LOCALIZATION OF IMAGE SPLICING UNDER SEGMENT ANYTHING MODEL WITH INTEGRATED COMPRESSION AND EDGE ARTIFACTS

Ruhao Zhao[1], *Xian Zhong*[1,3], *Liang Liao*[4] , *Wenxuan Liu*[1], *Wenxin Huang*[2,*], *and Zheng Wang*[5]

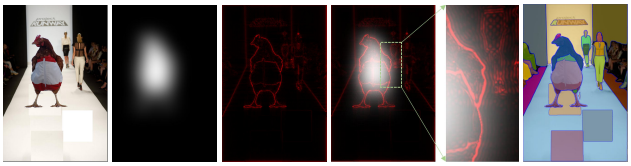
[1] Hubei Key Laboratory of Transportation Internet of Things,
School of Computer Science and Artificial Intelligence, Wuhan University of Technology
[2] School of Computer Science and Information Engineering, Hubei University
[3] Rapid-Rich Object Search Lab,
School of Electrical and Electronic Engineering, Nanyang Technological University
[4] College of Computing and Data Science, Nanyang Technological University
[5] National Engineering Research Center for Multimedia Software,
School of Computer Science, Wuhan University

ABSTRACT

The localization of image splicing involves identifying pixels in an image that have been spliced from other images, necessitating the discernment of splicing features. Despite significant advancements driven by the rise of social media and deep learning, existing methods exhibit limitations, often neglecting the integration of coarse and precise features and lacking the ability to understand objects. This leads to erroneous predictions in identifying spliced regions. This paper proposes Segment Anything Model with Integrated Compression and Edge artifacts (SAM-ICE) for the localization of image splicing, addressing these limitations by fusing forged edge features and compression artifact features. Leveraging SAM's object understanding ability, our method identifies spliced regions using the fused features as guidance. Specifically, we employ Edge Artifact Extractor (EAE) to extract fine high-frequency edge splicing features and Compression Artifact Extractor (CAE) to extract coarse compression artifact features. By combining these features, our method utilizes coarse-fine features to accurately pinpoint the spliced portions of the image. Experimental results demonstrate the superior accuracy, robustness, and generalizability of our method compared to the state-of-the-arts.

Index Terms— Image Splicing, Forgery Localization, Coarse-fine Feature Fusion, Edge Splicing Features, Compression Artifacts

∗ Corresponding author: wenxinhuang_wh@163.com. This work was supported in part by the National Natural Science Foundation of China under Grant 62301213, 62271361, and 62202349. The numerical calculations in this paper have been done on the supercomputing system in the Supercomputing Center of Wuhan University.



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

Fig. 1: Motivation for locating image splicing. (a) depicts the spliced image, (b) illustrates the coarse compression artifact, (c) showcases the fine edge artifact, (d) demonstrates the fusion of these two artifacts, and (e) reflects the ability of SAM to understand objects. Through the fusion of these artifacts, the method is directed to accurately and completely identify the spliced region.

1. INTRODUCTION

Image forgery poses significant challenges across various fields, *e.g.*, law enforcement, journalism, and social media. The advent of digital technology has revolutionized multiple aspects of our lives, encompassing image creation and sharing. However, this progress has concurrently given rise to a novel form of deception known as image splicing forgery, significantly undermining the trustworthiness of digital images. Image splicing involves pasting a region from one image onto another, prompting the need to precisely identify the tampered regions in some forgery images. This is essential for tasks, such as repairing the forgery image or other [1, 2, 3, 4] related purposes.

To identify forgery region in spliced images, detecting splicing traces and forgery features introduced by forgery tools is crucial. Some methods have been proposed to capture image-specific features and locate forged regions. Huh *et al.* [5] employ a self-consistency method to learn forgery

features. Similarly, Wu *et al.* [6] devise a self-supervised learning task for learning traces of image operations. Agrawal *et al.* [7] locate the forged region by capturing the consistency signature of the image. Kumar *et al.* [8] predict forged regions by extracting salient key-points. Some methods predict tampering positions through feature fusion. Kwon *et al.* [9] employ a dual-channel HRNet to extract compression artifacts and RGB features for locating forged positions. While these methods can locate forged regions to some extent, they lack object understanding and are insufficient in extracting forgery feature types. Consequently, this leads to incomplete results when splicing complex content.

As illustrated in Fig. 1(b), image compression leaves artifacts in the spliced region, aiding in rough localization. Through frequency domain analysis, we find that there are high-frequency splicing artifacts in the spliced images, which can precisely locate the splicing edge as Fig. 1(c). To leverage both features, as shown in Fig. 1(d), we fuse them to identify spliced objects within the image. Notably, the Segment Anything Model (SAM) [10] has exhibited robust object understanding abilities, demonstrating strength in robustness and generalization. Inspired by SAM's success, we utilize its ability of object understanding to fuse precise edge features with coarse compressed features, facilitating accurate and complete prediction of the splicing region as Fig. 1(e).

Building upon the aforementioned insights, we propose a splicing prediction method, that Integrates Compression and Edge artifacts features using SAM (SAM-ICE). To leverage the edge features retained in the high-frequency region by the forgery tool, we employ Fast Fourier Transform (FFT) [11] to transform the image to frequency domain for locating precise splicing contours, facilitating the extraction of forgery feature frequencies. However, for images with smooth forgery boundaries, extracting useful forgery information from the image frequency becomes challenging. Recognizing that the splicing region may exhibit a different distribution of Discrete Cosine Transform (DCT) coefficients in the Y-channel compared to the authentic region [9], we utilize this characteristic to extract forgery features, Integrating these two features into SAM allows us to predict the splicing region.

In summary, our main contributions are threefold:

- We extract coarse compression artifacts and fine splicing artifacts, generate fused features through the combination of coarse and fine granularity features, and lead to locate the splicing position.
- We propose Segment Anything Model with Integrated Compression and Edge artifacts (SAM-ICE), guiding at the feature level. The fused features are fed into each layer of SAM for guidance and feature calibration. To the best of our knowledge, we are the first to employ SAM for coarse-fine feature fusion in the splicing forgery localization task.
- Experimental results demonstrate our superiority overstate-of-the-art methods, excelling in locating various splicing forgery features, even those unseen during training.

2. PROPOSED METHOD

2.1. Framework Overview

As depicted in Fig. 2, we incorporate extracted edge artifact and compression artifact features into every layer of SAM [10]'s image encoder. This integration enables precise identification of the splicing position, enhancing the method's effectiveness in learning.

2.2. Edge Artifact Extractor

To capture edge artifact information within the forged image, we develop an Edge Artifact Extractor (EAE) utilizing Fourier Transform to extract specific forgery details from frequency domain of image [12].

We initiate the process by FFT to transform the input image If into frequency domain:

$$I_r, I_i = \mathrm{FFT}_h\left(I_f\right), \tag{1}$$

where $\mathrm{FFT}_h(\cdot)$ denotes extracting of high-frequency components from the image using FFT. I_r and I_i represent the real and imaginary parts, respectively, after the Fourier transform.

Subsequently, the edge forgery features are extracted from frequency domain through convolution:

$$F_r, F_i = \mathrm{Split}\left(\mathrm{ReLU}\left(\mathrm{BN}\left(\mathrm{Conv}\left(\mathrm{Concat}\left(I_r, I_i\right)\right)\right)\right)\right), \tag{2}$$

where ReLU refers to Rectified Linear Unit function, BN signifies Batch Normalization operation, Conv represents Convolution operation, $\mathrm{Concat}(\cdot)$ operation combines the two tensors I_r and I_i together, and $\mathrm{Split}(\cdot)$ splits the tensor into two at the corresponding position. F_r and F_i denote the real and imaginary parts of the edge forgery feature extracted from frequency domain, respectively.

Finally, we transform the edge forgery features acquired in frequency domain back into the image space domain:

$$F_e = \mathrm{iFFT}\left(F_r, F_i\right), \tag{3}$$

where $\mathrm{iFFT}(\cdot)$ denotes inverse FFT, and F_e represents final extracted edge feature, which serves as the input to SAM [10].

2.3. Compression Artifact Extractor

Owing to the distribution difference between the forged image region and the clean region, specific artifacts emerge. Consequently, we employ a Compression Artifact Extractor (CAE) to extract compression artifacts.

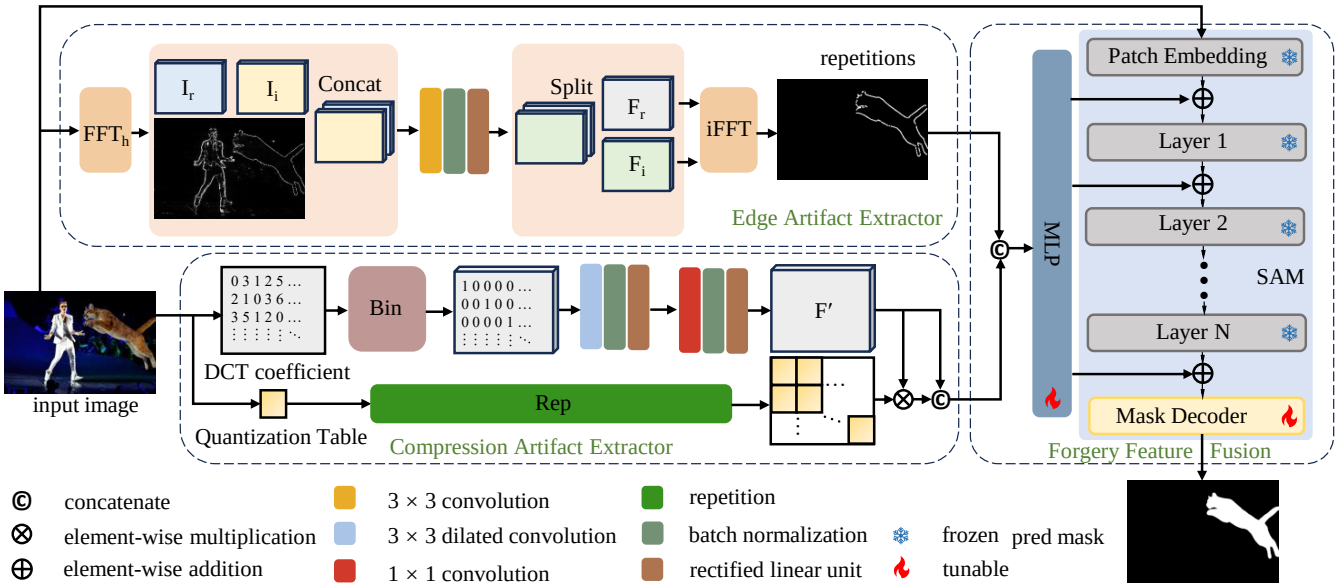


Fig. 2: Proposed framework for image splicing localization. Compression artifact features and edge splicing features are separately extracted from the spliced images. Subsequently, these two features are fused and fed into each layer of SAM.

Specifically, considering the substantial difference in DCT distribution histogram [9] of the Y-channel between the spliced and clean regions, we utilize Binarization (Bin) to process DCT coefficients (D), simulating DCT distribution histogram:

$$M_{i,j,t} = \begin{cases} 1, & \left|\text{Clip}_{[-T,T]}\left(D_{i,j}\right)\right| = t, \\ 0, & \text{Otherwise}, \end{cases} \tag{4}$$

where $\text{Clip}_{[-T,T]}(\cdot)$ operation constrains values within the range of $-T$ to T. M represents result after Bin operation. i, j, and t represent position index.

Subsequently, the obtained Bin undergo two Conv layers to extract the forgery distribution artifact features:

$$F^{'} = \text{ReLU}\left(\text{BN}\left(\text{Conv}\left(\text{ReLU}\left(\text{BN}\left(\text{Conv}_d\left(M\right)\right)\right)\right)\right)\right), \tag{5}$$

where Conv_d represents dilated convolution.

To simulate the compression process, we perform a Repetition (Rep) the Quantization Table (QT) and multiply it with $F^{'}$, followed by Concat with $F^{'}$:

$$F_a = \text{Concat}\left(F^{'}, \text{Rep}\left(\text{QT}\right) F^{'}\right), \tag{6}$$

where F_a represents extracted artifact feature, which will be utilized in the subsequent feature fusion process.

2.4. Forgery Feature Fusion

As aforementioned, we have obtained edge artifact features and compression artifact features. The coarse-grained compression artifact features offer an approximate localization of the forged region, while the fine-grained edge artifact features exhibit a superior localization effect along the edges of the forged region. By integrating these two feature types, we aim to achieve a more precise localization of the forged area.

To incorporate these extracted forgery feature information into SAM [10], we will fuse these features and subsequently feed them into each layer of image encoder of SAM. Specifically, we Concat the acquired edge artifact features and compression artifact features to form the total feature. We then employ Multilayer Perceptron (MLP) to extract the splicing features F_s:

$$F_s = \text{MLP}\left(\text{Concat}\left(F_e, F_a\right)\right). \tag{7}$$

Subsequently, F_s is input into each layer of image encoder, guiding the method to focus on the forged image regions.

2.5. Training and Inference

To maintain the generalization and object understanding abilities of SAM, inspired from the success of [10], we keep image encoder parameters of SAM frozen throughout training process, fining-tune only the remaining parameters.

To enhance the learning of compression artifact features, in alignment with [13], we initialize the weights of CAE through pre-training on double JPEG compression detection [9].

During inference process, we initially extract the Quantization Table and DCT coefficients from JPEG images. In the case of non-JPEG images, we directly compress them into JPEG format using 100 quality factor and no chroma subsampling.

3. EXPERIMENTAL RESULTS

3.1. Datasets and Implementation Details

To validate the efficacy of our method, we conduct experiments across various datasets.

IMD2020 dataset [14] encompasses 2,010 real forged images sourced from the internet.

SPLICED COCO dataset [9] consists of 917,648 forged images, enriched with diverse quantization tables.

NC16 SPLICING dataset [15], a part of National Institute of Standards and Technology (NIST) 2016, includes 288 forged images.

CARVALHO dataset [16] is used for image forgery localization and detection, containing 100 images, which are mainly tampered through splicing.

COLUMBIA dataset [17] comprises 180 forged images, where the mask is derived from specific post-processing operations.

Our method undergoes training and testing on IMD2020 and SPLICED COCO. To assess the method's generalizability, we further test it on unseen NC16 SPLICING and COLUMBIA datasets.

According to [10], we utilize Adaptive momentum Weight (AdamW) optimizer [18] decay optimizer, with a batch size set to 2. The initial learning rate is set to 2e-4, optimized through Cosine decay. The experiments are implemented by PyTorch on 2 Nvidia Tesla V100 GPUs.

3.2. Evaluation Metrics

We employ F1 as the fundamental metric to evaluate our method, which focuses on the positive class, is well-suited for evaluating the predictive performance in mask prediction tasks, particularly when dealing with imbalanced datasets:

$$\mathrm{F1}\left(G,P\right)=\frac{2\mathrm{TP}}{2\mathrm{TP}+\mathrm{FP}+\mathrm{FN}}, \tag{8}$$

where G represents the ground truth mask, P is the predicted mask, TP denotes true positives, FP represents false positives, and FN stands for false negatives.

For the task of image forgery localization, the ability to distinguish between original and tampered regions is crucial. Therefore, we adopt permutation metrics [5] for evaluation. Specifically, we calculate $\mathrm{p}-\mathrm{F1}$, which is defined as:

$$\mathrm{p}-\mathrm{F1}\left(G,P\right)=\max\left(\mathrm{F1}\left(G,P\right),\mathrm{F1}\left(G,\overline{P}\right)\right), \tag{9}$$

where $\overline{P}$ represents the inverted predicted mask.

Additionally, we evaluate the mask prediction performance using the average precision (AP), which measures the average precision across various thresholds by computing the area under the precision-recall curve. Similarly, we adopt permutation metrics and calculate $\mathrm{p}-\mathrm{AP}$ as:

$$\mathrm{p}-\mathrm{AP}\left(G,P\right)=\max\left(\mathrm{AP}\left(G,P\right),\mathrm{AP}\left(G,\overline{P}\right)\right). \tag{10}$$

The mean intersection over union (mIoU) is a widely adopted metric for evaluating the performance of image forgery localization models. In the context of binary classification, it is calculated as follows:

$$\mathrm{mIoU}\left(G,P\right)=\frac{1}{2}\left(\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}+\mathrm{FN}}+\frac{\mathrm{TP}}{\mathrm{TN}+\mathrm{FP}+\mathrm{FN}}\right), \tag{11}$$

where TN stands for true negatives.

To further evaluate the performance, we also employ permutation metrics [5]. Specifically, we calculate $\mathrm{p}-\mathrm{mIoU}$, which is defined as:

$$\mathrm{p}-\mathrm{mIoU}\left(G,P\right)=\max\left(\mathrm{mIoU}\left(G,P\right),\mathrm{mIoU}\left(G,\overline{P}\right)\right). \tag{12}$$

3.3. Comparison with the State-of-the-Arts

As demonstrated in Table 1, our method achieves the highest metrics on NC16 SPLICING, CARVALHO, and COLUMBIA, affirming its exceptional generalizability and proficiency in locating previously unseen images. Given that image forgery detection necessitates determining whether each pixel is forged, it is imperative to predict a hard mask. Specifically, regarding p-F1, our method outperforms the best-performing method CAT-Net [9] on all three datasets, reflecting a better balance between precision and recall. It also excels in terms of p-AP, providing a more comprehensive evaluation of our method's superiority. Since our method effectively extracts compression artifacts and edge artifacts, while inheriting the strong object concept understanding of larger models, it can accurately detect the forged areas, even for datasets that were not encountered during training. This demonstrates the robust generalization capabilities of our method.

As depicted in Table 2, our method outperforms the state-of-the-arts on both IMD2020 and SPLICED COCO, emphasizing its superior abilities in splicing localization. Because the Compression Artifact Extractor (CAE) and Edge Artifact Extractor (EAE) have successfully extracted the compression artifacts and edge artifacts in the spliced images, our method can more easily locate the splicing region. Furthermore, our method achieves the highest metrics on both NC16 SPLICING and COLUMBIA, affirming its remarkable generalizability and proficiency in locating previously unseen images.

Given the crucial role of the predicted forgery region mask in our research, we have focused on ensuring its accuracy for effective image repair. As the mask pinpoints the suspected manipulated areas, it directly impacts the subsequent repair process. We utilize LaMa [11] for image repair, which is highly sensitive to the edge pixels of the image being repaired.

Table 1: Comparison of p-F1 and p-AP with the state-of-the-arts on NC16 SPLICING, CARVALHO and COLUMBIA. Best results are highlighted in bold. † indicates reproduced results.

Method	Venue	NC16 Splicing		Carvalho		Columbia	
		p-F1	p-AP	p-F1	p-AP	p-F1	p-AP
DBA [19]	ICME '07	12.60	21.13	24.48	31.87	40.87	41.48
NOI1 [20]	IVC '09	17.66	25.51	36.27	37.23	48.13	54.77
ADQ [21]	PR '09	17.01	14.74	40.84	37.89	41.22	37.71
NADQ [22]	TIFS '12	12.69	7.91	24.54	14.87	48.14	36.21
CFA [23]	TIFS '12	16.59	18.60	27.87	25.88	72.54	75.02
NOI2 [24]	IJCV '14	14.26	13.18	25.84	23.74	43.28	46.70
CAGI [25]	JVCIR '18	14.45	24.81	34.87	50.05	48.28	56.99
EXIF-SC [5]	ECCV '18	40.72	51.60	43.98	53.01	78.05	94.50
ManTra-Net [6]	CVPR '19	27.85	33.38	41.68	52.86	50.97	64.66
Noiseprint [26]	TIFS '19	21.51	39.89	42.12	76.79	50.42	80.85
CAT-Net [9] (Baseline) †	IJCV '22	55.62	68.76	78.79	86.41	93.97	95.87
SAM-ICE (Ours)		**57.31**	**70.04**	**79.81**	**87.29**	**94.76**	**95.92**

Table 2: Comparison of mIoU and p-mIoU with the state-of-the-arts and our variants on IMD2020, SPLICED COCO, NC16 SPLICING, and COLUMBIA. Best results are highlighted in bold. † indicates reproduced results.

Method	Venue	IMD2020		Spliced COCO		NC16 Splicing		Columbia	
		mIoU	p-mIoU	mIoU	p-mIoU	mIoU	p-mIoU	mIoU	p-mIoU
EXIF-SC [5]	ECCV '18	-	-	-	-	48.68	53.55	80.81	85.29
ManTra-Net [6]	CVPR '19	-	-	-	-	50.12	50.34	52.34	52.40
CAT-Net [9] (Baseline) †	IJCV '22	76.00	76.53	93.87	93.87	68.41	69.18	83.05	90.99
SAM-ICE *w/o* CAE		76.35	76.69	94.04	94.04	72.24	72.65	83.27	90.80
SAM-ICE *w/o* EAE		75.94	76.47	93.80	93.80	69.37	70.52	82.27	87.99
SAM-ICE (Ours)		**76.51**	**76.96**	**94.17**	**94.17**	**73.05**	**73.60**	**85.13**	**91.81**

To overcome the challenges posed by LaMa's sensitivity, we implement a strategy of expanding the area around the forged markings in the forgery mask during the repair process. This method has proven to be effective, where we compare the results of using the mask generated by our method and the mask generated by CAT-Net [9]. Our method's superior performance in these datasets can be attributed to the accurate prediction of the forgery mask during the mask prediction phase. This accuracy enables a more precise localization of the manipulated area, leading to better repair performance.

It is important to note that the forgery mask only serves to distinguish between manipulated and authentic areas, and does not determine the specific type of manipulation. Therefore, in the repair process, we also employ two types of masks derived from the permutation metric and select the best result among them. Furthermore, our method has enabled us to frame the estimation of noise statistics as an optimization problem with a closed-form solution. We have developed this into an effective method for estimating local noise statistics, which further enhances the overall effectiveness of our image repair method. This advance not only improves the quality of the repaired images but also contributes to the broader field of image manipulation detection and restoration.

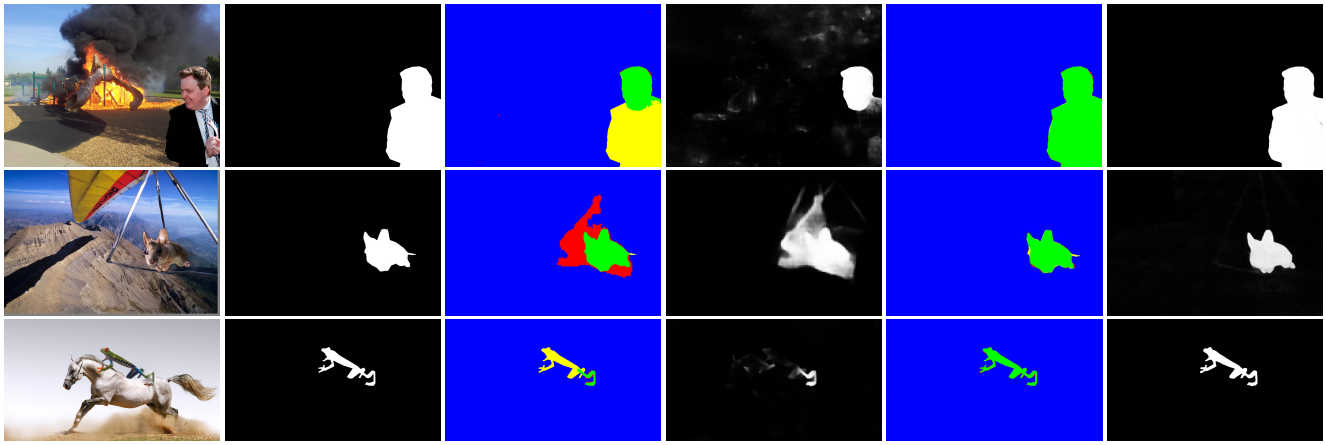
3.4. Ablation Studies

Observing Table 2, the integration of only EAE-extracted features into SAM [10], without guidance from compression artifact features, results in a decrease in both mIoU and p-mIoU compared to our method, and vice versa. Even when exclusively incorporating EAE-extracted features, we consistently outperform CAT-Net [9], showcasing the robust splicing edge localization capability of EAE.

The experiments conducted on the unseen dataset validate the efficacy and generalizability of our method. This affirms that EAE proficiently extracts edge splicing forgery features, and CAE adeptly captures compression artifact features. The fusion of these two features significantly enhances splicing feature extraction.

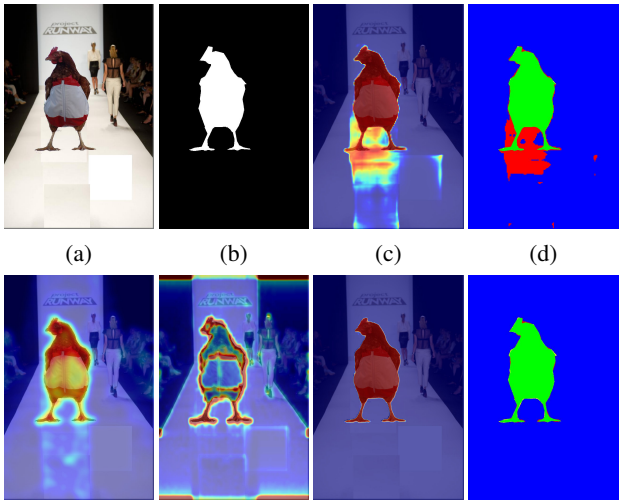
3.5. Results Visualization

Fig. 3 displays selected image splicing localization results, our results primarily consist of green and blue sections, indicating greater accuracy, while CAT-Net [9] exhibits noticeable



(a) Input (b) Ground-truth (c) CAT-Net (d) CAT-Net (e) Ours (f) Ours

Fig. 3: Comparison of image forgery localization results. (c) and (e) Green, blue, red, and yellow denote true positives, true negatives, false positives, and false negatives, respectively.



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

(e) (f) (g) (h)

Fig. 4: Visualization of attention map extraction by different modules. (a) Input image, (b) Ground-truth mask, (c) Attention map extracted by CAT-Net, (d) Comparison of CAT-Net predicted mask and ground-truth, (e) Attention map extracted by CAE, (f) EAE, (g) and SAM-ICE, and (h) Comparison of SAM-ICE predicted mask and ground-truth.

red and yellow sections, signifying more incorrect predictions. This illustrates that our method, benefiting from the extraction and fusion of forged edge artifact features and compressed artifact features, is less affected by similar non-forged pixels, resulting in enhanced forgery localization accuracy. Moreover, our predicted mask is predominantly black in the real region, whereas CAT-Net's mask shows gray and white, emphasizing the robust confidence of our method. This observation underscores the superior discriminability of our method.

Illustrated in Fig. 4, we visually inspect the attention of the method to compare the region of interest during localization. CAE and EAE modules focus on the spliced region and the spliced edge of the image, respectively. After the fusion module, the model accurately directs its attention to the spliced region of the image. Conversely, due to the lack of object understanding ability and relatively single-feature extraction, CAT-Net [9] produces some erroneous predictions. This underscores the effectiveness of the proposed modules.

4. CONCLUSIONS

This paper introduces a splicing forgery localization method that Integrates Compression and Edge artifact features via SAM (SAM-ICE). Edge Artifact Extractor (EAE) is employed to extract forgery features introduced at the edges during image splicing, while Compression Artifact Extractor (CAE) is utilized to extract compression artifact features. The fusion of these features is facilitated by the object understanding ability of SAM, guiding the method in accurately locating splicing positions within the image. Experiments substantiates the effectiveness, robustness, and generalizability of our method.

However, our method relies on the assumption that high-frequency artifacts are present at the edges of the forged regions. This assumption may limit the model's effectiveness in scenarios where the frequency content at the edges of the forged regions is low. In such cases, our method may struggle to extract the forgery artifacts from the edges, potentially leading to suboptimal prediction results. Therefore, as part of our future research, we aim to explore methods for extracting artifacts that are not only dependent on high frequencies, to further improve the performance and robustness of our splicing forgery localization method.

References

[1] Liang Liao et al. “Guidance and Evaluation: Semantic-Aware Image Inpainting for Mixed Scenes”. In: *Proc. Eur. Conf. Comput. Vis.* 2020, pp. 683–700.

[2] Kui Jiang et al. “Rain-Free and Residue Hand-in-Hand: A Progressive Coupled Network for Real-Time Image Deraining”. In: *IEEE Trans. Image Process.* 30 (2021), pp. 7404–7418.

[3] Xian Zhong et al. “Grayscale Enhancement Colorization Network for Visible-Infrared Person Re-Identification”. In: *IEEE Trans. Circuits Syst. Video Technol.* 32.3 (2022), pp. 1418–1430.

[4] Chunwei Tian et al. “A cross Transformer for image denoising”. In: *Inf. Fusion* 102 (2024), p. 102043.

[5] Minyoung Huh et al. “Fighting Fake News: Image Splice Detection via Learned Self-Consistency”. In: *Proc. Eur. Conf. Comput. Vis*. 2018, pp. 106–124.

[6] Yue Wu, Wael AbdAlmageed, and Premkumar Natarajan. “ManTra-Net: Manipulation Tracing Network for Detection and Localization of Image Forgeries With Anomalous Features”. In: *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.* 2019, pp. 9543–9552.

[7] Susmit Agrawal et al. “SISL: Self-Supervised Image Signature Learning for Splicing Detection & Localization”. In: *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops*. IEEE, 2022, pp. 22–32.

[8] Nitish Kumar and Toshanlal Meenpal. “Salient keypoint-based copy–move image forgery detection”. In: *Aust. J. Forensic Sci.* 55.3 (2023), pp. 331–354.

[9] Myung-Joon Kwon et al. “Learning JPEG Compression Artifacts for Image Manipulation Detection and Localization”. In: *Int. J. Comput. Vis.* 130.8 (2022), pp. 1875–1895.

[10] Alexander Kirillov et al. “Segment Anything”. In: *Proc. IEEE/CVF Int. Conf. Comput. Vis.* 2023, pp. 3992–4003.

[11] Roman Suvorov et al. “Resolution-robust Large Mask Inpainting with Fourier Convolutions”. In: *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis.* 2022, pp. 3172–3182.

[12] Kai Xu et al. “Learning in the Frequency Domain”. In: *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.* 2020, pp. 1737–1746.

[13] Mauro Barni et al. “Aligned and non-aligned double JPEG detection using convolutional neural networks”. In: *J. Vis. Commun. Image Represent.* 49 (2017), pp. 153–163.

[14] Adam Novozámský, Babak Mahdian, and Stanislav Saic. “IMD2020: A Large-Scale Annotated Dataset Tailored for Detecting Manipulated Images”. In: *Proc. IEEE/CVF Winter Appl. Comput. Vis. Workshops*. 2020, pp. 71–80.

[15] Haiying Guan et al. “MFC Datasets: Large-Scale Benchmark Datasets for Media Forensic Challenge Evaluation”. In: *Proc. IEEE/CVF Winter Appl. Comput. Vis. Workshops*. 2019, pp. 63–72.

[16] Tiago Jose de Carvalho et al. “Exposing Digital Image Forgeries by Illumination Color Classification”. In: *IEEE Trans. Inf. Forensics Secur.* 8.7 (2013), pp. 1182–1194.

[17] Tian-Tsong Ng, Shih-Fu Chang, and Q Sun. “A data set of authentic and spliced image blocks”. In: *Columbia Univ. ADVENT Tech. Rep.* 4 (2004).

[18] Ilya Loshchilov and Frank Hutter. “Decoupled Weight Decay Regularization”. In: *Proc. Int. Conf. Learn. Represent.* 2019.

[19] Shuiming Ye, Qibin Sun, and Ee-Chien Chang. “Detecting Digital Image Forgeries by Measuring Inconsistencies of Blocking Artifact”. In: *Proc. IEEE Int. Conf. Multimedia and Expo*. 2007, pp. 12–15.

[20] Babak Mahdian and Stanislav Saic. “Using noise inconsistencies for blind image forensics”. In: *Image Vis. Comput.* 27.10 (2009), pp. 1497–1503.

[21] Zhouchen Lin et al. “Fast, automatic and fine-grained tampered JPEG image detection via DCT coefficient analysis”. In: *Pattern Recognit.* 42.11 (2009), pp. 2492–2501.

[22] Tiziano Bianchi and Alessandro Piva. “Image Forgery Localization via Block-Grained Analysis of JPEG Artifacts”. In: *IEEE Trans. Inf. Forensics Secur.* 7.3 (2012), pp. 1003–1017.

[23] Pasquale Ferrara et al. “Image Forgery Localization via Fine-Grained Analysis of CFA Artifacts”. In: *IEEE Trans. Inf. Forensics Secur.* 7.5 (2012), pp. 1566–1577.

[24] Siwei Lyu, Xunyu Pan, and Xing Zhang. “Exposing Region Splicing Forgeries with Blind Local Noise Estimation”. In: *Int. J. Comput. Vis.* 110.2 (2014), pp. 202–221.

[25] Chryssanthi Iakovidou et al. “Content-aware detection of JPEG grid inconsistencies for intuitive image forensics”. In: *J. Vis. Commun. Image Represent.* 54 (2018), pp. 155–170.

[26] Davide Cozzolino and Luisa Verdoliva. “Noiseprint: A CNN-Based Camera Model Fingerprint”. In: *IEEE Trans. Inf. Forensics Secur.* 15 (2020), pp. 144–159.