

# CFL-Net: Image Forgery Localization Using Contrastive Learning

Fahim Faisal Niloy<sup>†</sup>, Kishor Kumar Bhaumik<sup>‡</sup> and Simon S. Woo<sup>‡</sup>

<sup>†</sup>Center for Computational & Data Sciences, Independent University, Bangladesh

<sup>‡</sup>Computer Science and Engineering Department, Sungkyunkwan University, Suwon, South Korea

niloy9542@gmail.com, {kishor25, swoo}@g.skku.edu

## Abstract

Conventional forgery localizing methods usually rely on different forgery footprints such as JPEG artifacts, edge inconsistency, camera noise, etc., with cross-entropy loss to locate manipulated regions. However, these methods have the disadvantage of over-fitting and focusing on only a few specific forgery footprints. On the other hand, real-life manipulated images are generated via a wide variety of forgery operations and thus, leave behind a wide variety of forgery footprints. Therefore, we need a more general approach for image forgery localization that can work well on a variety of forgery conditions. A key assumption in underlying forged region localization is that there remains a difference of feature distribution between untampered and manipulated regions in each forged image sample, irrespective of the forgery type. In this paper, we aim to leverage this difference of feature distribution to aid in image forgery localization. Specifically, we use contrastive loss to learn mapping into a feature space where the features between untampered and manipulated regions are well-separated for each image. Also, our method has the advantage of localizing manipulated region without requiring any prior knowledge or assumption about the forgery type. We demonstrate that our work outperforms several existing methods on three benchmark image manipulation datasets. Code is available at <https://github.com/niloy193/CFLNet>

## 1. Introduction

Image forgery has been a serious emerging socio-technical issue, as more advanced AI techniques have been leveraged to create fake images. Image is a significant medium for information transfer. In order to produce fake stories, academic trickery, and illegal conduct, manipulated photographs created utilizing image editing technology are constantly being mistaken for real ones. When a digital im-

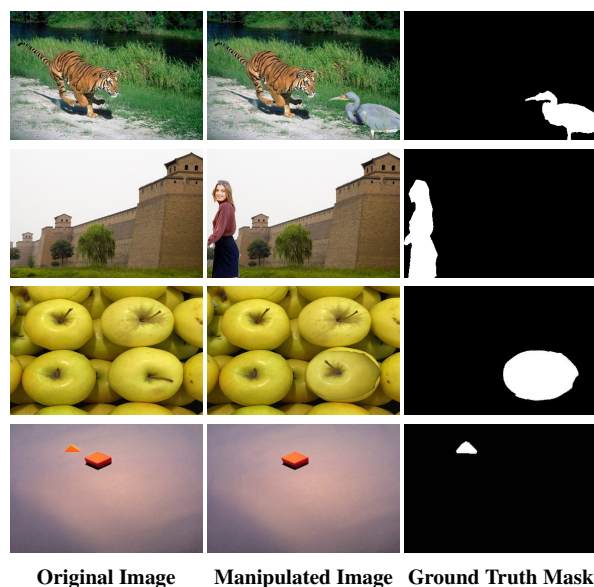


Figure 1: Examples of image manipulation. First two rows show examples of image splicing and the next two rows show examples of copy-move forgery and removal respectively.

age is manipulated, we frequently assume that image forensic investigations will be able to spot the tampered areas. However, collecting differentiating features of tampered areas with various forging types (including splicing, copy-move, removal, etc.) is still challenging and typically calls for utilizing the special qualities of numerous tampering artifacts.

Generally, image forgery can be broadly categorized into: splicing [12, 25], copy-move [11, 36, 35], removal [42], enhancement [4, 9], etc. First, in image splicing, content is copied and pasted from other source images, as opposed to copy-move forgery, where the content is obtained from the same image. On the other hand, removal

or inpainting techniques remove a selected region from the image and fills the space with new pixel values estimated from background [37]. Image enhancement exploits a wide collection of local manipulations, such as sharpening, brightness adjustment, etc. Each of the broader categories can be further divided into more fine-grained forgery types. For example, Gaussian blurring or JPEG compression may be applied to the tampered region before committing splicing or copy-move forgery. Recently, more general-purpose image forgery localization methods have been proposed, which can detect or localize more than one forgery type, such as RGB-N Net [41], Manipulation Tracing Network (ManTraNet) [37], Spatial Pyramid Attention Network (SPAN) [22], etc.

These general image forgery detection or localization methods usually rely on different forgery clues or footprints left by the forgery operation, such as JPEG artifacts [27, 1], edge inconsistency [32, 39], noise pattern [13, 38], camera model [31], EXIF inconsistency [23], etc., to detect or localize forgery. Table 1 of [37] summarizes existing major forgery localization methods and the forgery clues the methods focus on. For example, [2] employs LSTM based patch comparison to focus on edge inconsistency between the tampered patches and authentic patches. CAT-Net [26] leverages DCT coefficients to focus on resampling clues.

However, training models to focus on specific forgery clues has a major disadvantage. Because then, the model can only detect forgery if that particular forgery footprint is prominent in the forged image. This is unacceptable because, in real-life, different manipulation techniques can leave behind wide variety of forgery clues. Thus, focusing on specific forgery clues is not optimal. For example, if a method focuses on edge inconsistency to detect forgery, the method will not perform well on a forged image where the boundary between untampered and manipulated region is smooth. Again, if a method focuses on resampling features, it will struggle to detect forgery if an image has the same JPEG compression applied several times to both the untampered and manipulated regions.

Another major disadvantage of existing methods is that these methods use cross-entropy loss without additional constraints for training. Recently, [40] stated that traditional cross-entropy based methods assume that all instances within each category should be close in feature distribution. This ignores the unique information of each sample. Thus, cross-entropy loss encourages the model to extract similar features for same category. This might be helpful for classification or segmentation of datasets such as Imagenet or Cityscapes, where objects of the same category should have similar features. However, in the case of image forgery localization, extracting similar features for all the tampered regions in the dataset is not optimal as different manipulation operations leave behind different forgery foot-

prints in the tampered regions. Hence, without additional constraints, a common cross-entropy loss-based framework is prone to over-fitting on specific forgery patterns [28]. This is not conducive to generalization.

Taking all these limitations into consideration, we propose a novel forgery localization method named *Contrastive Forgery Localization Network* or *CFL-Net*, based on recently proposed contrastive loss [24]. Our method relies on the general assumption in underlying forged region localization that there remains a difference of feature statistics, i.e., color, intensity, noise, etc., between untampered region and manipulated region [22], irrespective of the forgery type. In this paper, we focus on leveraging this difference in the feature space to aid in image forgery localization via contrastive loss. Specifically, our model learns mapping into a feature space where the features between untampered and manipulated regions are well-separated and dispersed for each image. Thus, our method does not focus on specific forgery clues. Also, we calculate the contrastive loss for each sample. Hence, our method treats the forgery clues of each sample differently, which helps in generalization. Our main contributions are summarized as follows:

- We propose a novel image forgery localization method called *CFL-Net*. Our method leverages the difference of feature distribution between untampered and manipulated regions of each image sample and does not focus on specific forgery footprints. Hence, our method is more well-suited to detect real-life forgery.
- We address the problem of using cross-entropy loss without any constraints for general purpose image forgery localization. We incorporate contrastive loss and especially tailor it towards solving this problem.
- We perform extensive experiments on benchmark manipulation datasets to show that our method outperforms several existing image forgery localization methods.

## 2. Related Works

### 2.1. Image Forgery Localization

Image forgery methods are concerned with forgery classification or localization. Classification is basically predicting whether an image is forged or non-forged, whereas, forgery localization is concerned with locating the forged region as well. The latter is a segmentation task.

In pre deep learning era, methods used hand-crafted features such as local noise analysis [16, 10], CFA artifacts [15], JPEG compression [5] etc. Recent works usually use deep learning based methods in conjunction with these forgery traces to localize forged regions. Bappy et al. [2] exploit the edge inconsistency trace using LSTM to localize forgery. The work is later improved in [3], where the

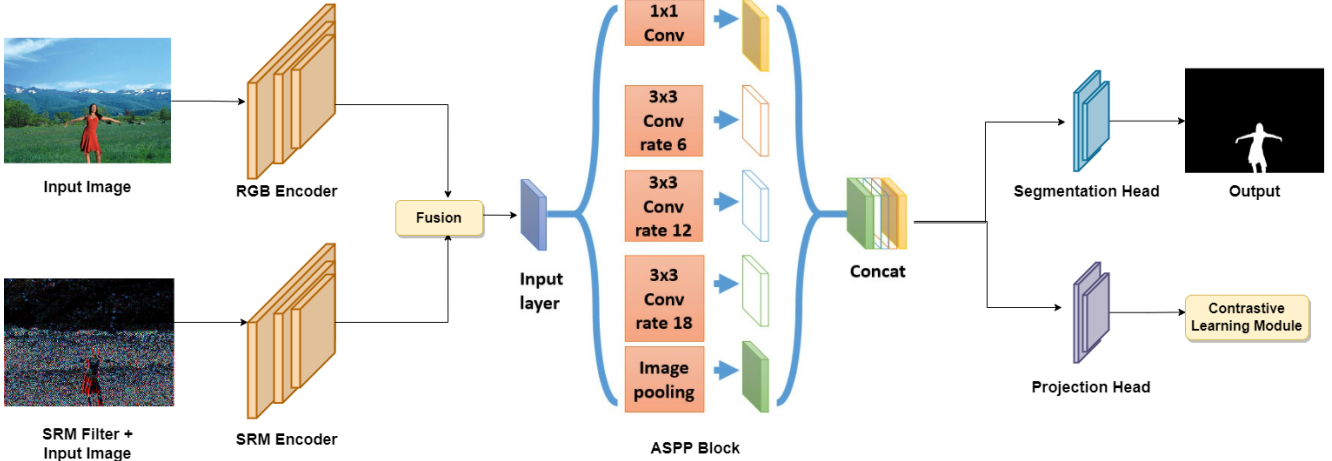


Figure 2: Overall architecture of the proposed CFL-Net. We use a two stream encoder, one for the RGB input image and the other for the SRM filtered image. The features produced by the encoders are fused and passed into the ASPP module. The output features from ASPP block then go through both the Segmentation Head and Projection Head, where the first produces the final prediction mask and the latter produces features that go into the contrastive learning module.

authors further exploit resampling traces using Laplacian filters. They also use a separate encoder-decoder structure to refine the predicted mask. RGB-N [41] proposes a two stream faster R-CNN network, one for the RGB image and other for the noise information traces generated by the Steganalysis Rich Model (SRM) filters [16]. SRM filters are high pass filters that enhance the high-frequency information, which becomes helpful in forgery localization. However, due to the R-CNN architecture, RGB-N is limited to localizing to a rectangular box whereas real objects are not necessarily rectangular. Mantra-Net [37] jointly detects and localizes forged images. ManTra-Net is composed of a VGG based feature extractor and an LSTM based detection module. The feature extractor is trained to detect various types of image manipulation traces. SPAN [22] proposes Spatial Pyramid Attention Network models the relationship between image patches at multiple scales by constructing a pyramid of local self-attention blocks. CAT-Net [26] uses two stream network similar to RGB-N, one for the RGB pixel stream and the other for DCT co-efficients. DCT helps to extract resampling trace features.

## 2.2. Contrastive Learning

Recently, contrastive learning [19, 8] has achieved great progress in unsupervised learning problem. SimCLR [8] proposes a simple framework to perform contrastive learning, where positive pairs are generated with two random augmented views of the same image and negative ones are obtained with different images, forming an image-level discrimination task. Furthermore, MoCo [19] maintains a queue of negative samples and turns one branch of Siamese network into a momentum encoder to improve consistency

of the queue. Recently [24] has extended unsupervised contrastive learning to fully-supervised setting that can effectively leverage label information. This setting has been used in semantic segmentation to improve the state-of-the-art performance. [34, 21] contrast the pixel embedding between different semantic categories in a supervised manner to aid in segmentation.

Sun et al. [33] have also used supervised contrastive loss to supplement cross-entropy loss for forgery detection task. However, their work is targeted toward forged face image classification. In contrast, our method is aimed toward general-purpose image forgery localization, which is a segmentation task. Also, the formation of our contrastive loss is different. Fung et al. [17] use unsupervised contrastive learning for deepfake face image forgery detection. This method is also aimed toward only forgery classification.

## 3. CFL-Net

In this section, we first describe the overall framework of our model. We then detail on the contrastive learning part.

### 3.1. Overall Framework

We present here the overall framework of our method. The overall diagram is shown in Figure 2. We opt for a two stream network similar to [41, 26, 33]. One stream takes the input RGB image  $I \in R^{3 \times H \times W}$  as input. We use SRM filters [16] to the RGB image and use that as an input for the other stream. SRM filters are high pass filters that enhance the high-frequency information of input image, thus highlighting the edge information more, which is helpful for localizing forgeries. We use ResNet [20] as the backbone for

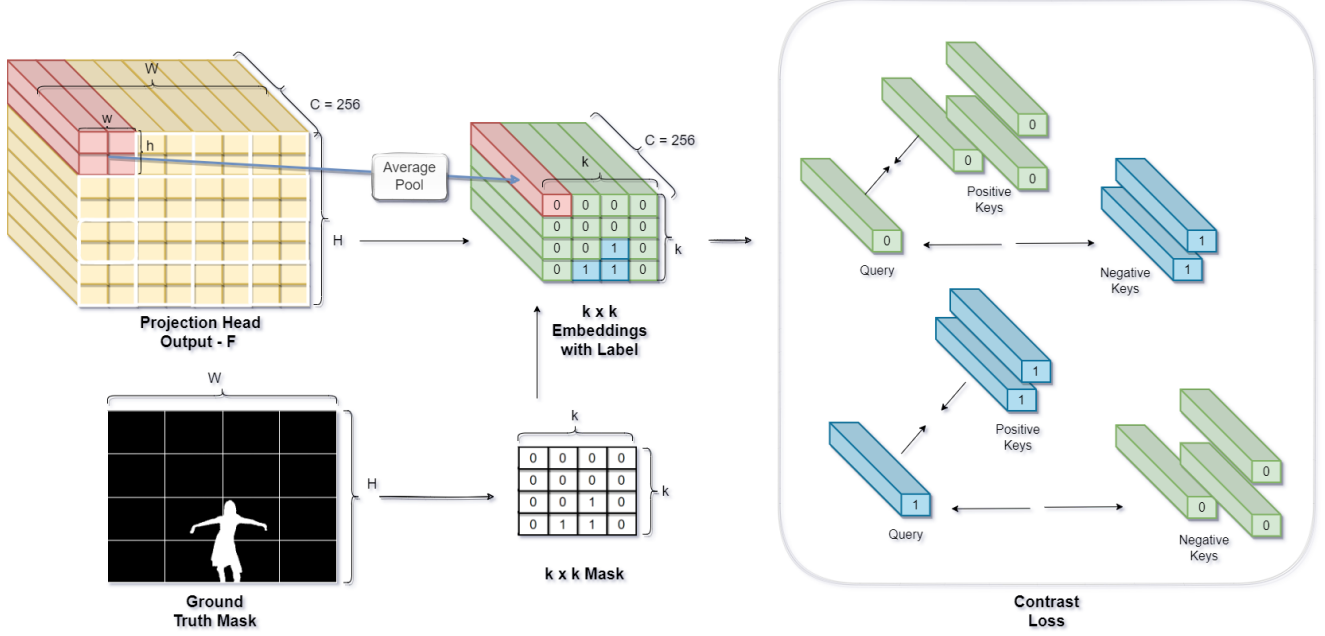


Figure 3: Contrastive Learning Module: For ease of visualization, the projection head in the figure is shown to output a feature map  $F$  of shape  $256 \times 8 \times 8$ . The feature map is then divided into  $4 \times 4$  patches. Then, all the 4 spatial vectors in each patch are averaged to get the embeddings of size  $4 \times 4$  (denoted as ' $k \times k$  Embeddings with Label' in figure). The ground truth mask is also divided into  $4 \times 4$  patches and maximum occurring pixel label in each patch is counted to get the output  $4 \times 4$  mask (denoted as ' $k \times k$  Mask' in figure). Eqn. (2) is then used to calculate the contrastive loss for each pixel embedding of the ' $k \times k$  Embeddings with Label'.

both the streams. We then fuse features from both streams by concatenating features channel-wise. ASPP module [7] is used on the fused feature map so that multi-scale information can be extracted. It is reported in [41] that global context helps to collect more clues, such as contrast difference, etc., for manipulation detection. ASPP module helps in this regard by extracting information in different scales, such that global context as well as more fine-grained pixel level context information becomes available.

We then use a segmentation head/decoder head and a projection head that takes the upsampled multi-scale feature extracted by the ASPP module as input. We opt for a DeepLab style segmentation head which outputs the final segmentation map of size  $H \times W$ . The projection map is composed of Conv-BatchNorm-Conv layer that projects the feature map to  $F \in R^{256 \times H \times W}$ , 256 being the embedding dimension. The embedded feature map  $F$  is passed on to the contrastive learning module. The projection head is not used during evaluation.

### 3.2. Contrastive Learning Module

Our goal is to contrast between the untampered and manipulated pixel embeddings of each sample so that the feature distributions between both regions get well-separated. As our embedded feature map is of size  $H \times W$  spatially and

we have the corresponding ground-truth mask  $M$  of similar size, we know the label of each pixel embedding. Thus, we can use supervised contrastive learning. For each query pixel embedding  $z_i$  the contrastive loss for that embedding becomes:

$$L_i = \frac{1}{|A_i|} \sum_{k^+ \in A_i} -\log \frac{\exp(z_i \cdot k^+ / \tau)}{\exp(z_i \cdot k^+ / \tau) + \sum_{k^-} \exp(z_i \cdot k^- / \tau)} \quad (1)$$

Here,  $k^+$  or positive key is a pixel embedding that has the same label as query  $z_i$ .  $A_i$  denotes the set of all  $k^+$  in the projection head output feature map  $F$ . Similarly,  $k^-$  or negative key, are pixel embeddings in  $F$  that have a different label than  $z_i$ .

However, calculating  $L_i$  in such a manner has some major limitations. First, calculating the contrastive loss based on single pixel embedding does not take into account the context information that the neighboring embeddings have. Also, to calculate the loss, a dot-product matrix of size  $HW \times HW$  needs to be stored, which is memory-consuming.

One possible solution is to randomly sample a few pixel embeddings from  $F$  corresponding to the two different

classes similar to [34]. Then, use those embeddings to calculate (1). This way, the memory requirement is greatly reduced. However, this solution does not take into account the context information from neighboring pixels. Also, similar to [21], another solution could be to average all the pixel embeddings of the two regions and then use the mean embeddings to calculate the loss. Although this may be helpful for computer vision tasks, such as segmentation of semantic objects etc., it is inappropriate for image manipulation detection tasks. Because, recent studies have shown that pooling is undesirable for tasks that require subtle signals since pooling reinforces content and suppresses noise-like signals [6]. These fine-grained traces are helpful for detecting forgery. Hence, to find a balance between context and fine-grained traces, we opt for dividing  $F$  into local regions.

We first partition  $F$  spatially into  $k \times k$  patches, thus getting  $f_i \in R^{256 \times h \times w}$ , where  $i \in \{1, 2, 3 \dots k^2\}$  and  $h = \frac{H}{k}$  and  $w = \frac{W}{k}$ . We then take the average of the pixel embeddings in each local region. Thus making each  $f_i$  to a shape of  $R^{256}$ . In a similar manner, we divide the ground truth mask  $M$  into  $k \times k$  patches.  $M$  has value of 0 in the untampered region and value of 1 in the forged region. We get  $m_i \in R^{h \times w}$ , where  $i \in \{1, 2, 3 \dots k^2\}$  and  $h = \frac{H}{k}$  and  $w = \frac{W}{k}$ . To get the value of the label of each  $m_i$ , we count the number of 0s and 1s in the  $h \times w$  patch. We then assign the value of  $m_i$  as the maximum count of value occurring in the patch.

Now, we have pixel embeddings  $f_i$  and corresponding label of each embedding  $m_i$ . We can now use the supervised contrastive loss as:

$$L_i = \frac{1}{|A_i|} \sum_{k^+ \in A_i} -\log \frac{\exp(f_i \cdot k^+ / \tau)}{\exp(f_i \cdot k^+ / \tau) + \sum_{k^-} \exp(f_i \cdot k^- / \tau)} \quad (2)$$

Here also,  $A_i$  denotes the set of all other pixel embeddings  $k^+$  that have the same label as  $f_i$ . Similarly,  $k^-$  are all the negative pixel embeddings that have different label than  $f_i$ . All the embeddings in the loss function are  $L_2$  normalized. For a single image sample, we get the final contrastive loss by averaging over all the embeddings:

$$L_{CON} = \frac{1}{k^2} \sum_{i \in k^2} L_i$$

Our final loss to optimize then becomes:

$$L = L_{CE} + L_{CON}$$

Here,  $L_{CE}$  is the cross-entropy loss.

## 4. Experiments

In this section, we describe experiments on three different manipulation datasets to explore the effectiveness of CFL-Net. These datasets are general manipulation datasets containing several manipulation types and are not specific to only a single manipulation type. The evaluation metric we use is pixel-wise Area Under Curve (AUC) score [22].

### 4.1. Datasets

- **IMD-20 [30]** is a real-life manipulation dataset made by unknown people and collected from the Internet. Hence, this dataset contains various types of manipulations. There are a total of 2010 image samples in the dataset.
- **CASIA [14]** CASIAv2 contains 5123 images and CASIAv1 contains 921 images. Samples from this dataset are manipulated by splicing and copy-move forgery. Also, image enhancement techniques including filtering and blurring are applied to the samples for post-processing.
- **NIST-16 [29]** contains 584 image samples with ground-truth masks. Samples from NIST16 are manipulated by splicing, copy-move and removal, and are post-processed to hide visible traces.

For each dataset, we use the same procedure as [18] for train-val-test splits. It should be noted that, previous methods such as, [22, 3, 26] usually pre-train their models on large ( $\approx 1M$  samples) synthetic manipulation datasets and then fine-tune the models on the datasets mentioned above to report the final result. However, in this paper, to evaluate solely the model's performance, we do not create a synthetic manipulation dataset to pretrain our model. Interestingly, without taking a resort to any large synthetic manipulation dataset, our model outperforms the baseline models.

### 4.2. Implementation Details

We use ResNet-50 as encoder for both the streams. We train CFL-Net with Adam optimizer with a learning rate of  $1e-4$ . We reduce the learning rate by 20% after each 20 epochs. We resize the input images to  $256 \times 256$ . We divide  $F$  into a total of  $64 \times 64$  patches. The temperature  $\tau$  of (2) is set as 0.1. Cross-entropy loss is weighted to give the tampered class ten times more weight. We set the batch size to 4 and train the model on NVIDIA RTX Titan GPU over 100 epochs.

### 4.3. Baseline Models

We compare our method with various baseline models, which are described below:

Methods	NIST	CASIA	IMD-20
J-LSTM (ICCV'17)	-	-	48.7
RGB-N (CVPR'18)	93.7	79.5	-
Mantranet (CVPR'19)	79.5	81.7	81.3
SPAN (ECCV'20)	96.1	83.8	-
Transforensics (ICCV'21)	-	85.0	84.8
Ours	<b>99.7</b>	<b>86.3</b>	<b>89.9</b>

Table 1: AUC Scores (in %).

- J-LSTM [2] employs a hybrid CNN-LSTM architecture to capture the discriminative features between untampered and tampered regions.
- RGB-N [41] adopts a two stream parallel network to separately discover tampering features.
- ManTraNet [37] uses a feature extractor to capture the manipulation traces and a local anomaly detection network to localize the manipulated regions.
- SPAN [22] uses a pyramid architecture to and self-attention blocks to model the dependency of image patches.
- Transforensics [18] uses vision transformers with dense self-attention encoders and dense correction modules to model all pairwise interactions between local patches at different scales.

## 5. Results

In this section we report the results of our experiments. We divide the result section into two subsections in order to show the quantitative and qualitative results separately. We also perform ablation study.

### 5.1. Quantitative Analysis

We report the AUC scores (in %) of our method and the baseline models in Table 1. It should be noted that the results of RGB-N and SPAN stated here are the fine-tuned results as reported in their respective papers. J-LSTM and Transforensics do not perform any pre-training. Although ManTraNet pre-trains their model on synthetic manipulation dataset, they do not fine-tune on specific dataset. Looking at the table, it can be seen that CFL-Net achieves the best localization performance on all the datasets amongst the baseline models. Especially, CFL-Net outperforms all the baseline models by a big margin on IMD-20 dataset, which is a real-life manipulation dataset with various forgery types. Specifically, CFL-Net achieves an AUC score of 89.9% on IMD-20 dataset, which is a 5.1% improvement over the second most well-performing model - Transforensics. Hence, it validates our claim that CFL-Net

Datasets		NIST	CASIA	IMD-20
NIST	w/o	98.3	67.1	66.4
	w	<b>99.7</b>	<b>67.6</b>	<b>69.8</b>
CASIA	w/o	79.3	84.9	75.5
	w	<b>79.9</b>	<b>86.3</b>	<b>77.8</b>
IMD-20	w/o	74.37	74.1	85.2
	w	<b>91.8</b>	<b>75.6</b>	<b>89.9</b>

Table 2: The left-most column shows the datasets models are trained on. The later columns are the datasets where the models are evaluated on. 'w/o' - CFL-Net trained without contrastive loss, 'w' - CFL-Net trained with contrastive loss. Results are in % AUC.

is well-suited to localize real-life forgery. Our model also outperforms baseline models on the rest of the datasets - Casia and Nist. Moreover, it is worth pointing out that CFL-Net achieves these results without pre-training on synthetic manipulation data.

We argued that, in consequence of adding contrastive loss, our proposed model does not focus on specific forgery footprints but learns more generalized features. Hence, our model should generalize better across different manipulation datasets than the model trained without contrastive loss. For this reason, in our next experiment, to get an idea of how well our proposed method generalizes across datasets, we evaluate the models trained on one dataset and evaluate on the test sets of the remaining datasets.

Table 2 shows the results. It is evident that CFL-Net trained with contrastive loss performs very well in generalizing across datasets. In all the cases this model performs better than the model trained without the contrastive loss. When trained on IMD-20 and evaluated on the test set of NIST, our proposed model even outperforms the AUC score of ManTraNet. The most performance boosts are seen when trained on IMD-20 dataset. IMD-20 is the real-life image manipulation dataset and hence training on this dataset helps the model learn most generalizable features. Hence our proposed model trained on IMD-20 and evaluated on rest of the datasets yields the most performance improvement over the model trained without contrastive loss.

It should also be noted that both models trained on NIST and evaluated on the other datasets perform poorly because NIST has very few images, i.e., 584 images in the dataset. Hence, it is difficult to generalize to other datasets using NIST. Still, our proposed model managed to perform better than the model trained without contrastive loss.

### 5.2. Qualitative Analysis

Here we visualize a few of the predicted masks from the test sets. We also show the corresponding predicted mask of ManTraNet [37] for comparison against our CFL-Net. ManTraNet's implementation and the saved model are made



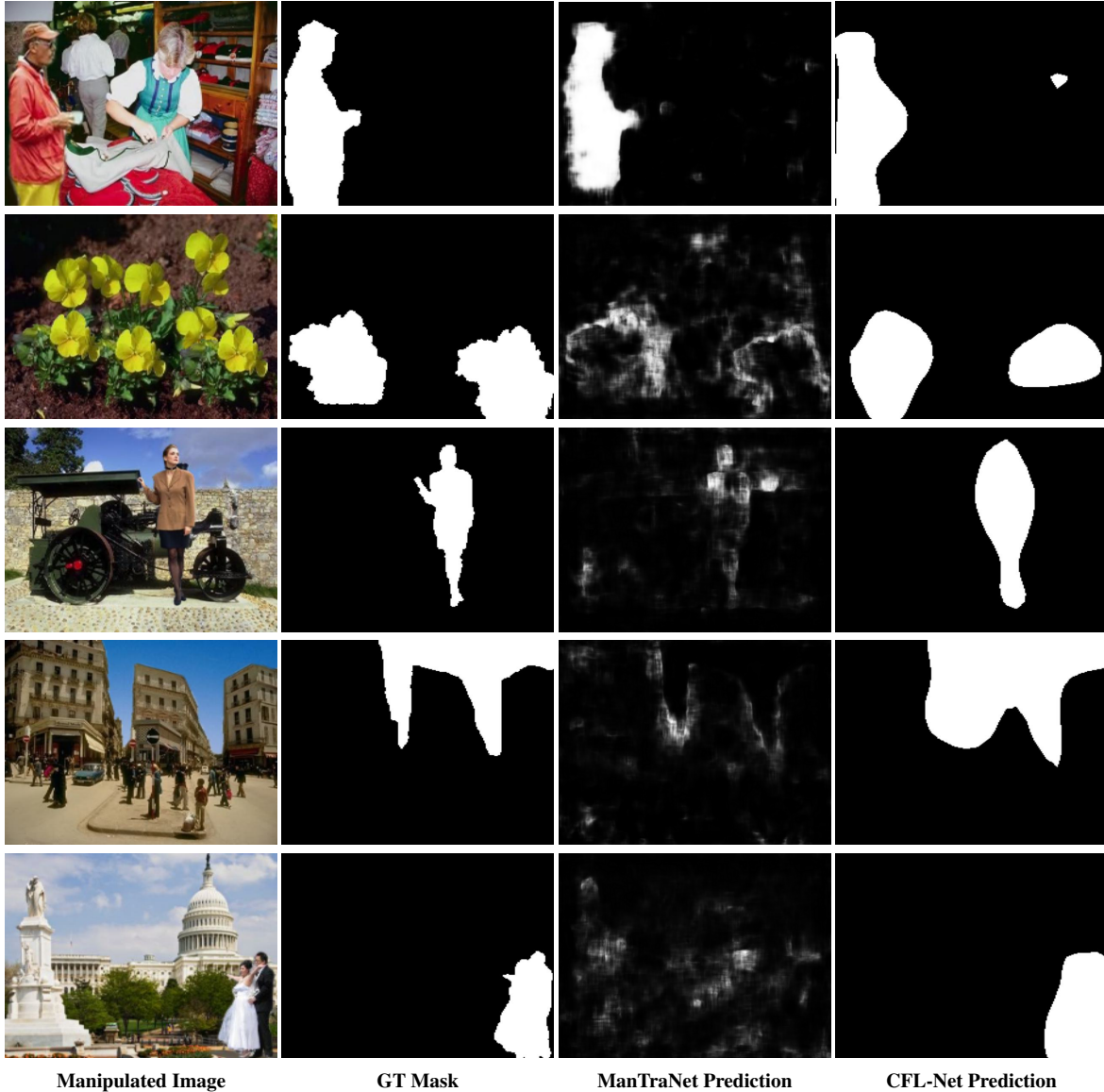


Figure 4: Comparison of the predicted mask with ManTraNet. It is evident that prediction of CFL-Net is closer to the GT mask compared to ManTraNet.

publicly available by the authors, which we employ here for the experiment. The results are shown in Figure 4. From the figure, it is evident that masks predicted by CFL-Net are closer to the ground truth masks. On the other hand, ManTraNet struggled to detect the manipulated region in most of the cases.

Next, in order to show that our contrastive loss preserves the feature variations by avoiding clustering of same class features, we visualize via t-SNE the class features obtained from the segmentation head in Figure 5. The left column shows the mean feature vectors per image sample on IMD-20 and CASIA test sets when CFL-Net is trained using

only cross-entropy loss. Visibly, the features corresponding to both untampered (green color in figure) and tampered (red color in figure) regions are congested here. On the other hand, the right column shows the mean features when CFL-Net is trained using both cross-entropy and contrastive loss. Here, the features corresponding to both regions are more dispersed. Hence, different manipulation footprints are more separable. This experiment demonstrates that the traditional cross-entropy loss reduces generalization in case of image forgery localization due to the intra-category invariance, while our proposed method can improve the generalization by diverging the feature distribution.

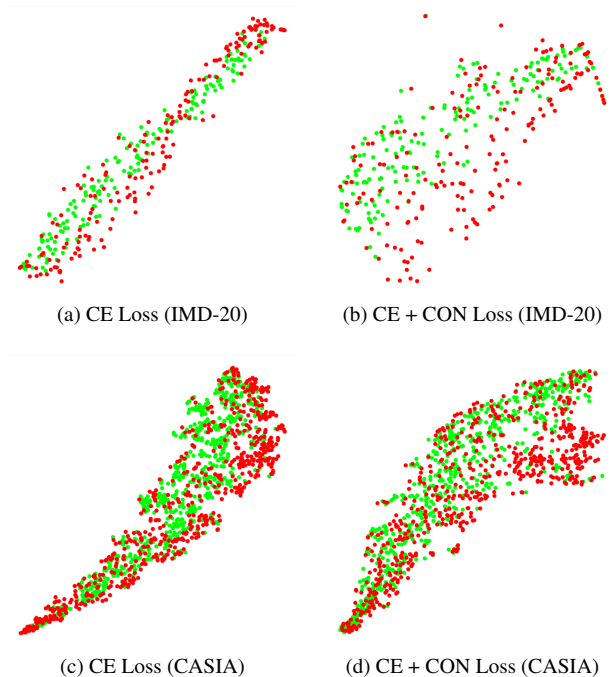


Figure 5: Left column shows t-SNE diagram of the mean features on IMD-20 and CASIA testsets when CFL-Net is trained using only cross-entropy loss. Right column corresponds to CFL-Net trained using both cross-entropy loss and contrastive loss. Green = Untampered feature, Red = Tampered feature.

### 5.3. Ablation Study

In this subsection, we conduct ablation experiments to study how the proposed loss of CFL-Net influence the localization performance. Specifically, we train CFL-Net without the contrastive loss and then report the results to get an idea of the influence of contrastive loss.

Methods	NIST	CASIA	IMD
CE Loss	98.3	84.9	85.2
CE + CON Loss	<b>99.7</b>	<b>86.3</b>	<b>89.9</b>

Table 3: AUC scores (in %) for CFL-Net trained with different loss settings. CE = Cross-entropy loss, CON = Contrastive Loss.

In Table 3 we report the results. It is clear from the table that adding the contrastive loss indeed helps in localization. The improvement is much more prominent on the real-life image manipulation dataset IMD-20. Contrastive loss helps to improve the AUC score by 4.7%. It should be noted that without the contrastive loss our method already achieves very good results. The reason is that our model

is similar to RGB-N [41] in the regard that we also use two stream network, i.e, RGB and SRM streams. In addition, we carefully supplement our network with ASPP module and Deeplab decoder head, which helps to improve the overall performance compared to RGB-N. Using contrastive loss further improves our results and helps to outperform all the other baseline models.

## 6. Conclusion

In this paper, we approached the general-purpose image forgery localization problem from a new perspective, i.e., using contrastive learning. We identified a major drawback of existing methods that focus on specific forgery footprints and use cross-entropy loss without any constraints to localize forgery. To address the drawbacks, we supplemented cross-entropy loss with contrastive loss and proposed a novel image forgery localization method named *Contrastive Forgery Localization Network* or *CFL-Net*. We conducted experiments on three benchmark image manipulation datasets and compared our results with major forgery localization methods of recent years. CFL-Net outperformed all the methods in terms of AUC metric. Moreover, the improvement is much more prominent on the real-life image manipulation dataset IMD-2020. Amongst the future works, a more sophisticated fusing mechanism can be considered to fuse the feature maps from the RGB and SRM streams. For example, attention modules or recently proposed vision transformers can be employed as a fusion mechanism.

## 7. Acknowledgment

This work was supported by ICT Division - Government of Bangladesh and Independent University Bangladesh (IUB). In addition, this work was partly supported by the Basic Science Research Program through National Research Foundation of Korea (NRF) grant funded by the Korean Ministry of Science and ICT (MSIT) under No. 2020R1C1C1006004 and Institute for Information & communication Technology Planning & evaluation (IITP) grants funded by the Korean MSIT: (No. 2022-0-01199, Graduate School of Convergence Security at Sungkyunkwan University), (No. 2022-0-01045, Self-directed Multi-Modal Intelligence for solving unknown, open domain problems), (No. 2022-0-00688, AI Platform to Fully Adapt and Reflect Privacy-Policy Changes), (No. 2021-0-02068, Artificial Intelligence Innovation Hub), (No. 2019-0-00421, AI Graduate School Support Program at Sungkyunkwan University), and (No. 2021-0-02309, Object Detection Research under Low Quality Video Condition).



## References

- [1] Irene Amerini, Tiberio Uricchio, Lamberto Ballan, and Roberto Caldelli. Localization of jpeg double compression through multi-domain convolutional neural networks. In *2017 IEEE Conference on computer vision and pattern recognition workshops (CVPRW)*, pages 1865–1871. IEEE, 2017.
- [2] Jawadul H Bappy, Amit K Roy-Chowdhury, Jason Bunk, Lakshmanan Nataraj, and BS Manjunath. Exploiting spatial structure for localizing manipulated image regions. In *Proceedings of the IEEE international conference on computer vision*, pages 4970–4979, 2017.
- [3] Jawadul H Bappy, Cody Simons, Lakshmanan Nataraj, BS Manjunath, and Amit K Roy-Chowdhury. Hybrid lstm and encoder–decoder architecture for detection of image forgeries. *IEEE Transactions on Image Processing*, 28(7):3286–3300, 2019.
- [4] Belhassen Bayar and Matthew C Stamm. Constrained convolutional neural networks: A new approach towards general purpose image manipulation detection. *IEEE Transactions on Information Forensics and Security*, 13(11):2691–2706, 2018.
- [5] Tiziano Bianchi, Alessia De Rosa, and Alessandro Piva. Improved dct coefficient analysis for forgery localization in jpeg images. In *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2444–2447. IEEE, 2011.
- [6] Mehdi Boroumand, Mo Chen, and Jessica Fridrich. Deep residual network for steganalysis of digital images. *IEEE Transactions on Information Forensics and Security*, 14(5):1181–1193, 2018.
- [7] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017.
- [8] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.
- [9] Hak-Yeol Choi, Han-Ul Jang, Dongkyu Kim, Jeongho Son, Seung-Min Mun, Sunghee Choi, and Heung-Kyu Lee. Detecting composite image manipulation based on deep neural networks. In *2017 International Conference on Systems, Signals and Image Processing (IWSSIP)*, pages 1–5. IEEE, 2017.
- [10] Davide Cozzolino, Diego Gragnaniello, and Luisa Verdoliva. Image forgery localization through the fusion of camera-based, feature-based and pixel-based techniques. In *2014 IEEE International Conference on Image Processing (ICIP)*, pages 5302–5306. IEEE, 2014.
- [11] Davide Cozzolino, Giovanni Poggi, and Luisa Verdoliva. Efficient dense-field copy–move forgery detection. *IEEE Transactions on Information Forensics and Security*, 10(11):2284–2297, 2015.
- [12] Davide Cozzolino, Giovanni Poggi, and Luisa Verdoliva. Spltcebuster: A new blind image splicing detector. In *2015 IEEE International Workshop on Information Forensics and Security (WIFS)*, pages 1–6. IEEE, 2015.
- [13] Davide Cozzolino and Luisa Verdoliva. Noiseprint: a cnn-based camera model fingerprint. *IEEE Transactions on Information Forensics and Security*, 15:144–159, 2019.
- [14] Jing Dong, Wei Wang, and Tieniu Tan. Casia image tampering detection evaluation database. In *2013 IEEE China Summit and International Conference on Signal and Information Processing*, pages 422–426. IEEE, 2013.
- [15] Pasquale Ferrara, Tiziano Bianchi, Alessia De Rosa, and Alessandro Piva. Image forgery localization via fine-grained analysis of cfa artifacts. *IEEE Transactions on Information Forensics and Security*, 7(5):1566–1577, 2012.
- [16] Jessica Fridrich and Jan Kodovsky. Rich models for steganalysis of digital images. *IEEE Transactions on information Forensics and Security*, 7(3):868–882, 2012.
- [17] Sheldon Fung, Xuequan Lu, Chao Zhang, and Chang-Tsun Li. Deepfakeucl: Deepfake detection via unsupervised contrastive learning. In *2021 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2021.
- [18] Jing Hao, Zhixin Zhang, Shicai Yang, Di Xie, and Shiliang Pu. Transforensics: image forgery localization with dense self-attention. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15055–15064, 2021.
- [19] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020.
- [20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
- [21] Hanzhe Hu, Jinshi Cui, and Liwei Wang. Region-aware contrastive learning for semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16291–16301, 2021.
- [22] Xuefeng Hu, Zhihan Zhang, Zhenye Jiang, Syomantak Chaudhuri, Zhenheng Yang, and Ram Nevatia. Span: Spatial pyramid attention network for image manipulation localization. In *European conference on computer vision*, pages 312–328. Springer, 2020.
- [23] Minyoung Huh, Andrew Liu, Andrew Owens, and Alexei A Efros. Fighting fake news: Image splice detection via learned self-consistency. In *Proceedings of the European conference on computer vision (ECCV)*, pages 101–117, 2018.
- [24] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *Advances in Neural Information Processing Systems*, 33:18661–18673, 2020.
- [25] Vladimir V Kniaz, Vladimir Knyaz, and Fabio Remondino. The point where reality meets fantasy: Mixed adversarial generators for image splice detection. *Advances in Neural Information Processing Systems*, 32, 2019.

- [26] Myung-Joon Kwon, In-Jae Yu, Seung-Hun Nam, and Heung-Kyu Lee. 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*, pages 375–384, 2021.
- [27] Ce Li, Qiang Ma, Limei Xiao, Ming Li, and Aihua Zhang. Image splicing detection based on markov features in qdct domain. *Neurocomputing*, 228:29–36, 2017.
- [28] Yuchen Luo, Yong Zhang, Junchi Yan, and Wei Liu. Generalizing face forgery detection with high-frequency features. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16317–16326, 2021.
- [29] NIST: Nist nimble 2016 datasets (2016). <https://www.nist.gov/itl/iad/mig>.
- [30] Adam Novozamsky, Babak Mahdian, and Stanislav Saic. Imd2020: a large-scale annotated dataset tailored for detecting manipulated images. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision Workshops*, pages 71–80, 2020.
- [31] Abdul Muntakim Rafi, Jonathan Wu, Md Hasan, et al. L2-constrained remnet for camera model identification and image manipulation detection. In *European Conference on Computer Vision*, pages 267–282. Springer, 2020.
- [32] Ronald Salloum, Yuzhuo Ren, and C-C Jay Kuo. Image splicing localization using a multi-task fully convolutional network (mfcn). *Journal of Visual Communication and Image Representation*, 51:201–209, 2018.
- [33] Ke Sun, Taiping Yao, Shen Chen, Shouhong Ding, Rongrong Ji, et al. Dual contrastive learning for general face forgery detection. *arXiv preprint arXiv:2112.13522*, 2021.
- [34] Wenguan Wang, Tianfei Zhou, Fisher Yu, Jifeng Dai, Ender Konukoglu, and Luc Van Gool. Exploring cross-image pixel contrast for semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7303–7313, 2021.
- [35] Yue Wu, Wael Abd-Almageed, and Prem Natarajan. Buster-net: Detecting copy-move image forgery with source/target localization. In *Proceedings of the European conference on computer vision (ECCV)*, pages 168–184, 2018.
- [36] Yue Wu, Wael Abd-Almageed, and Prem Natarajan. Image copy-move forgery detection via an end-to-end deep neural network. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1907–1915. IEEE, 2018.
- [37] Yue Wu, Wael AbdAlmageed, and Premkumar Natarajan. Mantra-net: Manipulation tracing network for detection and localization of image forgeries with anomalous features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9543–9552, 2019.
- [38] Qiuwei Yang, Fei Peng, Jiao-Ting Li, and Min Long. Image tamper detection based on noise estimation and lacunarity texture. *Multimedia Tools and Applications*, 75(17):10201–10211, 2016.
- [39] Zhongping Zhang, Yixuan Zhang, Zheng Zhou, and Jiebo Luo. Boundary-based image forgery detection by fast shallow cnn. In *2018 24th International Conference on Pattern Recognition (ICPR)*, pages 2658–2663. IEEE, 2018.
- [40] Nanxuan Zhao, Zhirong Wu, Rynson WH Lau, and Stephen Lin. What makes instance discrimination good for transfer learning? *arXiv preprint arXiv:2006.06606*, 2020.
- [41] Peng Zhou, Xintong Han, Vlad I Morariu, and Larry S Davis. Learning rich features for image manipulation detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1053–1061, 2018.
- [42] Xinshan Zhu, Yongjun Qian, Xianfeng Zhao, Biao Sun, and Ya Sun. A deep learning approach to patch-based image inpainting forensics. *Signal Processing: Image Communication*, 67:90–99, 2018.