DF-Net: The Digital Forensics Network for Image Forgery Detection

David Fischinger and Martin Boyer

Austrian Institute of Technology

Abstract

The orchestrated manipulation of public opinion, particularly through manipulated images, often spread via online social networks (OSN), has become a serious threat to society. In this paper we introduce the Digital Forensics Net (DF-Net), a deep neural network for pixel-wise image forgery detection. The released model outperforms several state-of-the-art methods on four established benchmark datasets. Most notably, DF-Net's detection is robust against lossy image operations (*e.g* resizing, compression) as they are automatically performed by social networks.

Keywords: Image Manipulation Detection and Localization, Digital Forensics, DF-Net

1 Introduction

"Fake News" poses an ever-growing challenge in our society as technological advancements facilitate the production of high-quality forgeries in digital media like audio, video, and images. This impact ranges from satirical memes to orchestrated political Fake News campaigns that aim to manipulate public opinion. In this paper, we introduce an effective approach to identify manipulated regions in images. This enables institutions such as media organizations and interested citizens to get a better indication of whether specific images may have been manipulated.

Over the past decade, various methods have been proposed to detect different categories of image forgery, including: copy-move, splicing, inpainting, and various enhancement techniques. However, these approaches often concentrate on specific features of each manipulation type. In recent years, more general approaches for multiple manipulation types were developed, such as [Wu et al., 2019] and [Wu et al., 2022]. Each of them promotes sophisticated and problem-specific network architectures and concepts, like modeling known and unknown noise on images that result from transmission to Online Social Networks (OSN).

In this paper we present the DF-Net, an image forgery detector trained on the DF2023 dataset [Fischinger and Boyer, 2023]. To be more specific, our main contributions are as follows:

- **Model:** Our proposed forgery detection model outperforms several state-of-the-art methods. We show its evaluation on four benchmark datasets. The model's deep learning network architecture combines the strengths of two specialized sub-models, trained from scratch on our DF2023 dataset.
- **OSN robustness:** We show the robustness of our model against lossy operations (e.g. resizing, compression) as automatically done by online social networks in an extensive evaluation.
- **Speed:** We present a processing time comparison with a SOTA approach that shows a significant reduction in time, especially for larger images.

2 Related Work

Many methods of detecting and localizing image forgery were published (see, for example, the review of [Verdoliva, 2020] and references therein) in order to ensure visual information authenticity. Some of these forensic techniques are designed to detect specific forms of tampering, such as splicing [Lyu et al., 2013],

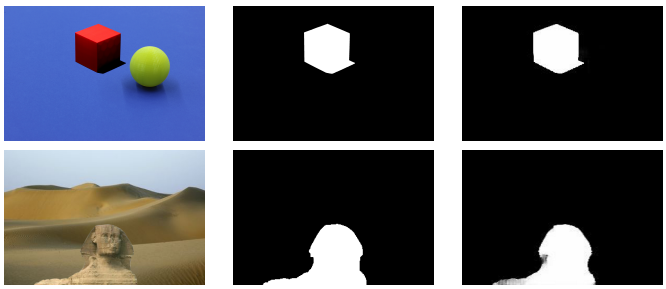

Figure 1: Forgery detection results of our network. Example images are taken from the CASIA [Dong et al., 2013] and the NIST [National Institute of Standards and Technology (NIST), 2016] datasets.

copy-move [Wang et al., 2017, Mahmood et al., 2017, Ouyang et al., 2019, Zedan et al., 2021, Zhong and Pun, 2020], and inpainting [Li et al., 2017]. Unfortunately, these forensic approaches can only be applied to detect specific tampering manipulations.

In recent years, deep learning-based methods were developed to address the problem of detecting general (compound) types of forgeries. Notably, [Wu et al., 2019] proposes a unified deep neural architecture called ManTra-Net, which is an end-to-end network that performs both detection and localization without extra preprocessing and postprocessing. ManTra-Net is a fully convolutional network which can handle images of arbitrary sizes and many known – and even unkown – forgery types. Furthermore, the authors design a self-supervised learning task to learn robust image manipulation features, formulate the forgery localization problem as a local anomaly detection problem, and propose a long short-term memory (LSTM) solution to assess local anomalies.

The work of [Zhuang et al., 2021] addresses the issue of tampering localization by focusing on the detection of commonly used editing tools and operations in Photoshop. A fully convolutional encoder-decoder architecture is designed, as well as a training data generation strategy by resorting to Photoshop scripting.

The widespread availability of online social networks (OSN), *e.g.*, Twitter, Facebook, Whatsapp, etc., makes them the dominant channels for transmitting forged images. However, almost all OSN manipulate the uploaded images in a lossy fashion (including format conversion, resizing, enhancement filtering and JPEG compression). The noise introduced by these lossy operations could severely affect the effectiveness of forensic methods. In a recent paper by [Wu et al., 2022], the problem of OSN-shared image forgeries is tackled by employing a dedicated training scheme. A baseline detector is presented, which is based on a modified U-Net [Ronneberger et al., 2015] as the backbone architecture. Next, an analysis of the noise introduced by OSN is conducted, and the noise is decoupled into two parts, i.e., predictable noise and unseen noise. These are then modelled separately and the modelled noise is further incorporated into the training framework.

Outline: The rest of this paper is structured as follows: In section 3, we present the DF-Net, evaluate different model design choices and investigate combinations of the models. In section 4, our proposed model is evaluated and compared to state-of-the-art methods, specifically for OSN transmitted images. Final remarks are made in section 5.

3 Network Architecture

3.1 Architecture

The DF-Net (available from *https://zenodo.org/record/8142658*) is designed to detect and localize image forgeries of various types. Essentially, this is a binary segmentation problem in which each pixel of an image is classified as either pristine or forged, resulting in a binary mask M.

Our proposed network comprises two sub-networks (M1, M2). Both networks use U-Net [Ronneberger et al., 2015] implementations, an architecture commonly used in the area of image segmentation.

The U-Net architecture of M1 is depicted in Fig. 2: U-Nets are Convolutional Neural Networks (CNNs) which consist of an encoding part where the spatial dimensions are downscaled (downsampling), and a decoding part where the spatial dimensions are increased (upsampling). On the first four sampling stages, skip connections are used which provide multi-channel feature maps from the encoding part directly to the decoding stage with the same spatial dimensions. Our U-Net implementation takes RGB images of size (256,256,3) as input. In each scaling step, we use two times a building block consisting of a 3x3 convolution, a batch normalization layer and a Relu activation layer, followed by a spatial and channel Squeeze & Excitation (scSE) block [Roy et al., 2018] and a (2x2) Max-pooling (downscaling) or a (3x3) Conv2DTranspose layer (upscaling). In the upscaling phase, the skip features are concatenated with the output of the Conv2DTranspose layer. The scSE layer can be seen as a re-calibration method for the network with a relatively small overhead regarding computing resources. During the training process, these blocks amplify spatial areas and channels which contribute more to better solutions and diminish the influence of worse performing network parts. At the final layer of the network, a 1x1 convolution with sigmoid activation function is used to calculate a value in $[0, 1]$ which indicates how likely each pixel is manipulated.

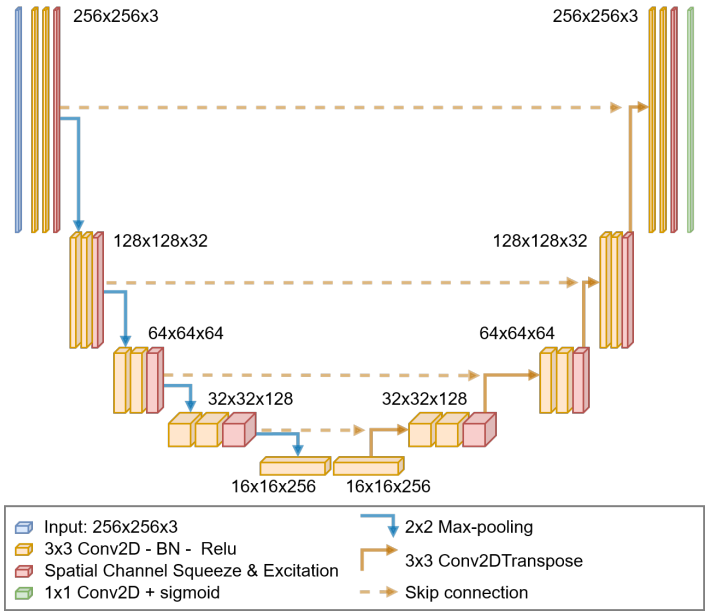

Figure 2: Network architecture of submodel M1: A U-Net architecture with 4 skip connections and spatial channel Squeeze & Excitation (scSE) extension. A more detailed description can be found in section 3.1.

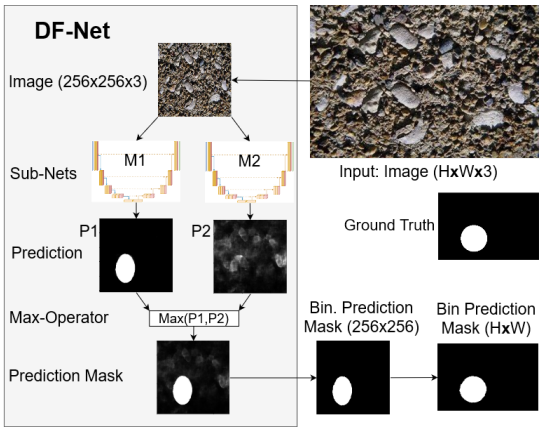

Figure 3: Network architecture of DF-Net: Example of Model combination for image Sp_S_NNN_C_txt0019_txt0019_0019.jpg from the CASIA_V1 dataset.

The architecture of model M2 slightly deviates from M1: for each Conv2D block of M2, the kernel size is set to 5x5 instead of 3x3 for exactly 4 filters. The output feature maps of the filters with different-sized kernels are concatenated again before the batch normalization is done.

We listed major evolutionary steps of the DF-Net in Tab. 1, where we compared networks on four benchmark datasets. Considerable performance boosts were accomplished when adding the spatial channel Squeeze & Excitation (scSE) calibration blocks (plus of 0.082) and after the input size for training and prediction was increased from ($224x224$) to ($256x256$) (0.029). Model M2 was only trained on splicing images from DF2023, but still performed best out of all single sub-networks. The architectural modification replacing four $3x3$ kernels with $5x5$ kernels in each convolution gave an additional boost of 0.014. A considerable performance gain was reached by combining networks. With the maximum operator, better results were achieved compared to averaging the predictions of two single networks. Taking the maximum prediction value from M1 and M2 for each image pixel, resulted in an average value of 0.583 which exceeds the best performing sub-model $M2$, which only achieved 0.535 overall. A deeper analysis showed that the maximum-operator-based model performs

worse than the better sub-model, for most images. For the CASIA dataset, the AUC value for max(M1,M2) is higher than the better performing submodule for only 86 out of 920 images. But the AUC performance is considerably higher than the average AUC value of the two sub-modules. In other words: By applying the maximum operator, the models prediction on each image is generally almost as good as the better-performing sub-module. So, the final DF-Net architecture depicted in Fig. 3 combines the strengths of the separately trained models M1 and M2 by taking the maximum prediction value per pixel from the outputs of both sub-nets.

Model Arch.	Average(metrics)				Test Datasets											
					CASIA			Columbia			DSO			NIST		
	Mean	AUC	F1	IoU	AUC	F1	IoU	AUC	F1	IoU	AUC	F1	IoU	AUC	F1	IoU
U-Net (224^2)	**.390**	.701	.263	.206	.75	.21	.18	.80	.52	.41	.60	.10	.06	.65	.22	.17
M1 (224^2)	**.472**	.797	.337	.282	.85	.43	.37	.83	.50	.43	.73	.15	.11	.78	.26	.22
M1 (256^2)	**.501**	.782	.396	.326	.85	.53	.47	.79	.54	.45	.71	.24	.16	.77	.28	.23
M2: M1-Ar.	**.521**	.790	.425	.348	.82	.42	.36	.87	.70	.61	.74	.32	.22	.74	.26	.20
M2	**.535**	.804	.439	.363	.82	.43	.37	.89	.74	.67	.76	.31	.21	.75	.27	.20
avg(M1,M2)	**.57**4	.842	.474	.405	.91	.60	.54	.88	.69	.63	.78	.30	.21	.80	.30	.24
max(M1,M2)	**.583**	.837	.501	.411	.91	.59	.50	.88	.76	.68	.77	.36	.25	.79	.30	.23

Table 1: Comparison of different network implementations we have trained and evaluated. The first two networks were trained on images of size $224x224$, all other networks on images of size $256x256$. Column **Mean** shows the average of the AUC, F1 and IoU metrics over all 4 datasets. The last row shows results from the DF-Net. More details are given in Sec. 3.1

Our experiments show that the combination of separately trained sub-networks results in a considerable performance improvement (see Tab. 1). Equally beneficiary is the advantage of splitting the whole model to sub-models that can be trained separately. This reduces time because researchers can faster assess if changes in the network architecture, the training data or the training parameters should be discarded or pursued. Furthermore, it allows to overcome hardware limitations which we see as a major advantage of this architecture.

3.2 Implementation Details

Model specification and training were done in the deep learning framework Tensorflow. For training and detection, the images were resized to $(256, 256)$ pixels. An `Nvidia GeForce GTX 1080 Ti` GPU with 11G memory was used for training, with batch size set to 32. We used the Adam optimizer [Kingma and Ba, 2015] and performed 1000 steps per epoch and stopped after the loss value of the validation dataset did not improve for 35 epochs. Training starts with a learning rate of 0.0001, which is halved after 10 epochs without improvement. M1 was first trained on all manipulation types and images. Subsequently it was refined by training on the 200.000 copy-move forgeries from DF2023 [Fischinger and Boyer, 2023]. Model M2 was trained only on the 400K splicing images from DF2023.

4 Evaluation

We compare the DF-Net to four state-of-the-art methods: ForSim [Mayer and Stamm, 2019], DFCN [Zhuang et al., 2021], ManTra-Net [Wu et al., 2019] and the work of Wu *et al.* [Wu et al., 2022], abbreviated below as Wu22. The approaches are evaluated on the four benchmark datasets `CASIA_V1` [Dong et al., 2013], `Columbia` [Hsu and Chang, 2006], `DSO` [Carvalho et al., 2013] and `NIST16` [National Institute of Standards and Technology (NIST), 2016]. See Table 2 for an overview of the datsets.

4.1 Online Social Networks

The popularity of online social networks (OSN) makes them the dominating channels for the distribution of manipulated images in the context of entertainment, but also for fake news, disinformation and propaganda.

Dataset	# Images	Format	t-WU22	t-DF-N
CASIA [Dong et al., 2013]	920	jpg	169	**155**
Columbia [Hsu and Chang, 2006]	160	tif	120	**28**
DSO [Carvalho et al., 2013]	100	png	701	**27**
NIST [National Institute of Standards and Technology (NIST), 2016]	564	jpg	15250	**235**

Table 2: Benchmark datasets with processing times (t) for Wu22 [Wu et al., 2022] and DF-Net (ours) in seconds for predictions per benchmark dataset. For datasets with huge images such as NIST (images of size up-to 5616×3744 pixels), tile-based approaches take considerably longer than approaches performing pre-scaling.

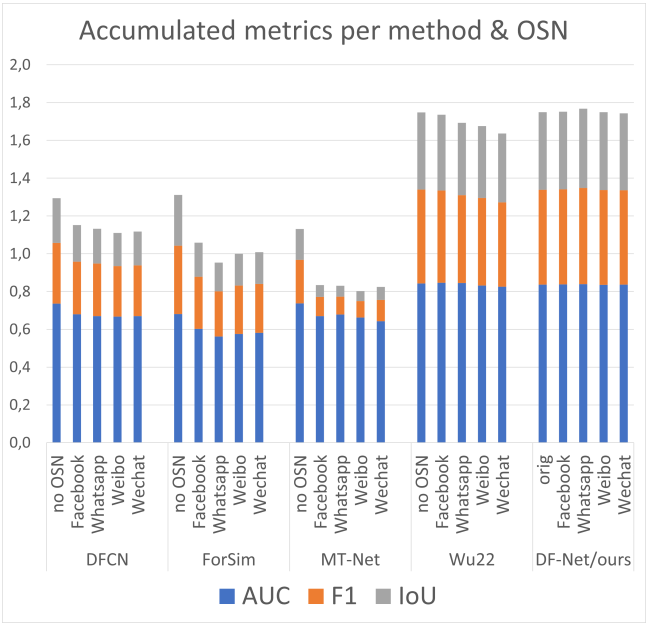

Figure 4: Metrics (AUC, F1, IoU) averaged over 4 benchmark datasets in accumulated presentation. Each column represents the combination of a method and the OSN used for dataset modification

Unfortunately, OSN automatically apply operations like compression and resizing, which reduce valuable information for image forgery detection. To show the robustness of DF-Net against these lossy operations, all the SOTA methods are tested against OSN adapted versions of the four benchmark datasets. In [Wu et al., 2022], the authors transmitted the images of the benchmark datasets via the social online platforms Facebook, Whatsapp, Weibo, and Wechat and made the collected datasets and their evaluation of several state-of-the-art methods available to the research community. We could reproduce the results for Wu22 [Wu et al., 2022]. For ManTra-Net [Wu et al., 2019], we tested the officially released TensorFlow model which is different from the model used for the authors' evaluation as the authors stated in [Wu, 2023], and a public PyTorch re-implementation [Abecidan, 2023] as well. Here we could reproduce the ManTraNet results as stated in [Wu et al., 2022]. For ForSim [Mayer and Stamm, 2019] and DFCN [Zhuang et al., 2021] results from the evaluation in [Wu et al., 2022] are stated in Tab. 3, together with the evaluation of the proposed DF-Net.

4.1.1 Evaluation Criteria:

We adopt three metrics commonly used in the area of image forgery detection: Area under the receiver operating characteristic curve (AUC), F1-score and Intersection over Union (IoU). The metrics are calculated on a pixel-level. For IoU and F1-score, the threshold for the output of the trained networks is set to 0.5.

Models	OSN	Test Datasets															
		CASIA			Columbia			DSO			NIST			Average			
		AUC	F1	IoU	AUC	F1	IoU	AUC	F1	IoU	AUC	F1	IoU	AUC	F1	IoU	Mean
DFCN	-	.654	.192	.119	.789	.541	.395	.724	.303	.227	.778	.250	.204	.736	.322	.236	.431
FSim	-	.554	.169	.102	.731	.604	.474	.796	**.487**	**.371**	.642	.188	.123	.681	.362	.268	.437
MNet	-	.776	.130	.086	.747	.357	.258	.795	.344	.253	.634	.088	.054	.738	.230	.163	.377
Wu22	-	.873	.509	.465	.862	.707	.608	**.854**	.436	.308	.783	**.332**	**.255**	**.843**	.496	.409	.5827
DF-Net	-	**.906**	**.589**	**.496**	**.880**	**.757**	**.679**	.769	.360	.246	**.793**	.299	.226	.837	**.501**	**.411**	**.5832**
DFCN	Facebook	.654	.190	.116	.687	.479	.338	.673	.238	.184	.705	.207	.138	.680	.278	.194	.384
FSim	Facebook	.537	.157	.094	.607	.450	.304	.689	.356	.238	.580	.140	.085	.603	.276	.180	.353
MNet	Facebook	.763	.102	.065	.626	.103	.056	.638	.109	.071	.652	.095	.057	.670	.102	.062	.278
Wu22	Facebook	.862	.462	.417	**.883**	.714	.611	**.859**	**.447**	**.320**	.783	**.329**	**.253**	**.847**	.488	.400	.578
DF-Net	Facebook	**.905**	**.587**	**.492**	.883	**.760**	**.681**	.770	.359	.245	**.795**	.304	.229	.838	**.502**	**.412**	**.584**
DFCN	Wechat	.651	.193	.119	.676	.487	.344	.653	.221	.137	.701	.176	.114	.670	.269	.179	.373
FSim	Wechat	.532	.153	.091	.650	.496	.354	.564	.247	.147	.581	.136	.082	.582	.258	.168	.336
MNet	Wechat	.724	.080	.048	.613	.199	.125	.582	.076	.045	.654	.095	.057	.643	.113	.069	.275
Wu22	Wechat	.833	.405	.358	**.883**	.727	.631	**.823**	**.366**	**.252**	.764	.286	.214	.826	.446	.364	.545
DF-Net	Wechat	**.902**	**.564**	**.467**	.881	**.759**	**.681**	.765	.358	.245	**.799**	**.314**	**.238**	**.837**	**.499**	**.408**	**.581**
DFCN	Whatsapp	.655	.191	.117	.692	.471	.331	.645	.264	.162	.689	.187	.125	.670	.278	.184	.377
FSim	Whatsapp	.525	.151	.091	.595	.436	.294	.542	.233	.139	.586	.137	.082	.562	.239	.152	.318
MNet	Whatsapp	.763	.099	.063	.630	.098	.052	.616	.081	.052	.702	.101	.062	.678	.095	.057	.277
Wu22	Whatsapp	.866	.478	.431	**.889**	.727	.628	**.839**	.341	.233	.785	.313	.239	**.845**	.465	.383	.564
DF-Net	Whatsapp	**.905**	**.588**	**.495**	.883	**.762**	**.685**	.765	**.361**	**.249**	**.803**	**.324**	**.247**	.839	**.509**	**.419**	**.589**
DFCN	Weibo	.653	.191	.117	.676	.458	.319	.639	.227	.140	.706	.192	.125	.668	.267	.175	.370
FSim	Weibo	.542	.165	.100	.610	.453	.312	.568	.260	.165	.581	.150	.094	.575	.257	.168	.333
MNet	Weibo	.754	.099	.063	.620	.103	.056	.606	.057	.036	.671	.088	.053	.663	.087	.052	.267
Wu22	Weibo	.858	.466	.421	.883	.724	.626	**.808**	**.370**	**.253**	.780	.294	.219	.832	.463	.380	.558
DF-Net	Weibo	**.902**	**.584**	**.490**	**.890**	**.766**	**.684**	.759	.354	.245	**.791**	**.303**	**.230**	**.836**	**.502**	**.412**	**.583**

Table 3: Comparison of the state-of-the-art approaches DFCN [Zhuang et al., 2021], ForSim [Mayer and Stamm, 2019] (FSim), ManTra-Net [Wu et al., 2019] (MNet), Wu22 [Wu et al., 2022] and our proposed DF-Net. Highest metric values per benchmark dataset and OSN are marked **bold**.

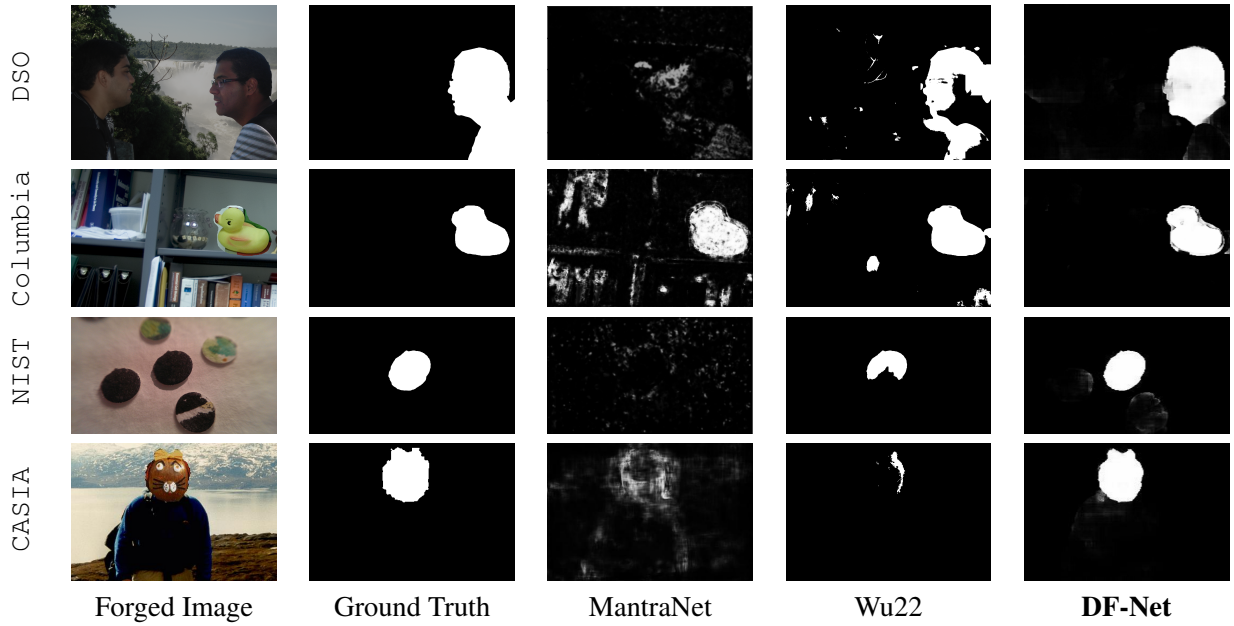

Figure 5: Examples of qualitative comparison of MantraNet [Wu et al., 2019], Wu22 [Wu et al., 2022] and our proposed DF-Net. Each line shows one example image for each of the four benchmark datasets DSO [Carvalho et al., 2013], Columbia [Hsu and Chang, 2006], NIST [National Institute of Standards and Technology (NIST), 2016], CASIA [Dong et al., 2013]. The five columns show: the forged image (input), manipulated area (ground truth), results (output) from MantraNet, Wu22 and DF-Net. We show example results of the M1 sub-model for the DSO and the NIST dataset.

4.2 Quantitative Comparison

As shown in Tab. 3, the DF-Net could clearly outperform ForSim, DFCN and ManTra-Net on the original benchmark datasets `CASIA V1`, `Columbia` and `NIST16`, and on the overall average of the metrics. The method Wu22 performed similar to the DF-Net. The overall average of DF-Net (0.5832) is just 0.0005 higher. Yet the situation for the OSN modified datasets has to be noted: in Fig. 4 we visualize the sum of AUC, F1 and IoU per dataset. The methods ForSim, DFCN and ManTra-Net show a large performance decrease for the modified datasets. Robustness against OSN transmitted data was the key contribution of Wu22 [Wu et al., 2022], hence their method performs only moderately worse on OSN transmitted images compared to the original ones. DF-Net on the other hand does not show a significant performance drop at all. This can be explained by the training process with image pre-scaling to $256x256$ pixels. This forces the DF-Network to learn manipulation traces which are even included in downscaled images.

Curiously enough, the metrics for DF-Net do sometimes even improve on the OSN-transmitted dataset versions. The effect is strongest for WhatsApp transmitted image data, where the average of the three metrics over all benchmark datasets increases by 0.00578. This effect can also be found for Wu22 [Wu et al., 2022], where all metrics are higher for the Facebook transmitted Columbia dataset compared to the original (non-transmitted) data.

5 Conclusion

In this paper, we propose a lightweight network architecture for image manipulation detection. We share our model, the Digital Forensics Network (DF-Net, with the community. This model shows a better performance than several state-of-the-art methods on four well-established benchmark datasets. In particular, DF-Net outperforms its competitors for images transmitted over popular social networks such as Facebook or WhatsApp. With a simple and practical training concept, the DF-Net addresses the challenges of lossy operations (downscaling, filtering) and focuses on robust manipulation features. In extensive evaluations, we show that DF-Net has virtually no performance degradation on OSN-transmitted images, which is a unique feature compared to competitors in the field of image forgery detection. Furthermore, the detection speed is significantly higher than that of its closest competitor since DF-Net does not rely on a tiling process (see Tab. 2).

Acknowledgement

Project 101083573 — GADMO

References

[Abecidan, 2023] Abecidan, R. (2023). ManTraNet pytorch implementation. `https://github.com/RonyAbecidan/Mantranet-pytorch`. Accessed: 2023-03-06.

[Carvalho et al., 2013] Carvalho, T., Riess, C., Angelopoulou, E., Pedrini, H., and Rocha, A. R. (2013). Exposing digital image forgeries by illumination color classification. *IEEE Trans. Inf. Forensics and Security*, 8(7):1182–1194.

[Dong et al., 2013] Dong, J., Wang, W., and Tan, T. (2013). Casia image tampering detection evaluation database. In *IEEE China Summit Inter. Conf. Signal Info. Proc.*, pages 422–426. IEEE.

[Fischinger and Boyer, 2023] Fischinger, D. and Boyer, M. (2023). DF2023: The digital forensics 2023 dataset for image forgery detection. *Irish Machine Vision and Image Processing conference*.

[Hsu and Chang, 2006] Hsu, Y. and Chang, S. (2006). Detecting image splicing using geometry invariants and camera characteristics consistency. In *IEEE Inter. Conf. Multim. Expo*, pages 549–552. IEEE.

[Kingma and Ba, 2015] Kingma, D. P. and Ba, J. (2015). Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference for Learning Representations (ICLR)*.

[Li et al., 2017] Li, H., Luo, W., and Huang, J. (2017). Localization of diffusion-based inpainting in digital images. *IEEE Transactions on Information Forensics and Security*, 12(12):3050–3064.

[Lyu et al., 2013] Lyu, S., Pan, X., and Zhang, X. (2013). Exposing region splicing forgeries with blind local noise estimation. *International Journal of Computer Vision*, 110:202–221.

[Mahmood et al., 2017] Mahmood, T., Irtaza, A., Mehmood, Z., and Mahmood, M. (2017). Copy-move forgery detection through stationary wavelets and local binary pattern variance for forensic analysis in digital images. *Forensic Science International, Elsevier*, 279:8–21.

[Mayer and Stamm, 2019] Mayer, O. and Stamm, M. C. (2019). Forensic similarity for digital images. *IEEE Transactions on Information Forensics and Security*.

[National Institute of Standards and Technology (NIST), 2016] National Institute of Standards and Technology (NIST) (2016). Nist nimble 2016 datasets. `https://www.nist.gov/itl/iad/mig/nimble-challenge-2017-evaluation/`.

[Ouyang et al., 2019] Ouyang, J., Liu, Y., and Liao, M. (2019). Robust copy-move forgery detection method using pyramid model and zernike moments. *Multimedia Tools and Applications*, 78:1–19.

[Ronneberger et al., 2015] Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*.

[Roy et al., 2018] Roy, A. G., Navab, N., and Wachinger, C. (2018). Recalibrating fully convolutional networks with spatial and channel 'squeeze & excitation' blocks. In *Medical Imaging, pages 540–549*.

[Verdoliva, 2020] Verdoliva, L. (2020). Media forensics and deepfakes: An overview. *IEEE Journal of Selected Topics in Signal Processing*, 14(5):910–932.

[Wang et al., 2017] Wang, Y., Tian, L., and Li, C. (2017). Lbp-svd based copy move forgery detection algorithm. In *2017 IEEE International Symposium on Multimedia (ISM)*, pages 553–556.

[Wu et al., 2022] Wu, H., Zhou, J., Tian, J., Liu, J., and Qiao, Y. (2022). Robust image forgery detection against transmission over online social networks. *IEEE Transactions on Information Forensics and Security*.

[Wu, 2023] Wu, Y. (2023). ManTraNet github repository. `https://github.com/ISICV/ManTraNet`. Accessed: 2023-06-09.

[Wu et al., 2019] Wu, Y., AbdAlmageed, W., and Natarajan, P. (2019). Mantra-net: Manipulation tracing network for detection and localization of image forgeries with anomalous features. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9535–9544.

[Zedan et al., 2021] Zedan, I. A., Soliman, M. M., Elsayed, K. M., and Onsi, H. M. (2021). Copy move forgery detection techniques: A comprehensive survey of challenges and future directions. *International Journal of Advanced Computer Science and Applications*, 12(7).

[Zhong and Pun, 2020] Zhong, J.-L. and Pun, C.-M. (2020). An end-to-end dense-inceptionnet for image copy-move forgery detection. *IEEE Transactions on Information Forensics and Security*, 15:2134–2146.

[Zhuang et al., 2021] Zhuang, P., Li, H., Tan, S., Li, B., and Huang, J. (2021). Image tampering localization using a dense fully convolutional network. *IEEE Transactions on Information Forensics and Security*, 16:2986–2999.