applied sciences

MDPI

Article

GP-Net: Image Manipulation Detection and Localization via Long-Range Modeling and Transformers

Jin Peng [1,†], Chengming Liu [1,†], Haibo Pang [1,*,†], Xiaomeng Gao [1], Guozhen Cheng [2] and Bing Hao [3]

[1] School of Cyber Science and Engineering, Zhengzhou University, No. 97, Wenhua Road, Zhengzhou 450002, China; jpeng@gs.zzu.edu.cn (J.P.); cmliu@zzu.edu.cn (C.L.); xmgao@stu.zzu.edu.cn (X.G.)
[2] Institute of Information Technology, Information Engineering University, Zhengzhou 450002, China; guozhencheng@hotmail.com
[3] Songshan Laboratory, Zhengzhou 450002, China; hbky2012@163.com
* Correspondence: phb@zzu.edu.cn
† These authors contributed equally to this work.

Abstract: With the rise of image manipulation techniques, an increasing number of individuals find it easy to manipulate image content. Undoubtedly, this presents a significant challenge to the integrity of multimedia data, thereby fueling the advancement of image forgery detection research. A majority of current methods employ convolutional neural networks (CNNs) for image manipulation localization, yielding promising outcomes. Nevertheless, CNN-based approaches possess limitations in establishing explicit long-range relationships. Consequently, addressing the image manipulation localization task necessitates a solution that adeptly builds global context while preserving a robust grasp of low-level details. In this paper, we propose GPNet to address this challenge. GPNet combines Transformer and CNN in parallel which can build global dependency and capture low-level details efficiently. Additionally, we devise an effective fusion module referred to as TcFusion, which proficiently amalgamates feature maps generated by both branches. Thorough extensive experiments conducted on diverse datasets showcase that our network outperforms prevailing state-of-the-art manipulation detection and localization approaches.

Keywords: image manipulation localization; long-range moeling; two-stream network; feature fusion

check for updates

Citation: Peng, J.; Liu, C.; Pang, H.; Gao, X.; Cheng, G.; Hao, B. GP-Net: Image Manipulation Detection and Localization via Long-Range Modeling and Transformers. *Appl. Sci.* **2023**, *13*, 12053. https://doi.org/10.3390/app132112053

Academic Editor: Sungho Kim

Received: 13 October 2023
Revised: 30 October 2023
Accepted: 3 November 2023
Published: 5 November 2023

Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

In recent years, due to rapid advancements in deep generative models [1–3], a multitude of image editing applications [4,5] have gained widespread popularity among the general public. However, there is growing concern regarding the potential misuse of these editing techniques for image forgery. The act of image forgery carries significant implications across various domains, including instances of academic misconduct and the dissemination of counterfeit images. Such occurrences emphasize the urgent need to intensify our focus on detecting image forgery. Consequently, the development of effective methodologies for accurately identifying tampered regions in images has become of utmost importance.

Generally, image manipulation techniques are categorized into three groups. For each type of manipultion, researchers have proposed numerous approaches for localizing image manipulations, such as splicing [6–10], copy-and-paste [11–13], and removal [14,15]. As depicted in Figure 1, these methods often operate at the object level to create semantically coherent and persuasive images, involving the addition or removal of objects within the image. Drawing inspiration from this, we contend that image manipulation detection and localization should consider consistency of objects with one another, necessitating the establishment of long-range relations. While certain CNN-based methods [16–20] have achieved some success in this regard, building long-range relations using these methods requires continuous downsampling and convolution operations. This approach

presents several drawbacks: (1) excessively deep networks can encounter the problem of gradient disappearance, and (2) the gradual reduction in spatial resolution results in the disappearance of crucial local information essential for image manipulation detection and localization tasks.

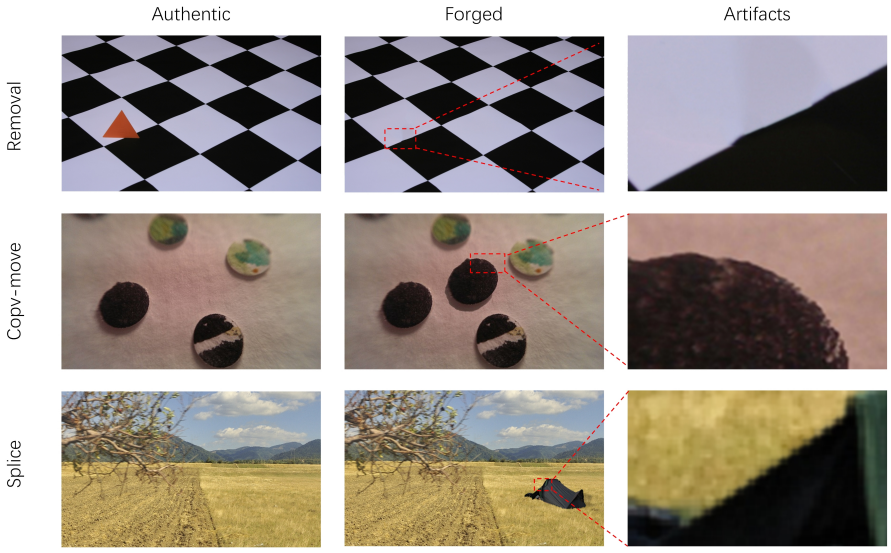

Figure 1. Image tampering detection and localization often requires capturing traces left on objects. Hence, it is crucial to leverage object-level consistency for effective forgery detection and localization.

The Transformer architecture [21], originally employed in the realm of Natural Language Processing (NLP), has garnered significant attention within the computer vision community. One notable application is the SETR model, which replaces the encoder layers of traditional encoder–decoder networks with transformers. This modification has led to impressive outcomes in image segmentation tasks, establishing SETR as a state-of-the-art solution. While the Transformer excels at establishing long-range relationships, it does exhibit limitations in capturing intricate details. In the context of image forgery detection, the ability to capture fine details is crucial. Consequently, a pure Transformer-based approach akin to SETR does not yield satisfactory results in this domain.

In recent research efforts, there have been attempts to combine CNNs and Transformers to leverage their respective strengths [22,23]. For example, ObjectFormer [22] utilizes an RGB stream and a frequency stream, employing CNNs to extract features that are then processed by Transformers to establish long-range relationships and generate prediction masks. However, this approach mainly focuses on replacing convolutional layers with Transformers without effectively leveraging the advantages of both architectures. To fully harness the strength of CNNs combined with Transformers in manipulation detection, we propose a novel two-stream network called GP-Net. In GP-Net, the CNN-based branch and the Transformer-based branch operate in parallel, allowing each branch specialization in its respective strengths. To combine feature maps from both branches in an effective way, the TcFusion module is introduced by us, which efficiently fuses the feature maps to make predictions. Through an extensive ablation study, we optimize our network structure, resulting in improved performance compared to SOAT approaches across various datasets. In addition, we summarize the main contributions of our work:

- GPNet represents a two-branch network which integrates CNNs and Transformers for image manipulation detection. It overcomes the limitations of excessively deep networks, addressing issues such as gradient vanishing and feature diminishment.
- We introduce the TcFusion module, a novel feature fusion mechanism that effectively combines features from the CNN branch and the Transformer branch. This fusion module enables the integration of complementary information from both branches.

- Through extensive experiments conducted on multiple baselines, we prove that our approach achieves SOAT performance in both detection and localization of image manipulations.

2. Related Work

2.1. Image Manipulation Detection

Image manipulation detection involves the task of differentiating tampered images from original images through image-level binary classification. Previous studies [11,24], approached the task by obtaining detection scores based on statistics, such as the mean [24] or the maximum [11]. These methods focus primarily on image-level detection without explicitly considering pixel-level manipulation localization. However, recent works have increasingly emphasized manipulation localization on a pixel level while overlooking the importance of imge detection. In contrast, our study takes a holistic approach by jointly considering both the manipulation detection and localization. We recognize the significance of accurately identifying manipulated regions within an image while also providing a binary classification of the overall image. By integrating detection and localization, our approach aims to provide a comprehensive solution for image manipulation analysis.

2.2. Image Manipulation Localization

Early related works primarily focused on localizing specific types of operations. For example, some studies [6–10] concentrated on detecting splicing, while others [11–13] focused on copy-move manipulation, and [14,15] dealt with removal operations. But in reality, the type of manipulation is not always known, which has motivated recent research interest in general manipulation detection. Consequently, recent works have attempted to develop models capable of handling multiple types of forgeries within a single framework. RGB-N [16], for instance, adopts a two-stream Faster R-CNN-based network that includes an RGB stream and a noise stream generated by the SRM filter [25]. However, when images are taken by the same type of camera, their noise patterns tend to be consistent, leading to the failure of the noise branch in distinguishing manipulations. Another approach, SPAN [19], introduces a spatial pyramidal attention network which enables efficient as well as explicit comparison of patches at multiple scales through local self-attention blocks. While SPAN leverages local area correlations, it does not fully exploit spatial correlation and thus has limited generality. PSCC-Net [20] takes a stepwise method to enhance multiscale feature representation and incorporates the Spatial and Channel Correlation Module (SCCM) to better explore the correlation between space and channel. However, the structure of PSCC-Net becomes complex in order to capture more spatial information. ObjectFormer [22] introduces the Transformer to image tampering detection, but it simply concatenates CNN and Transformer in a sequential manner without effectively leveraging the strengths of both approaches. In this paper, we leverage the modeling capability of Transformers to improve the identification of tampered areas, aiming to benefit from its advantages in capturing global relationships and context.

3. Method

GPNet is specifically designed to address the challenge of effectively localizing various types of image manipulations. Unlike image-level detection, which focuses on identifying the presence of tampering, pixel-level localization presents a greater difficulty. Consequently, GPNet places a stronger emphasis on solving the localization problem. We explain the architecture of GPNet in greater detail below. The tampered image is fed into GP-Net, and the feature maps are obtained by the CNN branch and the Transformer branch in parallel. Then, feature maps from both branches are fused by the TcFusion module we propose. The final tampered region is obtained through a process of upsampling. The architecture of GPNet is depicted in Figure 2. The green area represents the CNN stream, as discussed in detail in Section 3.1. The CNN stream is responsible for the extraction of relevant feature maps from the input images. The orange area represents the Transformer

stream. This stream is explained comprehensively in Section 3.2. The Transformer stream focuses on capturing long-range dependencies and refining the extracted features. The feature maps obtained from both the CNN stream and the Transformer stream, with the same resolution, are then fed into the TcFusion module, depicted in the pink area of Figure 2. The TcFusion module combines the features from both branches to make the final prediction. The specifics of this module are introduced in Section 3.3. By integrating both the CNN and Transformer branches and utilizing the TcFusion module, GPNet aims to leverage the strengths of each approach to enhance the overall performance in pixel-level localization of image manipulations.

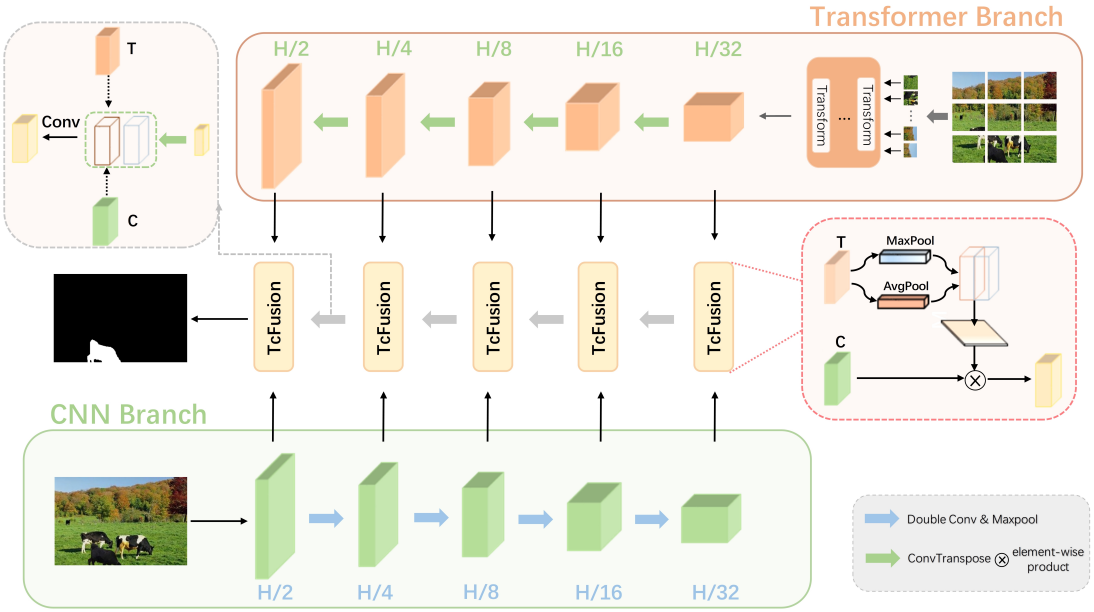

Figure 2. Overview of the GP-Net architecture for manipulating detection and localization tasks.

3.1. CNN Branch

In the CNN branch of GPNet, the input image undergoes a series of convolutional and downsampling operations to generate the feature map. Traditionally, the feature map is progressively downsampled to capture a global receptive field, which often requires a very deep network. However, in GPNet, we aim to leverage the advantages provided by Transformers while addressing the issue of gradient disappearance. Therefore, we choose not to use an excessively deep network. To tackle the gradient disappearance problem further, we introduce two algorithms: residual propagation [26] and residual feedback [7]. Algorithms are as Figure 3 shows. Residual propagation operates similarly to the recall mechanisms of our brain, mitigating the degradation problem in deeper networks by recalling input feature information. The definition of residual propagation is presented in Equation (1). This equation describes the way in which the residual propagation algorithm modifies the input features. However, without the specific details of Equation (1), we are unable to provide further explanation.

$$y_f = F(x, W_i) + M(x). \tag{1}$$

In the component block, x represents the input, y_f represents the output of the block and W_i represents the parameters of layer i. $F(x, W_i)$ is a learnable function that denotes the feature maps obtained after applying two convolutional layers and ReLU activation. Following that, a learnable linear mapping layer M is employed to adjust the dimension of input x, ensuring it matches the dimension of (x, W_i) so that they can be added together. The residual feedback mechanism aims to integrate the input feature information and amplify

the differences in essential image properties between the untampered and tampered regions. Equation (2) presents the definition of residual feedback.

$$y_b = (s(W(y_f)) + 1) \times x, \tag{2}$$

where x is input of the block and y_b represents the enhanced output of block. In addtion, y_f is the output of the previous step of residual Equation (1). W is a linear mapping that changes the dimension of y_f, and s denotes the sigmoid activation function.

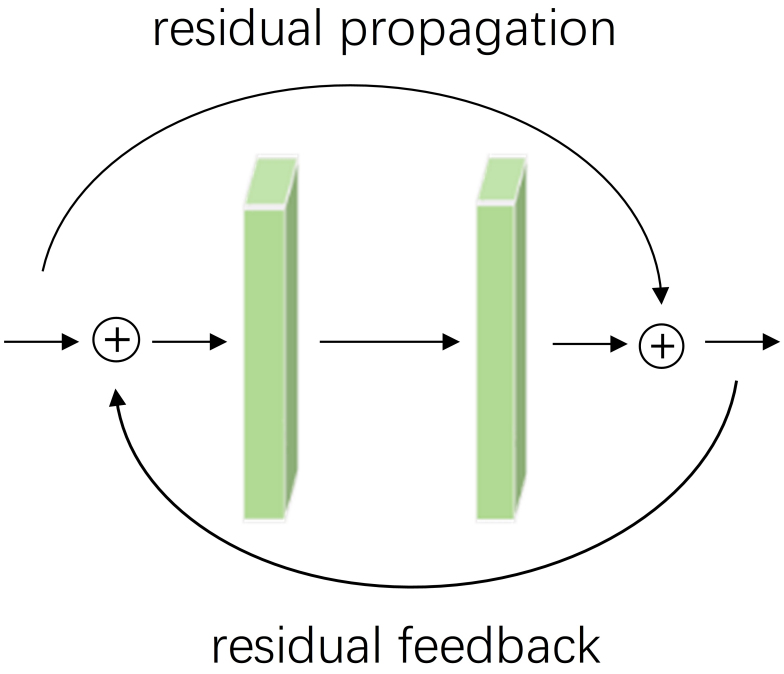

Figure 3. The structure of the residual propagation and residual feedback.

3.2. Transformer Branch

In the Transformer branch of our network, we adopt a typical encoder–decoder structure. In this branch, the input image $x \in \mathbb{R}^{H \times W \times 3}$ is initially divided into $N = \frac{H}{M} \times \frac{W}{M}$ patches (M often set to 16). These patches are then flattened and fed into a linear layer with an output dimension of D, resulting in an embedding sequence $S \in \mathbb{R}^{N \times D}$. To enable long-range modeling, a learnable position embedding of the same dimension as S is added to the embedding sequence.This position embedding helps the model capture spatial relationships between the patches. Subsequently, the embedding sequence $S^1 \in \mathbb{R}^{(N+1) \times D}$ is input into the Transformer encoder, which consists of n layers of Multi-Head Self-Attention (MSA) and Multilayer Perceptron (MLP), which includes n Layers of Multi-Head Self-Attention (MSA) and Multilayer Perceptron (MLP). The Self-Attention (SA) mechanism is the core principle of Transformer, which updates the state of each embedded patch by aggregating information in each layer. Its definition is presented in Equation (3).

$$SA(\mathbf{s}_i) = softmax(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{D_k}})\mathbf{V}, \tag{3}$$

where $[\mathbf{Q}, \mathbf{K}, \mathbf{V}] = \mathbf{s}\mathbf{W}_{QKV}, \mathbf{W}_{QKV} \in \mathbb{R}^{D \times 3D_k}$ represents the projection matrix. MSA is a combination of multiple self-attentive mechanisms, and MLP is a series of layers (for details on MSA and MLP, please refer to [27]). In the decoder layers, we employ a progressive upsampling approach. As depicted in Figure 2, we take the feature map obtained from the encoder layers and apply a sequence of upsampling operations to generate feature maps of different sizes. Subsequently, these upscaled feature maps are fused with feature maps of the same size from the CNN branch. The fusion process combines the complementary

information captured by both branches, leveraging the strengths of each to enhance the localization of image manipulations.

3.3. TcFusion Module

To achieve long-range modeling relationships within feature maps, we introduce a feature fusion module known as the TcFusion module. As shown in Figure 2, the feature map from Transformer branch is calculated by the Spatial Attention Module (SAM) to obtain a spatial attention map $\mathbf{M}(\mathbf{T}) \in \mathbb{R}^{H \times W}$ and then multiplied with the feature map from the CNN branch. Next, we describe the operation in detail. We use two pooling operations to summarize the information of the feature maps and generate two 2D feature maps: $\mathbf{T_{avg}} \in \mathbb{R}^{1 \times H \times W}$ and $\mathbf{T_{max}} \in \mathbb{R}^{1 \times H \times W}$. They represent the average-pooling feature and the max-pooling feature across the channel. These features are concatenated and passed through a convolutional operation with a 3×3 kernel size to generate a 2D spatial attention feature map. In a word, the definition of SAM is presented in Equation (4).

$$\mathbf{M}(\mathbf{T}) = \sigma(f^{3\times3}([\mathbf{T_{avg}}; \mathbf{T_{max}}])), \tag{4}$$

where σ is the sigmoid function, and f represents the convolution operation with a kernel size of 3×3.

Indeed, traditional CNN-based approaches achieve a global receptive field by repeatedly downsampling the input. While this approach is effective, it often requires very deep networks. The TcFusion module addresses these challenges by enabling the modeling of global relationships without the need for an excessively deep network.

4. Experiments

4.1. Experimental Settings

Experimental Dataset: We assess the performance of our network using three widely recognized datasets: CASIA [28], Nist Nimble 2016 (NIST16) [29], and Coverage [30]. Table 1 provides an overview of the forgery types present in each dataset, along with the number of images utilized in the evaluation of our network.

Table 1. Summary of test datasets for our network (number denotes the number of images used for training and testing, ✓ and ✗ represent whether or not the manipulation type is involved).

Datast	Number		Type		
	Train	Test	Splicing	Copy	Removal
Coverage	80	20	✗	✓	✗
CASIA	5078	532	✓	✓	✗
NIST	457	107	✓	✓	✓

Experimental Metrics: In the experiment, selecting appropriate evaluation metrics is crucial for effectively assessing the model's performance. When it comes to image tampering detection, accurately identifying the manipulation regions at a pixel level is of utmost importance. In this study, we adopt evaluation metrics that have been commonly used in previous related works [22]. The F1 score combines recall and precision to provide a comprehensive measure of network performance. Meanwhile, the AUC score represents the area under the Receiver Operating Characteristic (ROC) curve. ROC curves are widely used to evaluate the classification effectiveness of a given classifier. In the context of manipulaiton localization, we classify each pixel in images as either tampered or un-tampered. Therefore, we utilize the AUC score as the experimental metric to assess model performance.

Compared Methods: To evaluate the performance of the proposed method and ensure the reliability of the experiment, we select a set of well-regarded manipulation detection

and localization approaches as comparative methods. These approaches include RGB-N (detailed in [16]), SPAN ([19]), RRU-Net ([7]), PSCCNet ([20]), and ObjectFormer ([22]).

Implementation Details: The implementation of our method is executed on a server equipped with an NVIDIA GeForce RTX 3090 GPU. We utilize PyTorch 1.8.2 to implement both RRU-Net, PSCC-Net, and GP-Net.

4.2. Compared Detection Methods

To evaluate the effectiveness of our network, we conduct a comparative analysis with some excellent methods: SPAN [21], RGB-N [18], RRU-Net [7], PSCCNet [19] and ObjectFormer [22]. To ensure fairness in the experiments, we fine-tune the parameters of the compared methods to their optimal values.

Table 2 presents quantitative results (%) obtained by each method. From the table, it is evident that the network we propose outperforms other methods on the CASIA dataset and the NIST dataset. However, on the Coverage dataset, the F1-score and AUC are slightly lower compared to ObjectFormer. We speculate that this discrepancy may be due to the limited number of images available on the Coverage dataset. It does not allow the network optimal convergence. While our network does not require a large number of images, having fewer than 100 training images may hinder the network's convergence. We provide further evidence for this in Section 4.6. Overall, our network is superior to other related methods in terms of performance.

Moreover, we present visual forecasts of the different approaches in Figure 4. Unfortunately, we are unable to provide predictions for ObjectFormer [22] as the code for that method was not made available to us. Each column from left to right represents a distinct aspect, where the initial column showcases the forged images, the second column illustrates the visualization outcomes of the RRU-Net method's predictions, the third column displays the prediction results of the PSCC-Net method, the fourth column represents the prediction results of our method, and the last column depicts the ground truth image. Based on the visualized results, it is apparent that the RRU-Net method exhibits commendable performance in identifying general tampered regions and providing rough localization, although it still exhibits some instances of false detections and missed areas. Moreover, the method struggles to perform well when confronted with small tampered regions. In comparison, the PSCC-Net method demonstrates superior performance with fewer instances of false and missed detections. However, even in its predicted visualization results, the method does not exhibit remarkable performance in identifying minute tampered areas. Subjectively speaking, our model's predicted visualization results demonstrate its superiority and stability amongst the various methods. Not only can it accurately locate manipulation areas, but it also generates clearer boundaries.

Table 2. The F1/AUC results (%) of our method and the other methods on various datasets.

Methods	CASIA	NIST16	Coverage
SPAN [21]	38.2/83.8	58.2/96.1	55.8/93.7
RGB-N [18]	40.8/79.5	72.2/93.7	43.7/81.7
RRU-Net [7]	45.2/79.8	79.8/92.3	61.3/80.5
PSCCNet [19]	55.4/87.5	81.9/99.6	72.3/94.1
ObjectFormer [22]	57.9/88.2	82.4/99.6	75.8/95.7
Ours	61.4/88.4	91.2/99.7	72.4/94.5

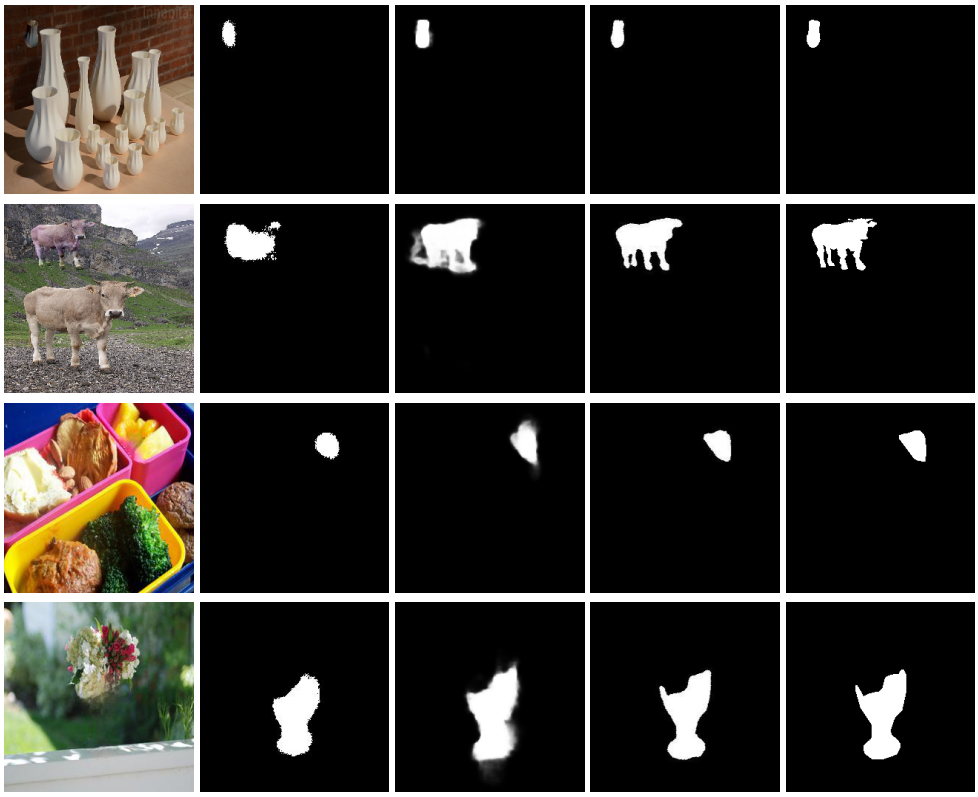

Figure 4. Visual comparisons of original models. From left to right, we present forged images, the predictions of RRUnet, PSCCNet and ours, GT masks.

4.3. Image Manipulation Detection

As recent works only consider pixel-level tampering detection and not image-level detection, there are no standard datasets for benchmarking.

Since the CASIA dataset and the RTD (Realistic Tampering Dataset) [31] contain both forgery images and its corresponding original image, we select 500 images from the CASIA dataset, 300 images from the NIST dataset and 220 images from the RTD with 50% forged and 50% original as a test dataset. From Table 3, it cn be seen that our model achieves good performance, especially on the nist dataset: our model achieves an F1 score of 84.8% and an AUC of 92.2%, which shows that our method is effective in capturing the manipulation artifacts.

Table 3. F1 scores and AUC (%) of manipulation detection results on various datas.

Dataset	Number	F1	AUC
CASIA [28]	500	78.1	88.7
NIST [29]	300	84.8	92.2
RTD [31]	220	69.2	71.3

4.4. Robustness Evaluation

In addition to the previous comparisons, we conduct further evaluations to assess the robustness of our method. To achieve this, we subject the images from the NIST dataset to various image attack methods. These methods consist of resizing, JPEG compression with the quality factor η and Gaussian blur with the kernel size κ. The parameters used and the performance of manipulation localization, measured by the F1-score and AUC, are presented in Table 4. Our model performs well under a variety of distortions. Especially on compressed images, the F1-score is only 0.4% lower than without the distortion when

the quality factor is 100 and 1.0% lower than without the distortion when the quality factor is 80. It is evident that our network demonstrates robustness against different image attack methods.

Table 4. Manipulationlocalization performance (AUC%) on NIST dataset under various distortions.

Distortion	F1	AUC
No distortion	91.2	99.7
Resize (0.78×)	90.8	96.5
Resize (0.25×)	87.7	92.4
GaussianBlur ($\kappa = 3$)	88.4	96.8
GaussianBlur ($\kappa = 7$)	85.1	90.1
JPEGCompress ($\eta = 100$)	90.8	98.4
JPEGCompress ($\eta = 80$)	90.2	98.1

4.5. Ablation Study

We employ the CNN branch and the Transformer branch, both branches running in parallel. Then, we fuse feature maps from both branches by the TcFusion module. To assess the usefulness of the branch-parallel approach to designed, we conduct experiments where each branch is removed separately from GPNet. Additionally, we replace the TcFusion module with a simple summation of feature maps to highlight the impact of the TcFusion module. We evaluate the manipulation localization performance on NIST and CASIA datasets.

The quantitative results are presented in Table 5. It is evident that when using only the CNN branch, both the F1 score and AUC decreased by 13.2% and 8.7%, respectively, on the NIST dataset and the CASIA dataset. Conversely, when employing only the Transformer branch, the AUC scores decreased by 5.3% on CASIA and 4.3% on NIST. Furthermore, without the TcFusion block, the F1-score and AUC experienced a decline of 6.8% and 7.1%, respectively. These deteriorations in the quantitative results demonstrate that the adoption of the parallel-in-branch design and the TcFusion module effectively contributes to the performance improvement of our method.

Table 5. Ablation results on CASIA and NIST datasets, AUC and F1 (%) are reported.

Variants	NIST16		CASIA	
	F1	AUC	F1	AUC
CNN Branch	83.7	96.0	49.8	80.3
Transformer Branch	74.3	86.5	49.3	76.7
w/o TcFusion	86.8	94.3	57.5	83.3
Ours	91.2	99.7	61.4	88.4

4.6. Limitation

GPNet indeed encounters a challenge when dealing with small datasets. To investigate this issue, we conduct a series of experiments on the CASIA and NIST16 datasets, and the results are displayed in Figure 5. We observe an overall upward trend in forgery localization performance as the number of training images increases. The AUC metric shows significant improvement, reaching its peak when training images reach 300. These findings indicate that the performance of GPNet benefits from a larger number of training samples.

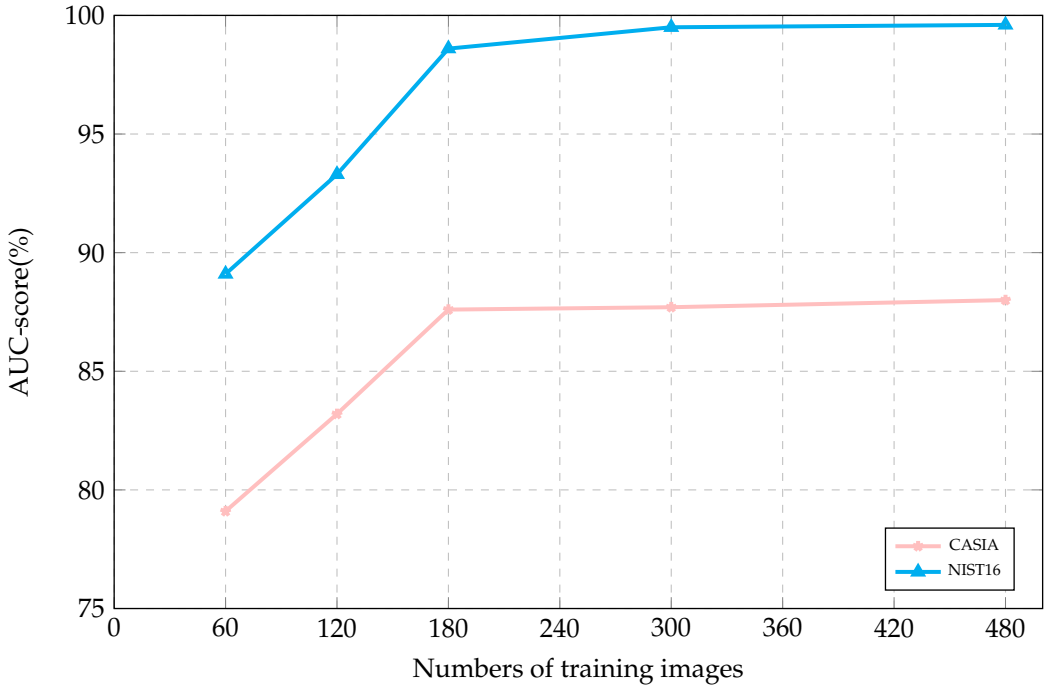

Figure 5. AUC score (%) of our method with different numbers of training images.

5. Conclusions

GPNet is introduced as an innovative method for detecting and localizing image manipulation. It combines a CNN branch for feature extraction and a Transformer branch for modeling global relationships. By leveraging these two branches, GPNet achieves precise manipulation localization without the need for a deep network. Extensive experiments on diverse datasets demonstrate that GPNet outperforms existing SOAT approaches. Our approach offers a fresh perspective on harnessing the synergies between CNN and Transformer architectures. Future work will focus on further enhancing the structure of the Transformer-based branch.

Author Contributions: Conceptualization, X.G. and J.P.; methodology, B.H. and J.P.; figures, X.G. and H.P.; formal analysis, J.P. and G.C.; data curation, C.L. and B.H.; original draft, B.H. and H.P.; review and editing X.G. and H.P.; the experiment J.P.; supervision, C.L. and G.C.; funding acquisition C.L. and H.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Nature Science Foundation of China (62006210, 62206252), the Key science and technology project of Henan Province (221100211200, 221100210100), the technological research projects in Henan province (232102210090).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The images used in this paper are all from famous popular image repositories which are publicly accessible. Intermediate data or results generated during and/or analysed during the current study are available from the corresponding author on reasonable request.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Razavi, A.; Oord, A.; Vinyals, O. Generating diverse high-fidelity images with vq-vae-2. In Proceedings of the Neural Information Processing Systems, Vancouver, BC, Canada, 8–14 December 2019; Volume 32.
2. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M. Generative adversarial nets. In Proceedings of the Neural Information Processing Systems, Montreal, QC, Canada, 8–13 December 2014; Volume 27.

3. Park, T.; Zhu, J.-Y.; Wang, O. Swapping autoencoder for deep image manipulation. In Proceedings of the Neural Information Processing Systems, Online, 6–12 December 2020.
4. Dhamo, H.; Farshad, A.; Laina, I. Semantic image manipulation using scene graphs. In Proceedings of the Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020.
5. Li, B.; Qi, X.; Lukasiewicz, T. Manigan: Text-guided image manipulation. In Proceedings of the Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020.
6. Liu, B.; Pun, C.-M. Deep fusion network for splicing forgery localization. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 237–251.
7. Bi, X.; Wei, Y.; Xiao, B.; Li, W. The Ringed Residual U-Net for Image Splicing Forgery Detection. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, Long Beach, CA, USA, 16–17 June 2019.
8. Bi, X.; Zhang, Z.; Liu, Y. Multi-task wavelet corrected network for image splicing forgery detection and localization. In Proceedings of the 2021 IEEE International Conference on Multimedia and Expo (ICME), Shenzhen, China, 5–9 July 2021; pp. 1–6.
9. Kwon, M.J.; Yu, I.J.; Nam, S.H.K. CAT-Net: Compression artifact tracing network for detection and localization of image splicing. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision 2021, Waikoloa, HI, USA, 3–8 January 2021; pp. 375–384.
10. Xiao, B.; Wei, Y.; Bi, X.; Li, W. Image splicing forgery detection combining coarse to refined convolutional neural network and adaptive clustering. *Inf. Sci.* **2020**, *511*, 172–191. [CrossRef]
11. Wu, Y.; Abd-Almageed, W.; Natarajan, P. Busternet: Detecting copy-move image forgery with source/target localization. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 168–184.
12. Zhu, Y.; Chen, C.; Yan, G.; Guo, Y. AR-Net: Adaptive attention and residual refinement network for copy-move forgery detection. *IEEE Trans. Ind. Inform.* **2020**, *16*, 6714–6723. [CrossRef]
13. Zhang, Y.; Zhu, G.; Wang, X. CNN-Transformer Based Generative Adversarial Network for Copy-Move Source/Target Distinguishment. *IEEE Trans. Circuits Syst. Video Technol.* **2022**, *33*, 2019–2032. [CrossRef]
14. Zhu, X.; Qian, Y.; Zhao, X. A deep learning approach to patch-based image inpainting forensics. *Signal Process. Image Commun.* **2018**, *67*, 90–99. [CrossRef]
15. Li, H.; Huang, J. Localization of deep inpainting using high-pass fully convolutional network. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 8301–8310.
16. Zhou, P.; Han, X.T.; Morariu, V.I. Learning rich features for image manipulation detection. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 1053–1061.
17. Zhou, P.; Chen, B.C.; Han, X. Generate, segment, and refine: Towards generic manipulation segmentation. In Proceedings of the AAAI Conference on Artificial Intelligence, Hilton New York Midtown, NY, USA, 7–12 February 2020; Volume 32, pp. 13058–13065.
18. Chen, X.; Dong, C.; Ji, J.; Cao, J. Image manipulation detection by multi-view multi-scale supervision. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021.
19. Hu, X.; Zhang, Z.; Jiang, Z. Span: Spatial pyramid attention network forimage manipulation localization. In Proceedings of the European Conference on Computer Vision, ECCV 2020, Glasgow, UK, 23–28 August 2020.
20. Liu, X.; Liu, Y.; Chen, J.; Liu, X. PSCC-Net: Progressive spatio-channel correlation network for image manipulation detection and localization. *IEEE Trans. Circuits Syst. Video Technol.* **2022**, *32*, 7505–7517. [CrossRef]
21. Vaswani, A.; Shazeer, N.; Parmar, N. Attention is all you need. In *Advances in Neural Information Processing Systems 30 (NIPS 2017)*; NIPS Foundation: La Jolla, CA, USA, 2017; Volume 30.
22. Wang, J.; Wu, Z.; Chen, J. ObjectFormer for Image Manipulation Detection and Localization. In Proceedings of the 2022 IEEE Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022.
23. Sun, Y.; Ni, R. ET: Edge-enhanced Transformer for Image Splicing Detection. *IEEE Signal Process. Lett.* **2022**, *29*, 1232–1236. [CrossRef]
24. Huh, M.; Liu, A.; Owens, A.; Efros, A.A. Fighting fake news: Image splice detection via learned self-consistency. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 101–11.
25. Fridrich, J.; Kodovsky, J. Rich models for steganalysis of digital images. *IEEE Trans. Inf. Forensics Secur.* **2012**, *7*, 868–882. [CrossRef]
26. He, K.; Zhang, X.; Ren, S. Deep Residual Learning for Image Recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
27. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A. An image is worth 16 $\times$ 16 words: Transformers for image recognition at scale. *arXiv* **2020**, arXiv:2010.11929.
28. Dong, J.; Wang, W.; Tan, T.N. CASIA image tampering detection evaluation database. In Proceedings of the 2013 IEEE China Summit and International Conference on Signal and Information Processing, Beijing, China, 6–10 July 2013.
29. Nist. Nimble 2016 Datasets. Available online : https://www.nist.gov/itl/iad/mig/nimble-challenge-2017-evaluation (accessed on 5 February 2016).

30. Wen, B.; Zhu, Y.; Subramanian, R. Coverage—A novel database for copy-move forgery detection. In Proceedings of the 2016 IEEE International Conference on Image Processing (ICIP), Phoenix, AZ, USA, 25–28 September 2016; pp. 161–165.
31. Korus, P.; Huang, J. Evaluation of random field models in multi-modal unsupervised tampering localization. In Proceedings of the IEEE International Workshop on Information Forensics and Security, Abu Dhabi, United Arab Emirates, 4–7 December 2016.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.