electronics

MDPI

Article

Multi-Supervised Encoder-Decoder for Image Forgery Localization

Chunfang Yu [1], Jizhe Zhou [2] and Qin Li [1,3,*]

[1] Shanghai Key Laboratory of Trustworthy Computing, East China Normal University, Shanghai 200062, China; cfyu@stu.ecnu.edu.cn
[2] Department of Computer and Information Science, University of Macau, Macau 999078, China; yb87409@um.edu.mo
[3] Shanghai Institute of Intelligent Science and Technology, Tongji University, Shanghai 200092, China
* Correspondence: qli@sei.ecnu.edu.cn

Abstract: Image manipulation localization is one of the most challenging tasks because it pays more attention to tampering artifacts than to image content, which suggests that richer features need to be learned. Unlike many existing solutions, we employ a semantic segmentation network, named Multi-Supervised Encoder–Decoder (MSED), for the detection and localization of forgery images with arbitrary sizes and multiple types of manipulations without extra pre-training. In the basic encoder–decoder framework, the former encodes multi-scale contextual information by atrous convolution at multiple rates, while the latter captures sharper object boundaries by applying upsampling to gradually recover the spatial information. The additional multi-supervised module is designed to guide the training process by multiply adopting pixel-wise Binary Cross-Entropy (BCE) loss after the encoder and each upsampling. Experiments on four standard image manipulation datasets demonstrate that our MSED network achieves state-of-the-art performance compared to alternative baselines.

Keywords: image forgery localization; multi-supervised; atrous convolution; upsampling

Citation: Yu, C.; Zhou, J.; Li, Q. Multi-Supervised Encoder-Decoder for Image Forgery Localization. *Electronics* **2021**, *10*, 2255. https://doi.org/10.3390/electronics10182255

Academic Editor: Gwanggil Jeon

Received: 21 August 2021
Accepted: 9 September 2021
Published: 14 September 2021

1. Introduction

In recent years, image forgery, brought about by the advances of image editing techniques and user-friendly editing software, has negatively affected many aspects of our life and even threatens the stability of society. Hence, it is quite necessary to propose an effective solution to fight against image manipulation and forgery. As shown in Figure 1, splicing, copy-move, and removal are the most common tampering techniques. Among these manipulations, splicing copies regions from the authentic images and pastes them to the other images, copy-move copies and pastes regions within the same image, and removal eliminates regions from the original images. In addition, post-processing, such as Gaussian smoothing [1], is also applied after these tampering techniques to conceal the manipulated traces, which makes it more difficult to recognize the tampered regions.

Diverse manipulating techniques and varying sizes of artifacts prompt us to focus on the higher-dimensional semantic information instead of image content, which is not easy for most models. To alleviate this problem, most recent works [2–4] all apply a large undisclosed synthesized dataset for pre-training, which, to some extent, reduces the generalization ability of the model. Moreover, those methods adopt an instance segmentation network [2] or the accumulation of deep convolution blocks [3,4], which ignore the property of the image forgery detection task. Differently, we propose a semantic segmentation framework to dig deeper into the dataset itself to extract the high-scale manipulated information adaptively and avoid any pre-training process.

More specifically, we adopt an encoder–decoder framework and perform end-to-end training. Following [5], we select ResNet101 [6] as the backbone for its outstanding

performance on semantic segmentation tasks and replace the last few blocks with atrous convolution to dilate the field-of-view of the network. In order to further capture the contextual information at multiple scales, we additionally apply Atrous Spatial Pyramid Pooling (ASPP) [5,7], which concatenates several parallel atrous convolutions with different atrous rates. Even though rich semantic information is encoded in the last feature map, detailed object boundary information is missing due to the pooling or convolutions within the network. To handle this problem, we apply a simple yet effective decoder to gradually recover sharp object boundaries by upsampling. On the other hand, the encoder–decoder frameworks [5,8] lend themselves to faster computation, since no features are dilated.

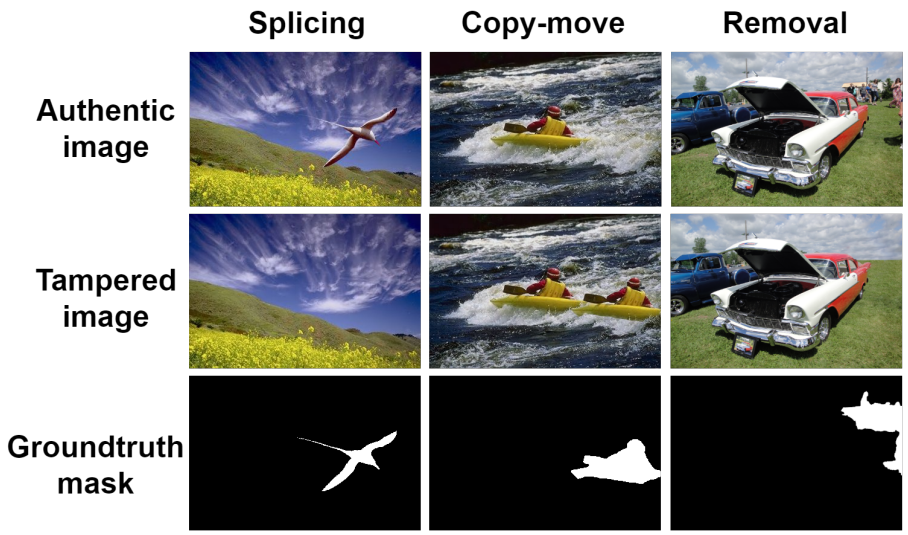

Figure 1. Examples of tampered images with different manipulations. From the (**left**) to (**right**) are the examples showing manipulations of splicing, copy-move and removal.

In particular, our proposed model, called the Multi-Supervised Encoder–Decoder (MSED), extends an encoder–decoder network [5] by adding a multi-supervised module to optimize the experimental performance on benchmarks. With the complex network structure, our basic encoder–decoder has a long process of convolutions and backpropagation, which weakens the supervision of the final pixel-wise classification loss. To alleviate this, we attempt to attach multiple supervision modules to guide the training process of different sub-nets by adopting pixel-wise Binary Cross-Entropy (BCE) loss.

Finally, we adopt the F_1 score and AUC as evaluation metrics, and conduct a set of comparison experiments and ablation studies on standard datasets. The experimental results demonstrate that our proposed MSED shows great performance compared to state-of-the-art methods, which verifies the effectiveness of our proposed model.

In summary, the main contributions of our work can be summarized as follows:

- We propose a Multi-Supervised Encoder–Decoder (MSED) to model high-scale contextual manipulated information and then conduct pixel-wise classification. As far as we know, we are the first to employ the semantic segmentation network for image forgery localization.
- A multi-supervised module is designed to guide the training process and optimize the network performance.
- Experiments on four benchmarks demonstrate that MSED achieves better performance compared to the state-of-the-art works without any pre-training process, which demonstrates the effectiveness of our proposed method.

The remainder of this paper is organized as follows. In Section 2, we review some related work on image forgery localization and CNN-based image semantic segmentation. We present our proposed Multi-Supervised Encoder–Decoder (MSED) in detail in Section 3. In Section 4, experimental results on benchmarks show the effectiveness and outperformance of our

proposed framework. We then conduct an ablation study and analyze the effectiveness of our basic encoder-decoder model and multi-supervised module in Section 5. Finally, we conclude our work and highlight the future research directions in Section 6.

2. Related Work

2.1. Image Forgery Localization

Early research mainly utilizes hand-crafted clues such as resampling [9], Color Filter Artifacts (CFA) [10–12], double JPEG compression [13–15], and Local Noise Analysis (LNA) [16,17] to classify and localize manipulated regions. CFA [10] uses nearby pixels to approximate the camera filter array patterns and then produces the tampering probability for each pixel. Park et al. [15] propose a prediction model to locate tampered areas by examining whether dual JPEG compression exists (tampered regions) or not (authentic regions). NOI [16] is a noise inconsistency-based method using high pass wavelet coefficients to model local noise. Chen et al. [18] propose a Focus Manipulation Inconsistency Histogram (FMIH) framework, which considers five types of features: color variance (VAR), image gradient (GRAD), Double Quantization (DQ) [19], CFA [12], and noise inconsistencies (NOI) [16], and receives five classification results after a neural network from the individual feature. Afterwards, a majority voting scheme is employed to determine the final classification label.

Inspired by these hand-crafted methods, recent work based on an adaptive feature extraction architecture involves similar low-level clues as the additional features and shows promising performance in image forgery localization. However, major existing methods are sensitive to different manipulation techniques, and hence only deal with a specific type of manipulation, such as splicing [17,20–23], copy-move [7,21,24–27], removal [28], and enhancement [29,30]. On the contrary, more recent works manage to break the shackle of manipulation types. J-LSTM [31] constructs a hybrid CNN-LSTM model to learn the boundary discrepancy between forgery and authentic regions by capturing discriminative features between manipulated regions and the boundaries shared with neighboring authentic regions pixels. RGB-N [2] adopts a two-stream Faster R-CNN network to capture noise inconsistency in manipulated artifacts through a Steganalysis Rich Model (SRM) filter, but only localizes manipulations at boundingbox level. ManTra [3] formulates the forgery localization problem as a local anomaly detection problem and designs a self-supervised learning solution to learn robust image manipulation traces. Hu et al. [4] propose a Spatial Pyramid Attention Network (SPAN) to model the relationship between image patches at multiple scales. These end-to-end networks show success in building models that have the robustness to perform the detection and localization of multiple manipulated techniques. Similar to the above methods, our proposed MSED also aims at image forgery detection regardless of manipulation type, and segments the pixel-wise forged mask from a single image.

2.2. CNN-Based Image Semantic Segmentation

The Convolutional Neural Network (CNN) is one of the standard algorithms of deep learning. It contains multi-layer convolutional computation to learn the representation of the input and then obtain the target output for each specific input. A CNN with deep layers shows great performance in digging deep into potential information of data themselves, which is considerably suitable for image semantic segmentation tasks.

Image semantic segmentation is a task of pixel-level classification and focuses on the semantic information instead of the image content. Fully Convolutional Network (FCN) is first proposed in [32] to conduct the pixel-wise prediction, which greatly outperforms traditional methods in image semantic segmentation tasks. The basic encoder–decoder structure also further gradually restores spatial dimensions and detailed information [8,33]. Moreover, atrous convolution [34] is also introduced to expand the Field-of-View (FOV) of the model with a fully connected Conditional Random Field (CRF) at the final Deep Convolutional Neural Network (DCNN) layer to improve the results around the segmentation

boundaries. Further, Chen et al. [35,36] propose Atrous Spatial Pyramid Pooling (ASPP) to robustly segment objects at multiple scales. In [5], they abandon CRF and extend [36] by adding a simple yet effective decoder module to refine the segmentation results, especially along object boundaries. Similarly, image forgery localization is a semantic segmentation task, which focuses on pixel prediction rather than object detection. Therefore, we adopt a semantic segmentation network and follow [5] to build our framework for image forgery localization.

3. Materials and Methods

In this section, we first introduce the overall framework of our proposed Multi-Supervised Encoder–Decoder (MSED). After that, the encoder and decoder network are explained, respectively, in detail. In addition, we present a multi-supervised module, which adopts pixel-wise Binary Cross-Entropy (BCE) loss multiplication to guide the training process of our model.

3.1. The Overall Structure of MSED

An encoder–decoder is a classical architecture for semantic segmentation tasks [5,8], which is composed of two sub-nets, an encoder and decoder:

$$encoder : f(x) = W_1 x + b_1, \quad decoder : g(x) = W_2 f(x) + b_2 \tag{1}$$

From Equation (1), the encoder and decoder have separate networks, and the output of the encoder is fed into the downstream decoder. Generally, the encoder learns to encode the input into some representation, and the corresponding decoder utilizes the generated representation to reconstruct the input. For the semantic segmentation task, the encoder removes fully connected layers of Deep Convolutional Neural Networks to make training smaller and easier. At the same time, the decoder uses the max-pooling indices received from the encoder to perform an upsampling strategy of their input feature maps. The encoder–decoder has several practical advantages:

(i) It improves boundary delineation;
(ii) It reduces the number of parameters enabling end-to-end training.

In our work, we select the encoder–decoder as the basic network architecture for image manipulation localization. As illustrated in Figure 2, the encoder adopts a Deep Convolutional Neural Network to encode the input, while the decoder restores partial pixels to amplify the encoded feature map using upsampling, which is followed by a final pixel-wise classification layer. Moreover, we develop a multi-supervised module to guide the training process. In the sequel, we will describe three important components of our proposed MSED in detail.

3.2. Encoder of Atrous Convolution

For image forgery detection, a big challenge is the extraction of the high-level semantic feature. Therefore, we select atrous convolution [37] to capture multi-scale semantic information through explicitly controlling the resolution of features generated from Deep Convolutional Neural Networks and adjusting the filter's field-of-view, which generalizes the standard convolution operation. In the case of two-dimensional representations, for each pixel i on the output feature map y and a convolution filter w, atrous convolution is applied over the input feature map x as follows:

$$y[i](r, W) = \sum_k x[i + r \cdot k] w[k] \tag{2}$$

where different r determines the stride of the input sample. Here, we employ the standard ResNet101 [6] as the backbone for its outstanding performance on the segmentation task, and utilize atrous convolution to replace the original striding of the last few blocks and extract features at an arbitrary resolution. Following [5], we use output stride to denote the

ratio of input image spatial resolution to the final output resolution before global pooling or the fully-connected layer. In our work, we also adopt output stride = 8 for denser feature extraction and apply atrous convolution with rate = 2 and rate = 4 in the last two blocks of the backbone.

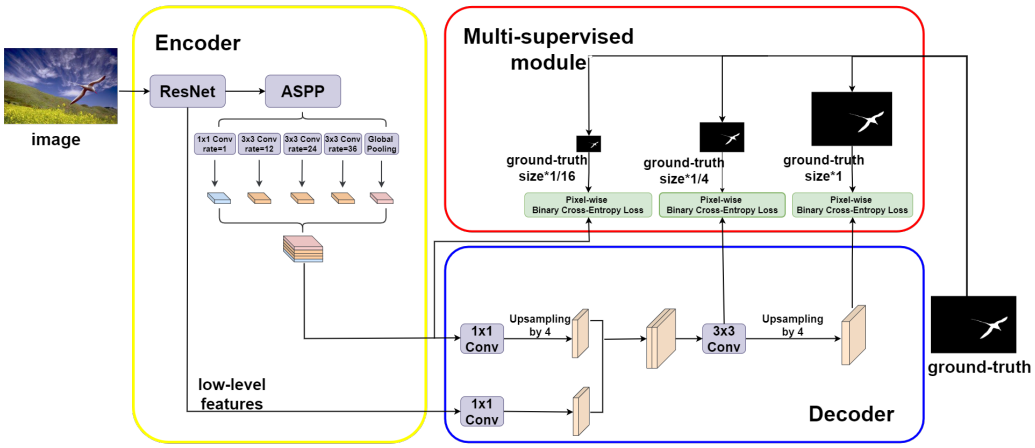

Figure 2. The overview of the proposed Multi-Supervised Encoder–Decoder (MSED) architecture for the image forgery detection and localization task, which contains three sub-modules: encoder (yellow outer box), decoder (blue outer box), and multi-supervised module (red outer box). In the multi-supervised module, we calculate pixel-wise Binary-Cross Entropy (BCE) loss between the ground-truth mask after interpolation and the encoded feature map after the encoder and each upsampling, which is highlighted by the green cubes.

After the backbone, we attach an Atrous Spatial Pyramid Pooling (ASPP) [5,36] block to capture multi-scale contextual information. As illustrated in Figure 3, ASPP contains four parallel atrous convolutions with different atrous rates. With the same two-dimensional representations of Equation (2), for each pixel i on the output feature map Y is the combination of different feature map y_j with separate stride r_j and convolution filter w_j:

$$Y[i] = \sum_j y_j[i](r_j, w_j) \tag{3}$$

By combining the atrous convolution layer with different rate values, we are able to resample features at different scales to accurately and efficiently classify regions of an arbitrary scale. Here, we utilize the atrous rates r = 1, 12, 24, and 36. After the ASPP block, a 1×1 convolution layer is attached to flatten the feature map, and then the encoded feature map is passed to the downstream decoder.

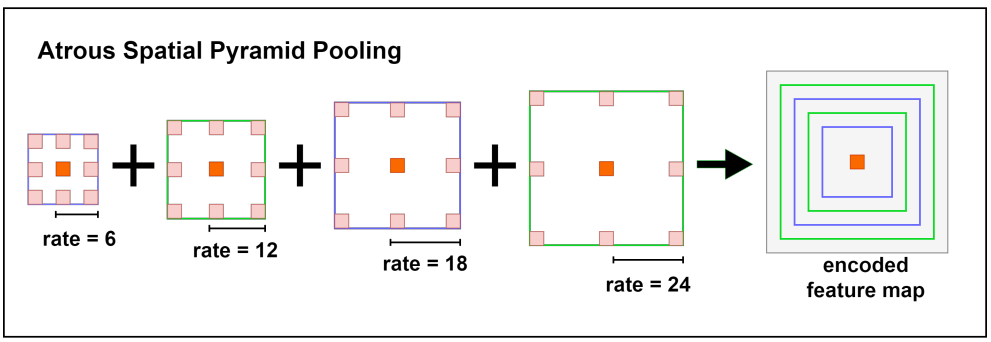

Figure 3. This is Atrous Spatial Pyramid Pooling (ASPP). The kernel of each convolution block is 3×3. The final encoded feature map concatenates multiple parallel filters with different rates, which are shown as the outer colorful box.

3.3. Decoder of Upsampling

The process of encoding features sometimes discards respectably essential information, especially high-scale semantic information. Recent works for semantic segmentation tasks [5,8] utilize interpolation-based upsampling to recover spatial details for sharper segmentations.

Interpolation is a method of constructing new data points within the range of a discrete set of known data points. Image interpolation refers to the "guess" of intensity values at missing locations. The most commonly used algorithms are the nearest interpolation, bilinear interpolation, and bicubic interpolation [38]. Here, we adopt bilinear interpolation [38] for its better performance and lower computation cost, which is the most commonly utilized in segmentation tasks. It utilizes the four nearest pixel values, which are located in diagonal directions from a given location to find the appropriate intensity values of the target pixel. After that, we can generate smoother feature maps that are closer to original images.

As shown in Figure 4, if we want to find the pixel value of the point P, we should first calculate the pixel value of R_1 and R_2 using a weighted average of (Q_{11}, Q_{21}) and (Q_{12}, Q_{22}), respectively, and then use a weighted average of R_2 and R_1 to find the value of P. Effectively, we interpolate in the x direction and then the y direction, or we could just as well flip the order of interpolation and obtain the exact same value. Given a point $P = (x, y)$ and four corner coordinates $Q_{11} = (x_1, y_1)$, $Q_{21} = (x_2, y_1)$, $Q_{12} = (x_1, y_2)$ and $Q_{22} = (x_2, y_2)$, we first interpolate in the x-direction:

$$\begin{aligned} f(R_1) &\approx \frac{x_2 - x}{x_2 - x_1} f(Q_{11}) + \frac{x - x_1}{x_2 - x_1} f(Q_{21}) \\ f(R_2) &\approx \frac{x_2 - x}{x_2 - x_1} f(Q_{12}) + \frac{x - x_1}{x_2 - x_1} f(Q_{22}) \end{aligned} \tag{4}$$

and finally, in the y-direction:

$$f(P) \approx \frac{y_2 - y}{y_2 - y_1} f(R_1) + \frac{y - y_1}{y_2 - y_1} f(R_2) \tag{5}$$

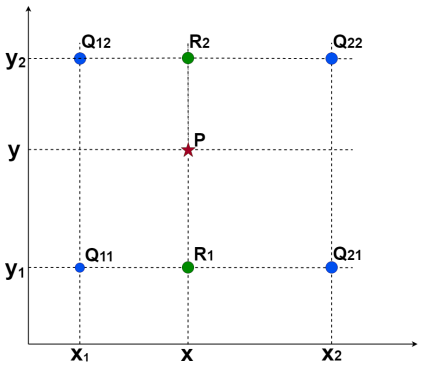

Figure 4. This is the example of bilinear interpolation. The red star P is our target. To obtain the target, we first calculate the pixel value of R_1 and R_2 by interpolating in the x direction using a weighted average of (Q_{11}, Q_{21}) and (Q_{12}, Q_{22}), respectively, and then interpolate in the y direction according to R_1 and R_2 to find the value of P.

In our work, we adopt a bilinear interpolation-based decoder [5,8] to capture sharper object boundaries by gradually recovering the spatial information. As demonstrated in Figure 2, first, bilinear upsampling is used after the encoder by a factor of 4. Furthermore, then we concatenate it with the corresponding low-level feature map from the second block in the ResNet101 backbone that is captured by another 1×1 convolution to reduce the number of channels. After the concatenation, we apply a few 3×3 convolutions to refine the features and then followed by another same bilinear upsampling by a factor of

4. Afterwards, the inflated feature map is the same size as the input image for pixel-wise manipulated classification.

3.4. Multi-Supervised Module

The goal of image forgery detection is to localize the manipulated regions, which is a pixel-wise classification task. However, it is hard to correctly capture high-scale semantic manipulated features and then identify the manipulated pixels. Moreover, the multiple convolution blocks and upsampling processes also discard a few important information and weaken the effectiveness of the supervision. Hence, we attempt to develop a multi-supervised module to guide the training process and optimize the classification performance. Specifically, we adopt the pixel-wise Binary Cross-Entropy (BCE) loss due to its robustness on segmentation tasks to supervise the pixel classification of the feature map:

$$l_{pbce}(f(X), G) = -\frac{1}{m}\sum_{i}^{m} G_i log(f(X_i)) + (1 - G_i) log(1 - f(X_i)) \tag{6}$$

$$L_{PBCE}(X, G) = l_{pbce}(f_1(x), \frac{1}{16}G) + l_{pbce}(f_2(x), \frac{1}{4}G) + l_{pbce}(f_3(x), G) \tag{7}$$

where $f(X_i)$ and G_i represent every pixel in the different feature map ($f(X)$) and ground truth (G) for the corresponding input X, respectively. i indicates the index of each pixel, while m represents the number of pixels. The manipulated artifacts and the authentic background conduct the contrastive pairs, supervised by a binary ground-truth mask with a 1 label denoting tampered pixels, while a 0 label denotes authentic pixels. As illustrated in Figure 2, we implement supervision after the encoder and each upsampling. Afterwards, we summarize the total loss and then back-propagate it continuously to train our network. The experimental results and ablation study verify the effectiveness of our proposed multi-supervised module. We provide more experimental details in Sections 4 and 5.

4. Results

4.1. Datasets

In this work, we follow the current SoTA methods for image forgery localization tasks and utilize four standard datasets for training and evaluation, including CASIA [1], NIST Nimble 2016 (NIST16) [39], Columbia [40], and Coverage [25]. These public datasets contain forgery images with common manipulated techniques, which are considered suitable for our task. Moreover, we aim to compare them fairly with baselines by adopting the same datasets.

- CASIA [1] provides spliced and copy-moved images with binary ground-truth masks. We use CASIA 2.0 for training and CASIA 1.0 for evaluation. CASIA 1.0 contains 921 samples, while CASIA 2.0 includes 5123 samples. They also apply image enhancement techniques such as filtering and blurring to post-process the samples.
- NIST16 [39] is a standard image manipulation dataset that contains three tampered techniques, including splicing, copy-move, and removal. They provide 564 manipulated images and corresponding binary ground-truth masks. Samples of NIST16 are post-processed to hide visible traces.
- Columbia [40] contains 180 splicing forged images with provided edge masks. We transform the edge masks into binary ground-truth masks, in which 1 denotes manipulated pixels, while 0 represents authentic pixels.
- Coverage [25] is a copy-move forgery dataset that only contains 100 samples with corresponding binary masks. It copies objects to another similar region to before in order to conceal the manipulated artifacts.

Following the practices in [2–4], we split each dataset into 75–25% for training and testing, except CASIA (CASIA 2.0 for training and CASIA 1.0 for testing). Besides for the fair comparisons, the proper allocation of training sets and test sets can also help avoid

overfitting of the model. After generating and organizing these datasets, we train and test our model on these splits.

Synthesized Pre-training Dataset: Some SoTA methods apply synthetic datasets for pre-training. RGB-N [2] creates a synthetic dataset with 42K tampered and authentic image pairs using the images and annotations from COCO [41]. ManTra-Net [3] uses four synthetic datasets, including the splicing dataset from [42], the copy-move dataset from [27], the synthesized removal dataset, and the enhancement dataset based on Dresden [43]. SPAN [4] also applies these four synthetic datasets for pre-training. Unlike these baselines, our proposed method only utilizes four standard datasets for training and evaluation without any extra pre-training dataset. Therefore, unlike those using Columbia to fine-tune the pre-trained model, we split 135 samples of Columbia for training and the rest for testing.

4.2. Experimental Details

As discussed in Section 3.1 and demonstrated in Figure 2, our proposed network is trained end-to-end and implemented through Python and PyTorch. For the network, we choose atrous rates with 1, 12, 24, and 36 in the ASPP block as [5]. In addition, we conduct early-stopping at 70 epochs to avoid overfitting and adopt a Stochastic Gradient Descent (SGD) optimizer with the initial learning rate of 0.007, momentum of 0.9, and weight decay of 5×10^{-4}, which can adapt the step size automatically during the training process. We set the batch size to 4 and crop size to 512×512 on each dataset. We apply the parameters for the whole training and validation process.

4.3. Evaluation Metrics

To achieve our goal of classifying the tampered pixels from authentic ones, it is necessary to use evaluation measures or metrics to assess the performance of our proposed method. Considering the imbalance of the size between different classifications, we follow [2–4] and adopt the F_1 score and the Area Under the Receiver Operator Characteristic Curve (AUC) to assess the model performance. We first define the *Precision* and *Recall* as:

$$Precision = \frac{TP}{TP + FP} \tag{8}$$

$$Recall = \frac{TP}{TP + FN} \tag{9}$$

where TP (True Positive) and TN (True Negative) refer to tampered pixels and authentic ones, which are correctly classified. FP (False Positive) and FN (False Negative) refer to tampered pixels and authentic ones that are misclassified. As defined in Equations (8) and (9), *Precision* or confidence is the number of pixels correctly assigned to be tampered compared to the total number of pixels predicted as tampered ones (total predicted positive), while *Recall* or sensitivity is the number of pixels correctly assigned to be tampered compared to the total number of pixels belonging to the tampered regions (total true positive). After generating *Precision* and *Recall*, we then calculate the F_1 score and AUC as:

$$F_1\text{score} = \frac{Precision \times Recall}{Precision + Recall} \tag{10}$$

$$\text{AUC} = \frac{\sum_{i \in positiveClass} rank_i - \frac{M(1+M)}{2}}{M \times N} \tag{11}$$

where $rank_i$ represents the index of i-th sample after ordering by increasing probability. M and N denote the number of positives and negatives. From Equation (10), the F_1 score subtly combines *Precision* and *Recall* to measure the performance of the classification. Differently, AUC evaluates how the number of correctly classified positive examples varies with the number of incorrectly classified negative examples.

Hence, we adopt the pixel-level F_1 score and AUC as our evaluation metrics for performance comparison. According to the pixel-level confusion matrix, we first aggregate all TP, FP, TN, and FN numbers over the whole dataset and then calculate the *Precision*, *Recall*, F_1 score, and AUC for each epoch. We evaluate the F_1 score and AUC at the validation of each epoch and pick the best model from the highest AUC score.

4.4. Evaluation and Comparisons

4.4.1. Baseline Models

We evaluate and compare our proposed model's performance on four standard datasets with current SoTA methods, including ELA [13], NOI1 [16], CFA1 [10], J-LSTM [31], RGB-N [2], ManTra [3] and SPAN [4] as described below:

- ELA: An error level analysis method [13] which aims to apply different JPEG compression qualities to find the compression error difference between tampered regions and authentic regions.
- NOI1: A noise inconsistency-based method detecting changes in noise level to capture manipulated information [16].
- CFA1: A Camera Filter Array (CFA) pattern estimation method [10] which approximates the CFA patterns using nearby pixels and then produces the tampering probability for each pixel.
- J-LSTM: An LSTM-based network [31] jointly training patch-level tampered edge classification and pixel-level tampered region segmentation.
- RGB-N: Bilinear pooling of RGB stream [2] and noise stream for manipulation classification.
- ManTra: An LSTM-based local anomaly detection network [3] which formulates the forgery localization problem as a local anomaly detection problem and captures the local anomaly.
- SPAN: A Spatial Pyramid Attention Network (SPAN) [4] which models the relationship between image patches at multiple scales by constructing a pyramid of local self-attention blocks.
- MSED (Ours): An encoder–decoder focusing on the spatial semantic manipulated information by atrous convolution with an additional multi-supervised module in the training process.

4.4.2. Comparisons

We list the results of the evaluation and comparisons in Table 1. From the table, we can observe that our MSED achieves better performance than conventional methods such as ELA [13], NOI1 [16] and CFA1 [10]. This is because they all focus on specific tampering artifacts that only contain partial information for localization, which limits their performance. One of the reasons our method outperforms J-LSTM [31] is that it seeks tampered edges as evidence of tampering, which cannot always detect the entire tampered regions. Compared to RGB-N [2], Mantra [3], and SPAN [4], which use a large synthesized dataset for pre-training and benchmarks for fine-tuning, our MSED performs without any pre-training and fine-tuning process and achieves state-of-the-art performance, especially in the F_1 score. This is probably because they rely on low-level clues as the additional features, while we aim to capture the high-scale semantic information brought by manipulated operations through a semantic segmentation network with multiple supervision modules.

As for the AUC, except the CASIA dataset, MSED achieves state-of-the-art performance without any pre-training and fine-tuning process. We explore the potential reasons for the worse AUC on CASIA and find that CASIA applies image enhancement techniques such as filtering and blurring to hide visible traces, which blurs the semantic information of manipulated artifacts. Moreover, CASIA contains many copy-move images that are constructed by flipping or shifting a certain part of the authentic image, or by copying one from several similar patterns and moving to another. Therefore, these copy-move manipulated images contain a lot of irrelevant semantic features related to the objects on the image, and the tampered information is not apparent. We show some samples in

Figure 5. From the right side, we can see MSED is still very effective in comparison to the other copy-move manipulations.

Table 1. F_1 score (%) and AUC (%) comparisons between our proposed method and baselines on benchmarks. *pt* donates pre-training, while *ft* represents fine-tuning process.

Method	*pt*	*ft*	NIST16		CASIA		Coverage		Columbia	
			F_1	AUC	F_1	AUC	F_1	AUC	F_1	AUC
ELA [13]	×	×	23.6	42.9	21.4	61.3	22.2	58.3	47.0	58.1
NOI1 [16]	×	×	28.5	48.7	26.3	61.2	26.9	58.7	57.4	54.6
CFA1 [10]	×	×	17.4	50.1	20.7	52.2	19.0	48.5	46.7	72.0
J-LSTM [31]	✓	✓	-	76.4	-	-	-	61.4	-	-
ManTra [3]	✓	✓	-	79.5	-	81.7	-	81.9	-	82.4
RGB-N [2]	✓	✓	72.2	93.7	40.8	79.5	43.7	81.7	69.7	85.8
SPAN (1) [4]	✓	×	29.0	83.6	33.6	81.4	53.5	91.2	81.5	93.6
SPAN (2) [4]	✓	✓	58.2	96.1	38.2	83.8	55.8	93.7	-	-
MSED (Ours)	×	×	96.0	96.2	74.7	67.8	95.1	96.1	94.6	94.5

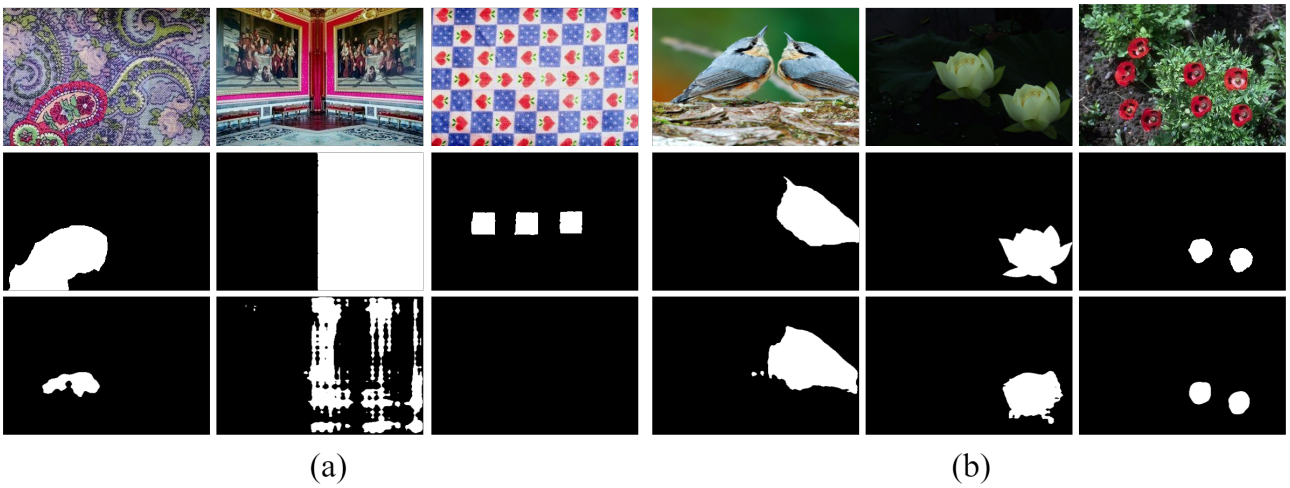

Figure 5. MSED detection results on copy-moved images on CASIA. (**a**) represents manipulated images with flipping manipulation and similar patterns within an image which are difficult to localize. (**b**) denotes the other types of copy-moved images.

F_1 score and model robustness: A good image forgery localization model shall have decent F_1 and AUC values simultaneously, even though F_1 is a pixel-wise measurement and AUC is the value of an integral. Checking Table 1, current methods share a huge gap between their F_1 score and AUC; this indicates the classification result of the model is, in fact, unsatisfying, but it could be improved by finding a better-suited classification threshold. Such a result further verifies the weakness of adaption for the hand-crafted features: they are very confusing in distinguishing the hard examples, and thereby, require a meticulous threshold to achieve high classification performance. Besides, considering the massive synthesized dataset they adopt for pre-training as well as the fine-tuning process on a relatively tiny benchmark dataset, the inconsistent behavior between the F_1 score and AUC should be the symbol of overfitting. MSED addresses this potential overfitting issue by focusing on the high-dimensional semantic information of the dataset itself and avoiding an extensive pre-training process. As a result, MSED achieves similar and consistent performance across the measurements, which verifies that our MSED has higher robustness and is less prone to overfitting.

4.5. Qualitative Result

We show some qualitative results in Figure 6 for comparison of a basic encoder–decoder and MSED in two-class image manipulation detection on four benchmarks. As shown in Figure 6, the basic encoder–decoder is effective for image forgery detection, and our MSED produces sharper manipulated artifacts with clear contours for different tampering techniques.

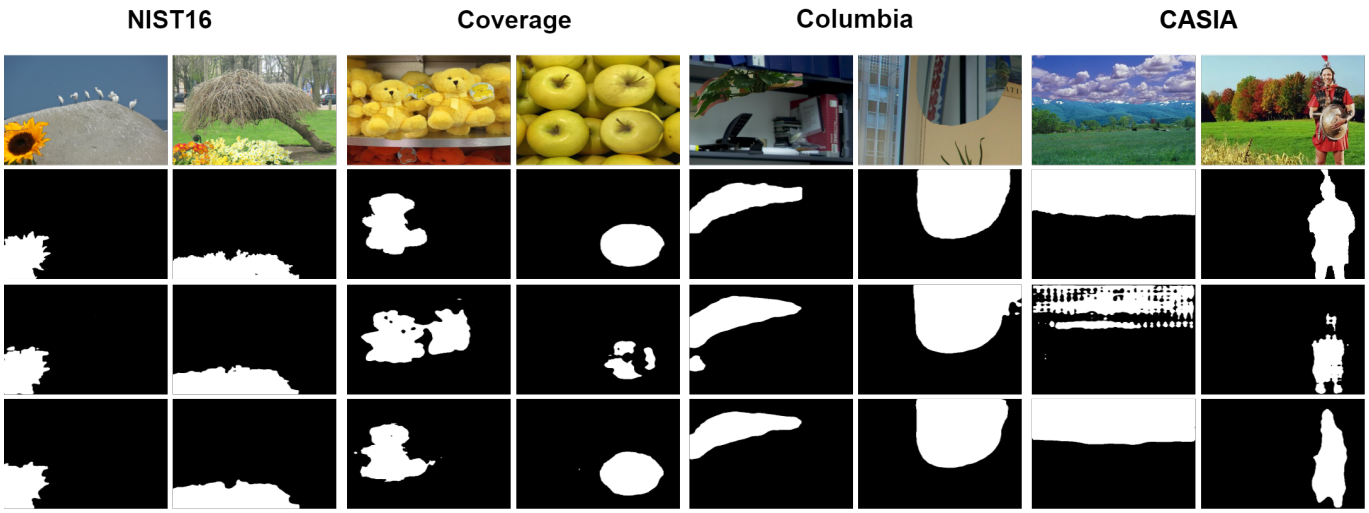

Figure 6. Qualitative visualization of prediction results on NIST16, Coverage, Columbia, and CASIA. From top to bottom are the manipulated image, corresponding ground-truth, the prediction results of basic encoder–decoder, and our proposed Multi-Supervised Encoder–Decoder model.

5. Discussion

We carefully conducted a set of ablation experiments to study the effectiveness of the multi-supervised module on four standard datasets. To ensure fair comparisons, all experiments shared the same implementation settings and differed from each other only in the components of the multi-supervised module. One may refer to Section 4.2 for more details.

Figures 7 and 8 show the F_1 score and AUC comparisons of the basic encoder–decoder framework and its advanced version (MSED) with the multi-supervised component in the training process. From the figure, our proposed model has similar and consistent performance across the measurements, which differs from previous works with high AUC but worse F_1 scores (as shown in Table 1). Therefore, MSED has higher robustness and is less prone to overfitting.

On the other hand, we can obviously observe that our proposed multi-supervised module achieves significant performance improvement compared to the original encoder–decoder on the benchmarks, especially on CASIA. This is probably because the basic encoder–decoder performed well on the other three datasets; hence, there is less scope for advancement. Moreover, MSED trains faster compared to the basic encoder–decoder, and the evaluation results reach the peak earlier, which verifies the effectiveness of our proposed multi-supervised module.

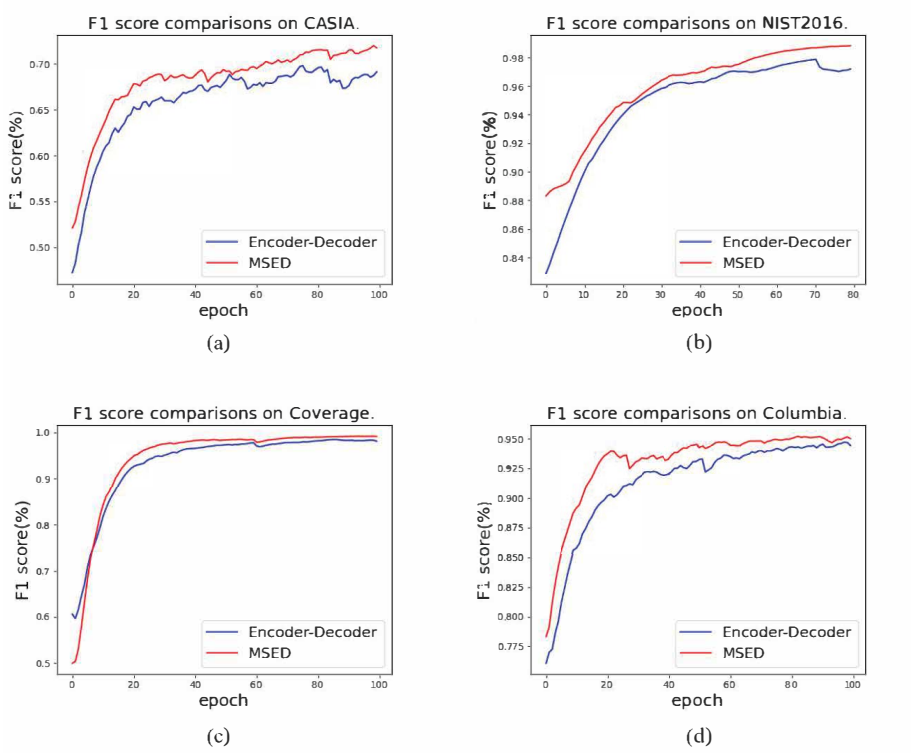

Figure 7. F_1 score (%) comparisons of MSED variants evaluated on CASIA (**a**), NIST2016 (**b**), Coverage (**c**), and Columbia (**d**). In each sub-graph, the blue curve represents the results of the basic encoder–decoder network, while the red curve means the results of the our proposed MSED model with additional multi-supervised module.

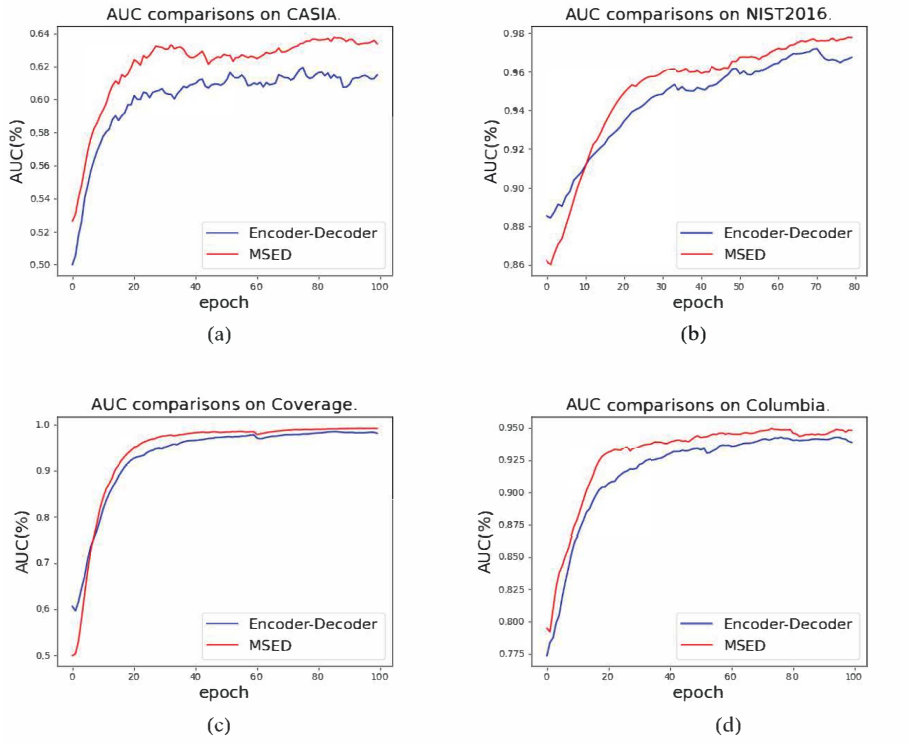

Figure 8. AUC (%) comparisons of MSED variants evaluated on CASIA (**a**), NIST2016 (**b**), Coverage (**c**), and Columbia (**d**). In each sub-graph, the blue curve represents the results of the basic encoder–decoder network, while the red curve means the results of the our proposed MSED model with additional multi-supervised module.

6. Conclusions

In this paper, we present a novel semantic segmentation network, named Multi-Supervised Encoder-Decoder (MSED), that encodes rich multi-scale contextual information and localizes forgery images with multiple manipulated techniques. For the basic encoder–decoder, the encoder applies the atrous convolution to extract the semantic features at an arbitrary resolution, while the decoder recovers the object boundaries by upsampling. Moreover, we propose a multi-supervised module to guide the training process. Our extensive experimental results on standard datasets demonstrate that our proposed MSED is not only sensitive to subtle manipulations and robust to post-processing disguising manipulations, but also outperforms the state-of-the-art models and performs without any pre-training processes.

In the future, we will attempt to create a synthetic dataset and explore more manipulated techniques to break through the limitation of the current image manipulation datasets in public. Another promising direction is manipulation classification, which helps to identify the specific tampering type for the forgery image.

Author Contributions: Conceptualization, C.Y. and J.Z.; methodology, C.Y. and J.Z.; software, C.Y.; validation, C.Y.; formal analysis, Q.L.; investigation, C.Y. and J.Z.; resources, C.Y.; data curation, C.Y.; writing—original draft preparation, C.Y.; writing—review and editing, C.Y., J.Z. and Q.L.; visualization, C.Y.; supervision, J.Z. and Q.L.; project administration, C.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Key Research and Development Program (grant number: 2020AAA0107800), and AI Project of Shanghai Science and Technology Committee (grant number: STCSM 20DZ1100300).

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

References

1. Dong, J.; Wang, W.; Tan, T. 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; pp. 422–426. [CrossRef]
2. Zhou, P.; Han, X.; Morariu, V.I.; Davis, L.S. 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, 18–22 June 2018; pp. 1053–1061. [CrossRef]
3. Wu, Y.; AbdAlmageed, W.; Natarajan, P. ManTra-Net: Manipulation Tracing Network for Detection and Localization of Image Forgeries with Anomalous Features. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019; pp. 9535–9544. [CrossRef]
4. Hu, X.; Zhang, Z.; Jiang, Z.; Chaudhuri, S.; Yang, Z.; Nevatia, R. SPAN: Spatial Pyramid Attention Network forImage Manipulation Localization. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020; Springer: Cham, Switzerland, 2020.
5. Chen, L.C.; Zhu, Y.; Papandreou, G.; Schroff, F.; Adam, H. Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018.
6. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778. [CrossRef]
7. Islam, A.; Long, C.; Basharat, A.; Hoogs, A. DOA-GAN: Dual-Order Attentive Generative Adversarial Network for Image Copy-Move Forgery Detection and Localization. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 14–19 June 2020; pp. 4675–4684. [CrossRef]
8. Badrinarayanan, V.; Kendall, A.; Cipolla, R. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 2481–2495. [CrossRef] [PubMed]
9. Bunk, J.; Bappy, J.H.; Mohammed, T.M.; Nataraj, L.; Flenner, A.; Manjunath, B.S.; Chandrasekaran, S.; Roy-Chowdhury, A.K.; Peterson, L. Detection and Localization of Image Forgeries Using Resampling Features and Deep Learning. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Honolulu, HI, USA, 21–26 July 2017; pp. 1881–1889. [CrossRef]
10. Ferrara, P.; Bianchi, T.; De Rosa, A.; Piva, A. Image Forgery Localization via Fine-Grained Analysis of CFA Artifacts. *IEEE Trans. Inf. Forensics Secur.* **2012**, *7*, 1566–1577. [CrossRef]

11. Alattar, A.M.; Memon, N.D.; Heitzenrater, C.D.; Goljan, M.; Fridrich, J. CFA-aware features for steganalysis of color images. *Proc. SPIE Int. Soc. Opt. Eng.* **2015**, *9409*, 94090V.
12. Dirik, A.E.; Memon, N. Image tamper detection based on demosaicing artifacts. In Proceedings of the 2009 16th IEEE International Conference on Image Processing (ICIP), Cairo, Egypt, 7–10 November 2009; pp. 1497–1500. [CrossRef]
13. Krawetz, N. A picture's worth. *Hacker Factor Solut.* **2007**, *6*, 2.
14. Amerini, I.; Uricchio, T.; Ballan, L.; Caldelli, R. Localization of JPEG double compression through multi-domain convolutional neural networks. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Honolulu, HI, USA, 21–26 July 2017.
15. Park, J.; Cho, D.; Ahn, W.; Lee, H.K. Double JPEG Detection in Mixed JPEG Quality Factors Using Deep Convolutional Neural Network. In Proceedings of the 15th European Conference, Munich, Germany, 8–14 September 2018.
16. Mahdian, B.; Saic, S. Using noise inconsistencies for blind image forensics. *Image Vis. Comput.* **2009**, *27*, 1497–1503. [CrossRef]
17. Cozzolino, D.; Poggi, G.; Verdoliva, L. Splicebuster: A new blind image splicing detector. In Proceedings of the 2015 IEEE International Workshop on Information Forensics and Security (WIFS), Roma, Italy, 16–19 November 2015; pp. 1–6. [CrossRef]
18. Mccloskey, S.; Chen, C.; Yu, J. Focus Manipulation Detection via Photometric Histogram Analysis. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018.
19. Bianchi, T.; De Rosa, A.; Piva, A. Improved DCT coefficient analysis for forgery localization in JPEG images. In Proceedings of the 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Prague, Czech Republic, 22–27 May 2011.
20. Huh, M.; Liu, A.; Owens, A.; Efros, A.A. Fighting Fake News: Image Splice Detection via Learned Self-Consistency. In Proceedings of the Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018.
21. Rao, Y.; Ni, J. A deep learning approach to detection of splicing and copy-move forgeries in images. In Proceedings of the 2016 IEEE International Workshop on Information Forensics and Security (WIFS), Abu Dhabi, United Arab Emirates, 4–7 December 2016; pp. 1–6. [CrossRef]
22. Kniaz, V.V.; Knyaz, V.A.; Remondino, F. The Point Where Reality Meets Fantasy: Mixed Adversarial Generators for Image Splice Detection. In Proceedings of the Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, Vancouver, BC, Canada, 8–14 December 2019; pp. 215–226.
23. Salloum, R.; Ren, Y.; Kuo, C.C.J. Image Splicing Localization Using A Multi-Task Fully Convolutional Network (MFCN). *J. Vis. Commun. Image Represent.* **2017**, *51*, 201–209. [CrossRef]
24. Cozzolino, D.; Poggi, G.; Verdoliva, L. Efficient Dense-Field Copy–Move Forgery Detection. *IEEE Trans. Inf. Forensics Secur.* **2015**, *10*, 2284–2297. [CrossRef]
25. Wen, B.; Zhu, Y.; Subramanian, R.; Ng, T.T.; Winkler, S. COVERAGE—A novel database for copy-move forgery detection. In Proceedings of the IEEE International Conference on Image Processing, Phoenix, AZ, USA, 25–28 September 2016.
26. Yue, W.; Abd-Almageed, W.; Natarajan, P. Image Copy-Move Forgery Detection via an End-to-End Deep Neural Network. In Proceedings of the 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), Lake Tahoe, NV, USA, 12–15 March 2018.
27. Wu, Y.; Abdalmageed, 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.
28. Zhu, X.; Qian, Y.; Zhao, X.; Sun, B.; Sun, Y. A deep learning approach to patch-based image inpainting forensics. *Signal Process. Image Commun.* **2018**, *67*, 90–99. [CrossRef]
29. Bayar, B.; Stamm, M. A Deep Learning Approach to Universal Image Manipulation Detection Using a New Convolutional Layer. In Proceedings of the 4th ACM Workshop on Information Hiding and Multimedia Security, Vigo, Galicia, Spain, 20–22 June 2016; pp. 5–10. [CrossRef]
30. Bayar, B.; Stamm, M.C. Constrained Convolutional Neural Networks: A New Approach Towards General Purpose Image Manipulation Detection. *IEEE Trans. Inf. Forensics Secur.* **2018**, *13*, 2691–2706. [CrossRef]
31. Bappy, J.H.; Roy-Chowdhury, A.K.; Bunk, J.; Nataraj, L.; Manjunath, B.S. Exploiting Spatial Structure for Localizing Manipulated Image Regions. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 4980–4989. [CrossRef]
32. Long, J.; Shelhamer, E.; Darrell, T. Fully Convolutional Networks for Semantic Segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *39*, 640–651.
33. Ronneberger, O.; Fischer, P.; Brox, T. U-Net: Convolutional Networks for Biomedical Image Segmentation. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Munich, Germany, 5–9 October 2015.
34. Chen, L.C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A. Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected CRFs. *arXiv* **2014**, arXiv:1412.7062.
35. Chen, L.C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *40*, 834–848. [CrossRef] [PubMed]
36. Chen, L.C.; Papandreou, G.; Schroff, F.; Adam, H. Rethinking Atrous Convolution for Semantic Image Segmentation. *arXiv* **2017**, arXiv:1706.05587.

37. Papandreou, G.; Kokkinos, I. Untangling Local and Global Deformations in Deep Convolutional Networks for Image Classification and Sliding Window Detection. *arXiv* **2014**, arXiv:1412.0296.
38. Jaderberg, M.; Simonyan, K.; Zisserman, A.; Kavukcuoglu, K. Spatial Transformer Networks. *arXiv* **2016**, arXiv:1506.02025.
39. NIST: Nist Nimble 2016 Datasets. 2016. Available online: https://www.nist.gov/itl/iad/mig/ (accessed on 13 June 2016).
40. Hsu, Y.-F.; Chang, S.-F. Detecting Image Splicing Using Geometry Invariants and Camera Characteristics Consistency. In Proceedings of the IEEE International Conference on Multimedia and Expo, Seattle, WA, USA, 26 December 2006.
41. Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Zitnick, C.L. Microsoft COCO: Common Objects in Context. In Proceedings of the Computer Vision—ECCV 2014, Zurich, Switzerland, 6–12 September 2014; pp. 740–755.
42. Wu, Y.; Abd-Almageed, W.; Natarajan, P. Deep Matching and Validation Network: An End-to-End Solution to Constrained Image Splicing Localization and Detection. In Proceedings of the 25th ACM International Conference on Multimedia, Mountain View, CA, USA, 23–27 October 2017; pp. 1480–1502. [CrossRef]
43. Gloe, T.; Bohme, R. The Dresden Image Database for Benchmarking Digital Image Forensics. *J. Digit. Forensic Pract.* **2010**, *3*, 150–159. [CrossRef]