb}(i, j) \log \hat{M}(i, j, 1) \quad (16)
\]

where $M_{tb}(i, j) \in \mathbb{R}^{H \times W}$ is the binary boundary mask of the forged region, which is obtained by using morphological operations on ground-truth target mask $\hat{M}(i, j, 1)$, and $\hat{M}(i, j, 1) \in \mathbb{R}^{H \times W}$ is the predicted mask of the target region.

**Detection Loss.** The detection loss $L_{det}$ is the binary cross-entropy loss between the image-level detection score...
from the detection branch and ground truth label (pristine or copy-move forged image):
\[
L_{\text{det}} = y_m \log(\hat{y}_m) + (1 - y_m) \log(1 - \hat{y}_m) \tag{17}
\]
where \( y_m \) is set to 1 if the image contains copy-move forgery, otherwise it is set to 0, and \( \hat{y}_m \) is the output from the detection branch.

### 3.4. Implementation Details

Our feature extraction module is based on the first three blocks of the VGG-19 network. We use the pretrained weights of VGG-19 trained on the ImageNet Dataset. The ASPP blocks are based on the ones used in DeepLabV3+ [5]. We set \( k = 10 \) for the top-k value in the 1st attention block.

We use two different learning rates for the generator and the discriminator networks, 0.001 and 0.0001 respectively, and the learning rate of the VGG-19 feature extractor is set to 0.00001. We decrease the learning rate by half when the training loss reaches plateaus after 5 epochs. We found that vanilla training of the GAN model does not work well, in particular the discriminator loss decreases to zero, whereas the generator loss increases. Therefore, we first optimize only the cross-entropy loss of the generator for 3 epochs, and then start optimizing all the losses. When the discriminator loss decreases to 0.3, we freeze the discriminator until the loss increases. In this way, we make sure that both the generator and the discriminator are learning at a similar pace, and the discriminator does not over-train.

### 4. Experiments

To verify the effectiveness of our DOA-GAN for copy-move forgery detection and localization, we conduct experiments on three benchmark datasets, *i.e.*, the USC-ISI CMFD dataset [36], the CASIA CMFD dataset [36], and the CoMoFoD dataset [31].

The USC-ISI dataset has 80,000 images for training, 10,000 images for validation, and 10,000 images for testing. The CASIA CMFD dataset contains 1313 forged images and their authentic counterparts (in total 2626 samples). The CoMoFoD dataset contains 5000 forged images, with 200 base images and 25 manipulation categories covering 5 kinds of manipulations and 5 different post-processing methods.

For evaluation on detection and localization performance, we report image-level (for detection) and pixel-level (for localization) precision, recall, and F1 score among 3 classes: pristine (background), source, and target, by averaging the score of each image. The unit is %.

### 4.1. Experiments on the USC-ISI CMFD dataset.

We train our proposed DOA-GAN with 80,000 copy-move forged images from USC-ISI dataset and 80,000 pristine images, and evaluate on the 10,000 testing forged images and 10,000 pristine images. The pristine images are collected from COCO dataset [18]. We compare against BusterNet [36] as a baseline, because to our best knowledge it is the only deep learning model that is able to distinguish between the copy source region and move forged region. To validate the effectiveness of the discriminator, we design several baselines, U-Net [28] baseline, DOA-GAN without any attention (denoted as NA-GAN), baselines using the 1st-order or 2nd-order attention only (denoted as FOA-GAN and SOA-GAN respectively). We also create another three baselines denoted as “DOA-GAN w/o \( L_{\text{adv}} \)”, (equivalent to DOA-CNN), “DOA-GAN w/o \( L_{\text{bound}} \)”, and “DOA-GAN w/o \( L_{\text{det}} \)”, by removing the loss functions \( L_{\text{adv}} \), \( L_{\text{bound}} \), and \( L_{\text{det}} \) in Equation 12, respectively. For fair comparison, we train the above-mentioned competing models with the same training images and evaluate on the held out testing images.

For pixel-level evaluation, we compute precision, recall,
and F1 score for each image, and finally report the average statistics. As F1 score is ill-defined for pristine images, the testing images for pixel-level evaluation include only the forged images. For image-level evaluation, we use both forged images and non-forged images (total 20k images). We predict an image to be forged if the output score from detection branch is greater than 0.5, otherwise it is predicted to be non-forged. For BusterNet and DOA-GAN w/o $L_{det}$, an image is forged if there are more than 200 pixels from the output mask predicted to be source or forged.

We summarize our detection results in Table 2 and the localization results in Table 1. From these two tables we can observe: (1) DOA-GAN w/o $L_{adv}$ performs better than BusterNet in terms of all the metrics, which clearly demonstrates promising performance for the generator in our proposed DOA-GAN; (2) DOA-GAN works better than DOA-GAN w/o $L_{adv}$ overall in both detection and localization tasks, which demonstrates the efficacy of the discrimination ability from the discriminator in our proposed DOA-GAN, (3) DOA-GAN performs better than DOA-GAN w/o $L_{bound}$ in all metrics. The difference is much apparent in pixel-level precision, recall, and F1 score for forged mask (T), which suggests that $L_{bound}$ increases the localization ability of target copies, (4) The detection and localization performance is worse in DOA-GAN w/o $L_{det}$ than that in DOA-GAN, especially in image-level precision, recall, and F1 score (Table 2), which demonstrates the efficacy of $L_{det}$, (5) FOA-GAN and SOA-GAN perform worse than the DOA-GAN in all metrics except F1 score of pristine pixels, which suggests the 1st-order and the 2nd-order attentions are complementary to each other to improve the performance on the copy-move forgery detection and localization, and (6) U-Net and NA-GAN baselines perform much worse than DOA-GAN, SOA-GAN, and FOA-GAN, specially in localization of source mask, which demonstrates the efficacy of affinity computation. This indirectly verifies the effectiveness of our dual-order attention module. To further understand the advantage of our DOA-GAN, we also provide some visualization results in Figure 5. As we can see, our DOA-GAN is able to generate more accurate masks than BusterNet, our FOA-GAN, and our FOA-GAN.

4.2. Experiments on the CASIA CMFD dataset.

Unlike the USC-ISI CMFD data, the CASIA CMFD dataset does not provide both ground-truth masks distinguishing source and target. It is more challenging because

<table>
<thead>
<tr>
<th>Methods</th>
<th>Precision</th>
<th>Recall</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>ManTra-Net</td>
<td>68.72</td>
<td>85.82</td>
<td>76.32</td>
</tr>
<tr>
<td>BusterNet</td>
<td>89.26</td>
<td>80.14</td>
<td>84.45</td>
</tr>
<tr>
<td>U-Net</td>
<td>82.61</td>
<td>66.13</td>
<td>73.46</td>
</tr>
<tr>
<td>NA-GAN</td>
<td>80.19</td>
<td>85.64</td>
<td>82.82</td>
</tr>
<tr>
<td>DOA-GAN w/o ASPP-1</td>
<td>95.31</td>
<td>80.13</td>
<td>87.02</td>
</tr>
<tr>
<td>DOA-GAN w/o ASPP-2</td>
<td>92.97</td>
<td>62.75</td>
<td>74.93</td>
</tr>
<tr>
<td>DOA-GNN w/o $L_{adv}$</td>
<td>95.45</td>
<td>93.09</td>
<td>94.25</td>
</tr>
<tr>
<td>DOA-GAN w/o $L_{bound}$</td>
<td>95.16</td>
<td>94.27</td>
<td>94.71</td>
</tr>
<tr>
<td>DOA-GAN w/o $L_{det}$</td>
<td>90.31</td>
<td>94.78</td>
<td>92.49</td>
</tr>
</tbody>
</table>

Table 2: The detection result on the USC-ISI CMFD dataset.

some uniform background is copied and pasted to the other background. To evaluate our proposed DOA-GAN on this dataset, we modify our network by replacing the final convolution layer of our network to a convolution layer of 1 channel output to get the mask of both copy and source parts as a single channel output. We train our model on the USC-ISI CMFD dataset and MS COCO dataset. For fair comparison, we do the same operation on BusterNet. In addition, we compare with four traditional copy-move forgery detection methods, i.e., a block-based CMFD with Zernike moment features (denoted as “Block-ZM”) [29], an adaptive segmentation based CMFD (denoted as “Adaptive-Seg”), a discrete cosine transform (DCT) coefficients based CMFD (denoted as “DCT-Match”), and a dense field-based CMFD (denoted as “DenseField”) [8]. We evaluate the pixel-level performance by computing precision, recall, and F1 score for each positive image where there is copy-move forgery, and report the final average. For image-level detection, we predict an image to contain forgery whenever there are more than 200 forged pixels in the output mask. We use both positive images and their authentic counterparts for image-level evaluation. All the images are resized to $320 \times 320$ before feeding into the models.

Table 3 shows performance comparisons with other baselines on CASIA CMFD dataset. As we can see, our proposed DOA-GAN performs the best in terms of all metrics except the precision in detection. This strongly demonstrates the promising advantages of our proposed method.

---

1With the implementation code publically available on https://github.com/MohsenZandi/Copy-Move_Forgery_Detection.
Figure 6: Visualization examples on the CASIA CMFD dataset. From left to right are the input image; results of Adaptive-Seg, DenseField, BusterNet, and our DOA-GAN; and the ground truth mask, respectively.

Figure 7: Visualization examples on the CoMoFoD dataset. From left to right are the input image; results of Adaptive-Seg, DenseField, BusterNet, and our DOA-GAN; and the ground truth mask, respectively.

Note that the result on BusterNet is different from the results reported in [36], as in the original BusterNet, the manipulation branch was trained on external image manipulation datasets, whereas, for fair comparison, we train both our model and BusterNet only on the above-mentioned copy-move datasets.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Year</th>
<th>Precision</th>
<th>Recall</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Block-ZM</td>
<td>2010</td>
<td>68.97</td>
<td>53.69</td>
<td>60.38</td>
</tr>
<tr>
<td>DCT-Match</td>
<td>2012</td>
<td>63.74</td>
<td>46.31</td>
<td>53.46</td>
</tr>
<tr>
<td>Adaptive-Seg</td>
<td>2015</td>
<td>93.07</td>
<td>25.59</td>
<td>40.14</td>
</tr>
<tr>
<td>DenseField</td>
<td>2015</td>
<td>99.51</td>
<td>30.61</td>
<td>46.82</td>
</tr>
<tr>
<td>BusterNet</td>
<td>2018</td>
<td>48.34</td>
<td>80.12</td>
<td>60.29</td>
</tr>
<tr>
<td>DOA-GAN</td>
<td>2019</td>
<td>58.82</td>
<td>82.14</td>
<td>68.55</td>
</tr>
</tbody>
</table>

Table 3: The performance on the CASIA CMFD dataset.

Figure 6 provides a visualization result, from which we can see our proposed DOA-GAN is able to detect more accurate masks than DenseField and BusterNet for the copy-move forgery manipulation.

4.3. Experiments on the CoMoFoD dataset.

We continue to evaluate the performance on the CoMoFoD dataset and report results in Table 4. Again, our proposed DOA-GAN achieves the best performance except the precision in detection and localization. Note that different types of transformations are applied in this dataset to create copy-move manipulated images, e.g., translation, rotation, scaling, combination, and distortion. Various post-processing methods, such as JPEG compression, blurring, noise adding, and color reduction, are also applied to all forged and original images. Taking each post-processing method as a specific attack, we use this dataset to further analyze the effects of our proposed DOA-GAN under different attacks.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Year</th>
<th>Precision</th>
<th>Recall</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Block-ZM</td>
<td>2010</td>
<td>51.72</td>
<td>20.87</td>
<td>29.74</td>
</tr>
<tr>
<td>DCT-Match</td>
<td>2012</td>
<td>50.48</td>
<td>29.77</td>
<td>37.46</td>
</tr>
<tr>
<td>Adaptive-Seg</td>
<td>2015</td>
<td>65.66</td>
<td>43.37</td>
<td>52.24</td>
</tr>
<tr>
<td>DenseField</td>
<td>2015</td>
<td>80.34</td>
<td>20.10</td>
<td>32.15</td>
</tr>
<tr>
<td>BusterNet</td>
<td>2018</td>
<td>53.20</td>
<td>57.41</td>
<td>55.22</td>
</tr>
<tr>
<td>DOA-GAN</td>
<td>2019</td>
<td>60.38</td>
<td>65.98</td>
<td>63.05</td>
</tr>
</tbody>
</table>

Table 4: The performance on the CoMoFoD dataset.

We provide a visualization example in Figure 7. Figure 9 shows the number of correctly detected images on CoMoFoD dataset under different types of attacks, where an image is correctly detected if its pixel-level F1 score is greater than 30%. Figure 8 shows F1 scores for all attacks. From these two figures, we can see that our proposed DOA-GAN is robust and consistently performs the best under all types of attacks.

4.4. Discussion

Our DOA-GAN is able to use the copy-move region attention to extract manipulation attentive features, as well as the co-occurrence feature with patch-to-patch interdependence taken into consideration. However, when the copy region is just extracted from the uniform background and pasted on the same background, our DOA-GAN may fail. It also might fail when the scale has been changed significantly. We provide two failure cases in Figure 10. As we see, the backgrounds for the first example are uniform, and the scale of the copy-move regions are very small in the second example.
5. Extension of DOA-GAN for Other Manipulation Types

Note that our proposed DOA-GAN is based on an affinity matrix calculated on the same image. It is easy to extend DOA-GAN to an affinity matrix calculated from two different images, i.e., donor image and probe image, and the corresponding manipulation types include image splicing and video copy-move.

For image splicing manipulation, we train DOA-GAN, and the two state-of-the-art approaches, DMVN [35] and DMAC [20], on the same synthetic image splicing dataset following the generation process described in [20] and then evaluate on the generated dataset from MS-COCO consisting of 42,093 testing image pairs. The results in Table 5 demonstrate DOA-GAN’s superiority in splicing localization.

For video copy-move manipulation, which can be considered as a inter-frame splicing between two consecutive frame sequences in a video. We evaluate our DOA-GAN’s performance on the video copy-move datasets generated from video object segmentation datasets including DAVIS [24], SegTrackV2 [32] and Youtube-object [25], and summarize the results in Table 6. Again, our proposed DOA-GAN works the best.

It is worth mentioning that due to page limit, we cannot provide sufficient technical details about extending DOA-GAN for image splicing and video copy-move. For more details of technical details, please refer to the supplementary.

### Table 5: Performance comparison of image splicing localization on the generated dataset from MS-COCO.

<table>
<thead>
<tr>
<th>Method</th>
<th>source</th>
<th>forge</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>IoU</td>
<td>F1</td>
</tr>
<tr>
<td>DMVN</td>
<td>37.2</td>
<td>48.4</td>
</tr>
<tr>
<td>DMAC</td>
<td>76.5</td>
<td>81.2</td>
</tr>
<tr>
<td>DOA-GAN</td>
<td>86.4</td>
<td>91.0</td>
</tr>
</tbody>
</table>

### Table 6: Performance Comparison on the generated video CMFD dataset in terms of pixel-level F1 score and IoU. Here, S, and T, and A denote source mask, target mask, and source-target agnostic mask, respectively.

<table>
<thead>
<tr>
<th>Method</th>
<th>F1 score</th>
<th>source</th>
<th>forge</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>S</td>
<td>T</td>
<td>A</td>
</tr>
<tr>
<td>DMVN</td>
<td>27.2</td>
<td>33.8</td>
<td>37.2</td>
</tr>
<tr>
<td>DMAC</td>
<td>39.5</td>
<td>39.0</td>
<td>45.2</td>
</tr>
<tr>
<td>DOA-GAN</td>
<td>62.9</td>
<td>62.3</td>
<td>65.0</td>
</tr>
</tbody>
</table>

6. Conclusion and Future Work

In this paper, we propose a dual-order attentive Generative Adversarial Network (DOA-GAN) for copy-move forgery detection and localization. The dual-order attention module is designed in the generator to extract the manipulation location aware attention map and the underlying co-occurrence relations among patches. The discriminator is to further confirm the accuracy of the prediction masks. Our proposed DOA-GAN has been experimentally proved to be able to produce more accurate copy-move masks and distinguish move forged regions from copy source regions. Our future work includes extending the current work to solve other challenging vision tasks like co-saliency detection and localization.
References


[32] David Tsai, Matthew Flagg, and James M.Rehg. Motion coherent tracking with multi-label mrf optimization. BMVC, 2010. 8


[38] Tao Xu, Pengchuan Zhang, Qiuyuan Huang, Han Zhang, Zhe Gan, Xiaolei Huang, and Xiaodong He. AttnGAN: Fine-grained text to image generation with attentional generative adversarial networks. In CVPR, pages 1316–1324, 2018. 2
