Mathematical Biosciences and Engineering

MBE, 16(5): 4581–4593
DOI: 10.3934/mbe.2019229
Received: 24 January 2019
Accepted: 26 April 2019
Published: 23 May 2019

http://www.aimspress.com/journal/MBE

Research article

Detection and localization of image forgeries using improved mask regional convolutional neural network

Xinyi Wang, He Wang, Shaozhang Niu*and Jiwei Zhang

Beijing Key Lab of Intelligent Telecommunication Software and Multimedia, Beijing University of Posts and Telecommunications, Beijing, China, 100876.

*** Correspondence:** Email: szniu@bupt.edu.cn.

Abstract: The research on forgery detection and localization is significant in digital forensics and has attracted increasing attention recently. Traditional methods mostly use handcrafted or shallow-learning based features，but they have limited description ability and heavy computational costs. Recently, deep neural networks have shown to be capable of extracting complex statistical features from high-dimensional inputs and efficiently learning their hierarchical representations. In order to capture more discriminative features between tampered and non-tampered regions，we propose an improved mask regional convolutional neural network (Mask R-CNN) which attach a Sobel filter to the mask branch of Mask R-CNN in this paper. The Sobel filter acts as an auxiliary task to encourage predicted masks to have similar image gradients to the groundtruth mask. The overall network is capable of detecting two different types of image manipulations, including copy-move and splicing. The experimental results on two standard datasets show that the proposed model outperforms some state-of-the-art methods.

Keywords: image forensics; copy-move forgery; splicing forgery; Mask R-CNN; sobel filter; edge detection

1. Introduction

The number of digital images has grown exponentially with the advent of new cameras, smartphones, and tablets. Social media such as Facebook and Twitter have further contributed to their distribution. However, digital content can be easily modified or tampered by photographic software such as Photoshop, Neoimaging, etc, which destroy the people's traditional concept of

"seeing is believing". There are certain types of manipulations such as copy-move, splicing that can easily deceive the human perceptual system. Once these fake images are maliciously used to mislead the public about the truth, it will be no doubt to seriously threaten the stability and development of the society. Therefore, how to identify the authenticity of digital images and conduct forensic analysis has become one of the important topics in diverse scientific and security/surveillance applications.

To authenticate a digital image, many techniques have been developed. In general, these techniques can be divided into two types, referred to passive detection techniques [1–3] and active detection techniques [4,5]. Active detection techniques embed particular data into the image. When verifying the authenticity of the image, the data is extracted from the suspicious image and compared to the original image. Compared with the active detection techniques, passive detection techniques can verify the authenticity of the image without the support of any additional pre-processing operation, which has attracted more and more attention recently.

Although any trace may not be left on the vision in tampered images, it would inevitably change the local or entire features of the image. Based on this idea, a large number of passive techniques have been developed to detect these images. The main two types of image forgery are copy-move and image splicing. Copy-move is the most generally used by attackers, where some parts of the image are copied and pasted to other parts of the same image. The primary mission is to detect if there exist two or more similar regions in a single image. Until now, a number of passive techniques have been developed to detect the copy-move forgery [6–8]. The primary mission of image splicing is to detect whether a given image is a composite one which is generated by cutting and joining two or more images. There are many studies on image splicing detection. For example, Shi et al. [9] proposed a natural image model for image splicing detection. Later, Zhao et al. [10] further developed a 2-D Markov model to characterize the underlying image dependency and achieved splicing localization. In image forensics above, most of the state-of-the-art image tampering detection approaches exploit the frequency domain characteristics and/or statistical properties of an image. At present, many traditional image forensics tasks can be solved by designing correct feature sets and then using these features to distinguish the original image from the processed image. Therefore, research usually focuses on the construction of complex handcrafted features. However, for many tasks, it is difficult to determine which features should be extracted.

Recently, deep learning has become popular due to its promising performance in different visual recognition tasks such as object detection [11,12], scene classification [13], and semantic segmentation [14]. Deep neural networks have shown to be capable of extracting complex statistical dependencies from high-dimensional sensory inputs and efficiently learning their hierarchical representations. Moreover, deep learning based approaches have been increasingly used in passive image forensics. Chen finished the first work of applying CNN in median filtering image forensics [15], then Qian proposed a new paradigm for steganalysis to learn features automatically via deep learning models [16]. Bayar *et al*. [17] changed the low pass filter layer to an adaptive kernel layer to learn the filtering kernel used in tampered regions. Rao *et al*. [18] presented a new image forgery detection method based on deep learning technique, which utilizes a convolutional neural network (CNN) to learn hierarchical representations from the input RGB color images automatically. P. Zhou *et al*. [19] proposed a two-stream Faster R-CNN network and trained it end-to-end to detect the tampered regions given a manipulated image. Most of these deep learning forensic techniques focus on single

tamper detection. Only a few can learn a more general forensics model, such as method [19], but the tampered area it locates is not pixel-level, and can only mark the tampered area with the bounding box.

To overcome these issues, we perform an end-to-end classification model trained by the Mask Regional Convolutional Neural Network (Mask R-CNN) [20] to distinguish manipulated regions from authentic regions and attach an Edge Agreement Head [21] to the mask branch of Mask R-CNN. Here this head uses traditional edge detection filter—Sobel kernel [22] on both the predicted mask and the groundtruth mask to encourage their edges to agree and improve detection accuracy. As the additional network head is only relevant during training, inference speed remains unchanged compared to Mask R-CNN. The overall framework (shown in Figure 1) is capable of detecting two types of image manipulations, including copy-move and splicing.

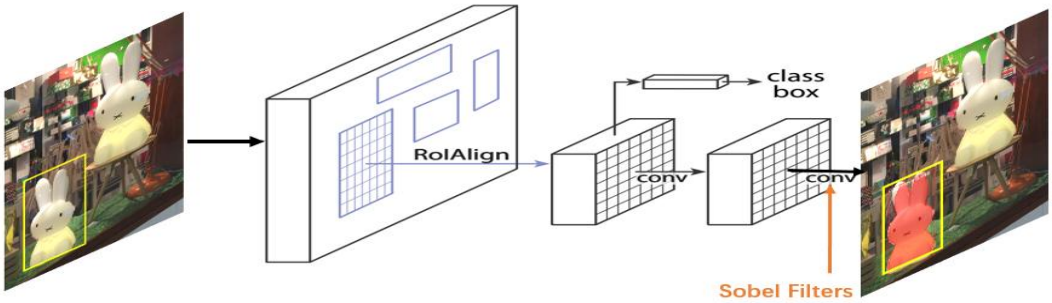

Figure 1. The framework of our method for forgery detection.

Our main contributions are as follows:

1. Apply the Mask R-CNN model to detect and locate the manipulated regions successfully. Pixel-level prior information of the tamper regions is utilized to provide the supervisory information for the training of Mask R-CNN.
2. Add a Sobel edge detection filter to focus on manipulated boundaries. This filter encourages predicted manipulated masks to have similar image gradients to the ground-truth mask and improves detection accuracy.
3. Create a synthetic tampering dataset based on COCO [23]. Since the previous image tampering datasets [24,25] are not enough to train a deep network.

2. The proposed method

To make full use of edge information and prior knowledge, a Mask R-CNN model and the Sobel filter are employed in the proposed method. A feature pyramid network (FPN) based on ResNet, and altered according to the images of the tampered area, is utilized as the backbone of the Mask R-CNN. Then, the Mask-RCNN is used to extract pyramid feature maps suitable for the images through the pixel-level prior information of the tamper region. Moreover, the recognition and coarse segmentation of the tamper region is performed, and the region of interest (RoI) is obtained by extending the bounding box of the tamper region provided by the recognition of Mask R-CNN. We also introduce a parameter-free network head, the Edge Agreement Head. This head uses traditional edge detection filter - Sobel filter on both the predicted mask and the groundtruth mask to encourage their edges to agree. The architecture of our method is shown in Figure 2.

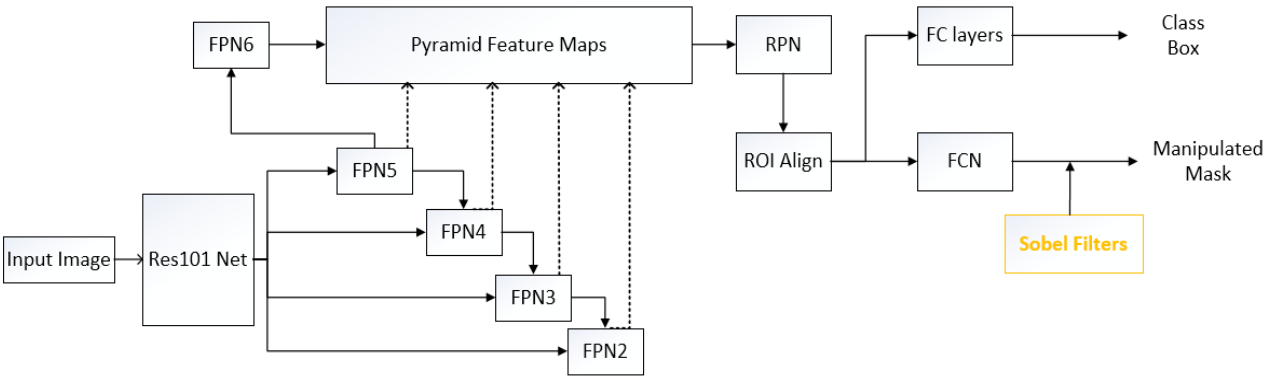

Figure 2. The Architecture of our proposed method.

2.1. *Mask-RCNN-based coarse detection and localization*

The Mask Regional Convolutional Neural Network (Mask R-CNN) is a simple but effective complement to the Faster R-CNN architecture which adds mask prediction, it replaces the classical RoI Pooling layer in Faster R-CNN network with RoI Align. RoI Align introduces an interpolation process, which can largely solve the alignment problem caused by direct sampling only through Pooling. On this basis, a parallel Fully Convolution Networks layer (FCN) is added [26]. FCN can be used to predict pixel-level instance masks. In addition to the mask branch, it also uses the Feature Pyramid Network (FPN) backbone [27]. With this addition, the network can perform a precise location using the high-resolution function maps in the lower layers, it can also use lower resolution semantics for more complex features. Compared with Faster R-CNN, only a small increase in expenditure can achieve the processing speed of 5FPS, and Mask R-CNN can be easily extended to other tasks, such as human attitude estimation. Without resorting to skills, the performance of each task is better than that of all single model detections at present.

In our method, the Mask R-CNN constructs three stages for coarsely tampering detection and localization: feature extraction, region proposal, and prediction. First, for an input image, Mask R-CNN uses RPN network to generate candidate region ROI, so the features are extracted by residual convolution network ResNet-101, then the pyramid feature maps of the image are obtained. The feature extraction process here is the same as that of Faster RCNN. The next step is to get the feature map of each ROI region in the image and correct each ROI using ROI Align. After getting the feature map of each ROI region, the classification and bounding box of each ROI are predicted. Each ROI uses the designed FCN framework to predict the category of each pixel in the ROI region, Finally, a rough segmentation result of the image tampering region is obtained.

The Mask R-CNN loss is a multi-task loss based on the Faster R-CNN loss, and its loss function is defined as

$$L_{MRCNN} = L_{Cl\,a\,s\,s} + L_B + L_{o\,xM} \tag{1}$$

By examining the predicted masks of the mask branches, we realize that these masks usually have blurred boundaries and do not follow the clear and fine contours of the original masks. When Mask R-CNN is used directly without adding edge detection, only coarse segmentation of the tampered area can be obtained, and its loss function is the same as that in [20].

2.2. *Edge detection using sobel filter*

When training a Mask R-CNN for image forgery detection and segmentation, we can always observe incomplete or poor masks, especially during early training steps (shown in Figure 3). Furthermore, the masks often do not follow the real tamper boundaries.

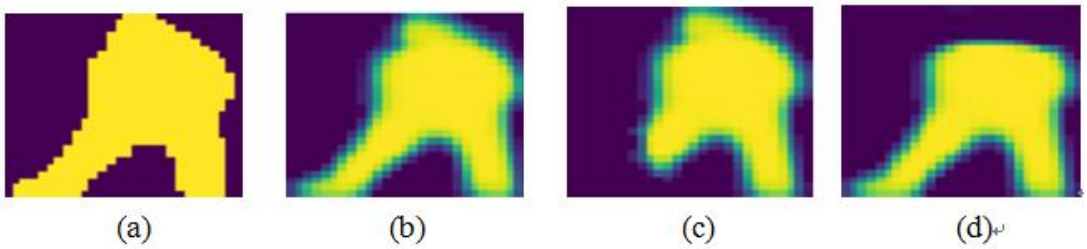

Figure 3. Overview of different example masks to illustrate the effect of the Edge Agreement Loss.

In Figure 3, (a) corresponds to the groundtruth, and from (b) to (d) represent three example mask predictions which demonstrate early-stage predictions of the Mask R-CNN during training. The Figure 3 shows possible mistakes such as missing parts or over segmentation in training. To overcome this problem, we need to find mask edge to supervise the training of Mask R-CNN, so we consider using an edge detection filter.

The edge detection is used to identify the points with significant brightness changing in a digital image. It is the basic process of image processing and computer vision. The application of edge detection acquires edge information better, so we combine it with the Mask RCNN. It can encourage predicted masks to have similar image gradients to the groundtruth mask, thus the tampering region segmentation will be better.

There are many ways to perform edge detection. However, most may be grouped into two categories, Gradient and Laplacian. The gradient method detects the edges by looking for the maximum and minimum in the first derivative of the image. The Laplacian method searches for zero crossings in the second derivative of the image to find edges. The edge detection filter which can be described as a convolution with a 3×3 kernel, such as the Sobel and Laplacian filters. In this paper, we choose the Sobel filter. Here, the Sobel filter is an Edge Agreement Head which attaches to the mask branch of Mask R-CNN (shown in Figure 2).

The Sobel operator is a directional algorithm that includes two operators corresponding to horizontal edges and vertical edges detection. It is not a simple average or difference, but a center with a weight of four directions.

$$\begin{aligned} f_x^{'} &= f(x-1,y+1)+2f(x,y+1)+f(x+1,y+1)-f(x-1,y-1)-2f(x,y-1)-f(x+1,y-1) \\ f_y^{'} &= f(x-1,y-1)+2f(x-1,y)+f(x-1,y+1)-f(x+1,y-1)-2f(x+1,y)-f(x+1,y+1) \\ G[f(x,y)] &= \left|f_x^{'}(x,y)\right|+\left|f_y^{'}(x,y)\right| \end{aligned} \tag{2}$$

Where $f_x^{'}(x,y)$, $f_y^{'}(x,y)$ represent the first derivative of X and Y directions respectively. $G[f(x,y)]$ is the gradient of Sobel filter, and $f(x,y)$ is the input image with integer pixel coordinates.

The Sobel filter has two filters, include two groups of 3 × 3 matrix in lengthways and transverse directions, two out of three-dimensional matrix can be expressed as follows:

$$G_x = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}, \qquad G_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \tag{3}$$

The overall gradient is calculated on the basis of lengthways and transverse gradients:

$$G = \sqrt{G_x^2 + G_y^2}$$
$$\theta = arc\tan\frac{G_y}{G_x} \tag{4}$$

If θ equals zero, it means that the image has a lengthways edge here and the left side is dimmer than the right side.

2.3. *Loss Construction*

We attach the Sobel filter as an Edge Agreement Head to the mask branch of Mask R-CNN, which results in the construct of an auxiliary loss called Edge Agreement Loss (L_{edge}), the Edge Agreement Loss is computed using L^2-norm loss function.

$$L_{edge} = \frac{1}{n}\sum_{i=1}^{n}\left(y - \hat{y}\right)^2 \tag{5}$$

L_{edge} reflects the difference between the target $\hat{y}$ and the prediction y, which is shown in Fig 4.

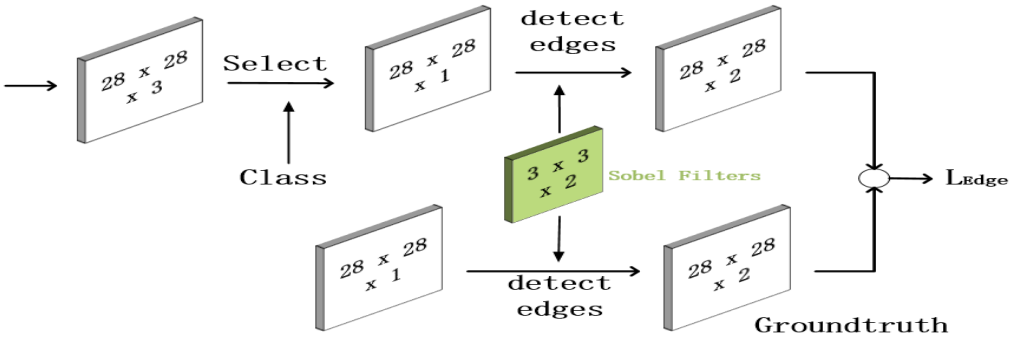

Figure 4. Edge Agreement Head: We extend the existing mask branch architecture. The head computes a convolution of the selected mask and the groundtruth mask with the 3 × 3 × 2 dimensional edge detection filter (the green part), and the L_{Edge} loss is calculated between these.

Thus, we use the following formula to represent the total loss.

$$L_{Total} = L_{MRCNN} + L_{Edge} = L_{cls} + L_{box} + L_{mask} \tag{6}$$

The total loss L_{Total} consists of the original Mask R-CNN loss L_{MRCNN} (eq. 1) and the new Edge Agreement Loss L_{Edge}. The classification loss L_{Class} and bounding-box loss L_{Box} are the same as those defined in [20]. The branch of the mask has a Km^2 - dimension output for each RoI, and its encoding resolution is K binary masks of $m \times m$, and each K class corresponds to one. For this purpose, we use sigmoid per pixel and define L_{Mask} as the average binary cross-entropy loss. For the RoI associated with the ground reality class k, L_{Mask} is defined only on the K mask (other mask outputs do not cause loss).

3. Implementation detail

All training images are sized to maintain their aspect ratio. The mask size is 28 × 28 pixels, and the image resolution is 1024 × 512 pixels. This method differs from the one used in the original Mask R-CNN [20], where resizing is done such that the smallest size is 800 pixels and the largest is trimmed at 1024 pixels. We set the hyperparameter according to the characteristics of the method and the object detection in original papers [20]. The anchors are selected based on the intersection over union (IoU) ratio of the anchor and ground-truth (GT) boxes and the mask loss is only defined on the positive ROI. The mask target is the intersection between the ROI and its associated ground truth mask. Here, the RoI is considered positive if it has IoU with a ground-truth box of at least 0.5 and negative otherwise.

Each mini-batch has 2 images per GPU, and each image has an ROI of N samples with a plus or minus ratio of 1:3. For the C4 backbone, N is 64, and for FPN, N is 512. A batch size of 2 on a single GPU machine for 640K iterations with a learning rate of 0.01 and a 10 reduction at 240K iterations. The optimization is done by SGD with momentum set to 0.9 and weight decay set to 0.0001.

4. Experimental results

In this Section, experimental results are presented to demonstrate the efficiency of our method of tampering detection and localization. As mentioned above, we introduce an Edge Agreement Head, which uses Sobel filter on both the predicted mask and the groundtruth mask to encourage their edges to agree. Therefore, we want to verify whether the segmentation accuracy is improved after adding edge detection.

All experiments are conducted using NVidia GeForce GTX 2080 Ti with 11 GB memory in Ubuntu 16.04, the operating environment is Intel Core CPU i7-9700K, GeForce GTX 2080Ti with 32 GB RAM.

4.1. *Pre-trained Model*

Current standard image tampering datasets do not contain enough images for deep neural network training. To overcome this problem, we create a synthetic dataset using the images from COCO [23]. We pre-train our model on our synthetic dataset and use the segmentation annotations to randomly select a different kind of objects. Then we copy and paste them to the same or other images. The tampered images in our synthetic dataset are divided into two classes: (a) copy-move, (b) splicing. Separate the training (80%) and the test set (20%) to ensure that the same background and tampered objects do not appear in the training and test set. At last, we create 30K tampered and

authentic image pairs and train our model end-to-end on this synthetic dataset. We use Average Precision (AP) for evaluation, the metric of which is the same as COCO [22] detection evaluation. In Table 1, we can see that the Mask R-CNN with the added sobel filter performs better than the single Mask R-CNN.

Table 1. AP comparison on our synthetic COCO dataset.

AP	synthetic test
Single Mask R-CNN	0.713
Mask R-CNN+Sobel filter	0.769

4.2. *Testing on standard datasets*

4.2.1. Dataset and evaluation metrics

We compare our method with current state-of-the-art methods on Cover [24] and Columbia dataset [25]. The Cover dataset [24] is a dataset focusing on copy-move and it covers similar objects as the pasted regions to conceal the tampering artifacts. The Columbia dataset [25] focuses on splicing based on uncompressed images. Ground-truth masks of these two standard datasets are provided. The CASIA dataset [25] contains more tampering images. However, it does not provide the corresponding Ground-truth masks, so we don't choose it in this paper.

The evaluation metrics standards are AP (averaged over IoU thresholds), AP_{50}, AP_{75} (AP at different scales) and F_1 (a pixel localization metric) score. AP is evaluated using mask IoU and the F_1 metric is defined as below:

$$F_1 = \frac{2 \cdot TP}{2 \cdot TP + FN + FP} \tag{6}$$

TP represents the number of pixels classified as true positive and *FN* represents the number of pixels classified as false negative where a tampered pixel is incorrectly classified as authentic, and *FP* represents the number of pixels classified as false positive where an authentic pixel is incorrectly classified as tampered.

4.2.2. Performance Comparison

We evaluate the performance of the improved Mask R-CNN model and compare it with the existing baseline approaches [19–30] on the same training and testing split protocol as [31] (for COVER) and [32] (for Columbia).

Table 2. F_1 score comparison on two standard datasets.

Methods	**Columbia**	**Cover**
ELA [28]	0.470	0.222
NOI1[29]	0.574	0.269
CFA1 [30]	0.467	0.190
RGB-N [19]	0.697	0.437
Single Mask R-CNN	**0.7405**	**0.530**
Mask R-CNN+Sobel filter(proposed)	**0.7825**	**0.612**

The average F_1 score is calculated according to the evaluation metric for each method. The results are shown in Table 2. Obviously, the proposed method outperforms the existing baseline methods in F_1 score. When the Sobel Filter is added, the F_1 score is also improved compared to the single Mask R-CNN.

Table 3. AP comparison on two standard datasets using Mask R-CNN with the Sobel edge detection filter.

	Cover	Columbia	Mean
AP	0.936	0.978	0.957

Tamper detection and localization results of the Mask R-CNN with Edge Agreement Head using the Sobel Edge Detection filter are shown in Figure 5. The first and second rows of Figure 5 show the detection results of copy-move and splicing tampering respectively. We can see our proposed method produces accurate results for copy-move and splicing tampering detection. By attaching the Edge Agreement Head to the Mask R-CNN, the network also produces the correct classification for different types of forgery. We change the classes for manipulation classification to be splicing and copy-move to learn distinct visual tampering artifacts and features for each class. The detection performances of the two types of tampering are shown in Table 3.

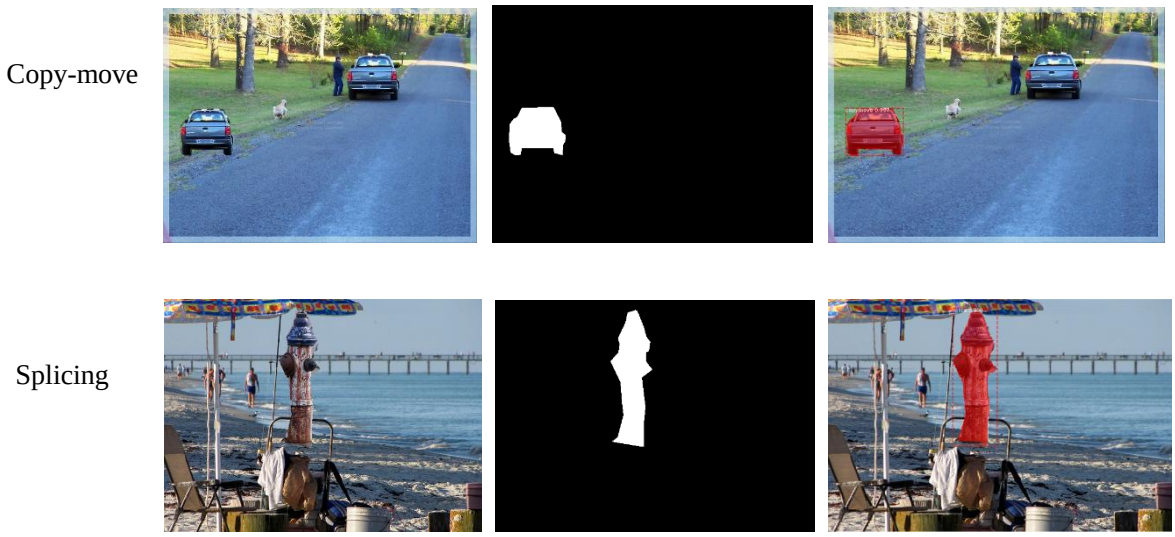

Figure 5. Detection results of the two-class tampered images. The first column is the tampered images. The second column is the mask of groundtruth. The last column is the detection results.

We consider that the superiority of the Sobel filter can be explained by its structure. Since it consists of two filters, not only the edge strength along the x and y-axes can be used in the gradient descent process, but also the direction of the edge can be used to minimize the total loss, So this extra information can speed up training and improve the tamper accuracy. Table 4 shows the AP metrics comparisons on training sets of before and after adding Sobel filter to the network structure, and they are 320, 500 and 640 steps respectively.

Table 4. Comparison of the manipulated region segmentation mask AP metrics of our best performing model with the single Mask R-CNN model after an extended training duration.

Methods	Steps	AP	AP_{50}	AP_{75}
Mask R-CNN	320k	73.2±0.09	84.3±0.29	72.5±0.06
+Sobel filter	500k	74.5	84.9	74.4
(Proposed)	640k	**75.4.**	**86.1**	**76.8**
Single	320k	72.6±0.15	81.4±0.23	71.2±0.11
Mask R-CNN	500k	73.1	82.6	73.7
	320k	73.4	83.3	74.3

When the training time is longer, the difference between the single Mask R-CNN and the edge protocol header still exists, demonstrating the effectiveness of the additional loss not only in the early stages of training but also in subsequent steps. We observe that the edge contour of the detected tampering area is more accurate after increasing the edge agreement loss to train the model. The contrast effects are shown in Figure 5.

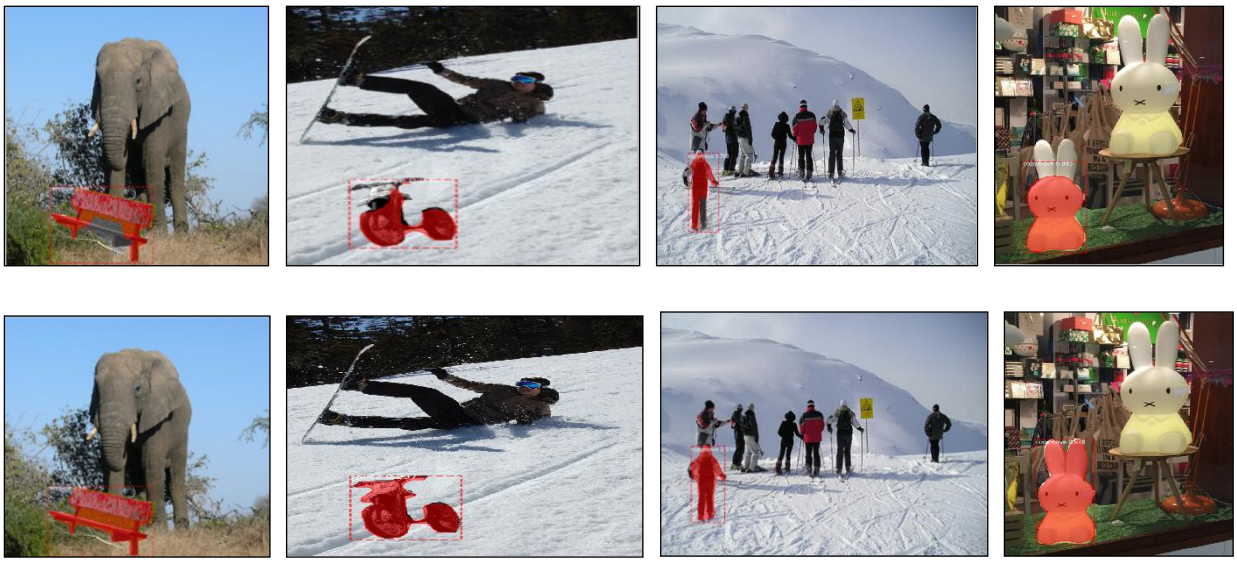

Figure 5. Comparison of segmentation results in tampering regions by Single Mask R-CNN (the first row) and Mask R-CNN with Edge Agreement Head using the Sobel edge detection filter (the second row).

4.3. *Robustness*

Considering tampering images are often attacked by JPEG compression and Resizing, we test the robustness of the proposed method and compare with two methods in Table 5. Experimental results show our approach is more robust to these attacks and better than other methods.

Table 5. F_1 score on test dataset for JPEG compression (with quality 90 and 70) and resizing (with scale 0.9 and 0.7) attacks. Each entry is the F_1 score of JPEG/Resizing.

JPEG	100/1	90/0.9	70/0.7
ELA [28]	0.305/0.305	0.221/0.245	0.175/0.188
NOI1 [29]	0.347/0.347	0.261/0.275	0.230/0.244
Our method	**0.633/0.633**	**0.562/0.580**	**0.543/0.564**

5. Conclusion

In this paper, we analyze the behavior of the Mask R-CNN network in the early training steps. By observing the prediction mask of the mask branches, we recognize that these often exhibit blurred boundaries, sometimes not following the clear and complete contours of the original tamper-area mask. To improve the accuracy of tamper localization, we introduced a parameter-free network head that applies the Sobel edge detection filter to the mask to calculate the L^2 loss between the predicted and groundtruth mask contours. We demonstrate the superior performance of the proposed method over other state-of-the-art image tampering detection methods. More features will be explored in the future.

Acknowledgments

This work is supported by National Natural Science Foundation of China (No. 61370195, U1536121).

Conflict of interest

The authors declare no conflict of interest.

References

1. W. Luo, Z. Qu, F. Pan, et al., A survey of passive technology for digital image forensics, *FCS*, **1** (2007), 166–179.
2. J. G. R. Elwin, T. Aditya and S. M. Shankar, Survey on passive methods of image tampering detection, *INCOCC*, Erode, **2**(2010), 431–436.
3. G. K. Birajdar and V. H. Mankar, Digital image forgery detection using passive techniques: A survey, *Digit. Inve*, **10** (2013), 226–245.
4. L. Verdoliva, D. Cozzolino and G. Poggi, A feature-based approach for image tampering detection and localization, *WIFS*, Atlanta, *GA*, (2014), 149–154.
5. U. H. Panchal and R. Srivastava, A comprehensive survey on digital image watermarking techniques, *ICCSNT*, (2015), 591–595.
6. Al-Qershi, M. Osamah and B. E. Khoo, Passive detection of copy-move forgery in digital images: State-of-the-art, *Foren. Sci. Int.*, **23** (2013), 284–295.
7. H. Huang, W. Guo and Y. Zhang, Detection of copy-move forgery in digital images using SIFT algorithm, *CIIA*, (2008), 272–276.

8. T. Mahmood, A. Irtaza, Z. Mehmood, et al., Copy-move forgery detection through stationary wavelets and local binary pattern variance for forensic analysis in digital images, *Forensic Sci. Int.*, **279** (2017), 8–21.
9. Y. Q. Shi, C. Chen and C. Wen, A natural image model approach to splicing detection, In *Proceeding MM&Sec '07 Proceedings of the 9th workshop on Multimedia & security*, (2007), 51–62.
10. X. Zhao, S. Wang, S, Li, et al., Passive image-splicing detection by a 2-D noncausal markov model, *IEEE Trans. CSVT*, **25** (2015), 185–199.
11. R. Girshick, Fast R-CNN, *ICCV*, (2015), 1440–1448.
12. B. H. Jawadul and A. K. Roy-Chowdhury, CNN based region proposals for efficient object detection, *ICIP*, (2016), 3658–3662.
13. B. Zhou, A. Lapedriza, J. Xiao, et al., Learning deep features for scene recognition using places database, *ANIPS*, **1**(2015), 13–20
14. L. Jonathan, E. Shelhamer and T. Darrell, Fully convolutional networks for semantic segmentation, *IEEE Trans. Pattern Anal. Mach. Intel.*, **39** (2014), 640–651.
15. J. Chen, X. Kang, Y. Liu, et al., Median filtering forensics based on convolutional neural networks, *IEEE Sig. Pro. Lett.*, **22** (2015), 1849–1853.
16. Y. Qian, J. Dong, W. Wang, et al., Deep learning for steganalysis via convolutional neural networks, *Pro. SPIE. ISOE*, **94** (2015), 9–14.
17. B. Belhassen and M. C. Stamm, A deep learning approach to universal image manipulation detection using a new convolutional layer, *IH&MMSec*, (2016), 5–10.
18. R. Yuan and J. Ni, A deep learning approach to detection of splicing and copy-move forgeries in images, *WIFS*, (2016), 1–6.
19. P. Zhou, X. Han, V. I. Morariu and L. S. Davis, Learning rich features for image manipulation detection, *IEEE Conf. Comput. Vis. Pattern Recognit.*, Salt Lake, (2018), 1053–1061.
20. K. He, G. Gkioxari, P. Doll ár, et al., Mask R-CNN. *ICCV*, **99** (2017), 1–11.
21. R. S. Zimmermann and J. N. J. a. p. a. Siems, Faster Training of Mask R-CNN by Focusing on Instance Boundaries, *arXiv:* 1809.07069.
22. C. Lopez-Molina, H. Bustince, J. Fern ández, et al., A t-norm based approach to edge detection, *IWCANN*, Springer, Berlin, Heidelberg, (2009), 302–309.
23. T.Y. Lin, M. Maire, S. Belongie, et al., Microsoft coco: Common objects in context, *ECCV*, Springer, Cham, (2014), 740–755.
24. B. Wen, Y. Zhu, R. Subramanian, et al., Coverage novel database for copy-move forgery detection. *ICIP*, (2016), 161–165.
25. Y. F. Hsu and S. F. Chang, Detecting image splicing using geometry invariants and camera characteristics consistency, *ICME*, (2006), 549–552.
26. L. Jonathan, E. Shelhamer and T. Darrell, Fully Convolutional Networks for Semantic Segmentation, *CVPR*, **39** (2015), 3431–3440.
27. T.Y. Lin, P. Dollar, R. Girshick, et al., Feature pyramid networks for object detection, *CVPR*, (2017), 2117–2125.
28. N. Krawetz and H. F. J. H. F. S. Solutions, A Picture's Worth, *Hacker Fact. Solut.*, **6** (2007), 1–31.
29. B. Mahdian and S. Saic, Using noise inconsistencies for blind image forensics, *Image Vis. Comput.*, **7** (2009), 1497–1503.

30. P. Ferrara, T. Bianchi, A. De Rosa, et al., Image forgery localization via fine-grained analysis of CFA artifacts, *IEEE Trans. Inf. Foren. Secur.*, **7**(2012), 1566–1577.
31. J. H. Bappy, A. K. Roy-Chowdhury, J. Bunk, et al., Exploiting spatial structure for localizing manipulated image regions, *ICCV*, (2017), 4970–4979.
32. R. Salloum, Y. Ren and C.-C. J. Kuo, Image splicing localization using a multi-task fully convolutional network (MFCN), *J. Vis. Com. Image Repre.*, **51** (2018), 201–209.

AIMS Press