DEFACTO : Image and Face Manipulation Dataset

Gaël MAHFOUDI
ICD, University of Technology of Troyes
Troyes, France
gael.mahfoudi@utt.fr

Badr TAJINI
EURECOM
Biot, France
badr.tajini@eurecom.fr

Florent RETRAINT
ICD, University of Technology of Troyes
Troyes, France
florent.retraint@utt.fr

Frédéric MORAIN-NICOLIER
CReSTIC, University of Reims Champagne-Ardenne
Troyes, France
frederic.nicolier@univ-reims.fr

Jean Luc DUGELAY
EURECOM
Biot, France
jean-luc.dugelay@eurecom.fr

Marc PIC
SURYS
Bussy-Saint-Georges, France
m.pic@surys.com

***Abstract*—This paper presents a novel dataset for image and face manipulation detection and localization called DEFACTO. The dataset was automatically generated using Microsoft common object in context database (MSCOCO) to produce semantically meaningful forgeries. Four categories of forgeries have been generated. Splicing forgeries which consist of inserting an external element into an image, copy-move forgeries where an element within an image is duplicated, object removal forgeries where objects are removed from images and lastly morphing where two images are warped and blended together. Over 200000 images have been generated and each image is accompanied by several annotations allowing precise localization of the forgery and information about the tampering process.**

***Index Terms*—Image Forensics, Copy-Move, Splicing, Inpainting, Object-Removal, Face Morphing, Face-Swapping.**

I. Introduction

Digital image forensic has gained a lot of attention as it is becoming easier for anyone to make forged images. Several areas are concerned by the image manipulation: a doctored image can increase the credibility of fake news, impostors can use morphed images to pretend to be someone else. It became crucial to detect and locate these manipulations. Various techniques are involved in image manipulations, from simple global characteristic improvements (color enhancement, saturation, color remapping, contrast increase) to complex local forgeries (object incrustation or deletion, camouflage inpainting, morphing of structure). In this work we propose a novel dataset called DEFACTO meant for the study of a wide range of forgeries. Four types of forgeries have been automatically generated in our dataset: copy-move, splicing, object-removal and morphing.

As pointed out by the authors of [1], having publicly available datasets is positive for the researchers community as it can serve as a base to compare research results therefore we decided to publicly released the DEFACTO dataset at : *https://defactodataset.github.io*

Acknowledgments : The construction of this dataset was supported by the ANR project DEFACTO ANR-16-DEFA-0002 and by the French General Delegation for Armaments (DGA).

II. Related Work

Several publicly available datasets exist, the first one was the Columbia Gray Dataset [2]. It contains only splicing forgeries, no ground truth and images were only in grayscale. Two years later in 2006, they released a new version called Columbia Color Dataset [3] to extend the first one.

CASIA V1.0 and V2.0 [4] were introduced in 2009 to propose a larger (about 6000 tampered images) dataset with more realistic tampering. It contains splicing and copy-move forgeries with post-processing (blurring along edges or in other regions). They are still widely used as they contain a large number of forged images and their associated authentic images. Many datasets have been released later which only propose copy-move forgeries (MICC [5], IMD [6], CoMoFoD [7], COVERAGE [8], GRIP [9], FAU [6]). The National Institute of Standards and Technology (NIST) also released multiple datasets in the context of the Media Forensic Challenge 2019 [10]. It includes both image and video manipulations. While some parts are accessible upon a simple request, one needs to participate in the challenge to get access to the most recent version of the data.

Previously cited datasets were all made manually by the authors. The Wild Web dataset [11] released in 2015 was the first to introduce the real world tampered images. This dataset contains about 10000 images for which they manually created the ground truth binary masks to locate the forgery. More recently, the PS-Battle Dataset [12] was released. In this work, the author collected images from the active Reddit thread Photoshop Battles. In this thread, people try to produce the best photo manipulation from a given image. They gathered more than 10000 original images and 90000 tampered images. This dataset is meant to provide a long-lasting benchmark dataset and will keep growing with the Reddit community.

In this paper, we propose a novel dataset called DEFACTO meant for the study and training of image manipulation detection algorithms. We tried to produce a large amount of semantically meaningful forgeries for each category we defined in III-A.

978-9-0827-9703-9/19/$31.00 ©2019 IEEE

III. Dataset Overview

A. Forgery categories

In this dataset, we wanted to cover most of the common methods that one could use when creating a forgery. Hence, four major categories of forgeries have been considered : copy-move, splicing, object-removal and morphing.

Copy-move forgeries consist in the duplication of an element within the image. For splicing, one portion of an image is copied and pasted onto another image. In object-removal, an object is removed from the image by the use of inpainting algorithms. Finally, morphing consists in warping and blending two images together. For each forgery, post-processing may be applied (rotation, scaling, contrast ...). Those four categories can be seen as elemental forgery operations. An image composite would most likely be a composition of those basic operations. As the methods to detect those categories can be quite different, we decided to first construct a dataset where each image as only been forged using one of those categories only. The whole dataset content is detailed in Table. I. Generating those forgeries in a random manner is simple. This allows to produce a large number of forged images but they would be semantically meaningless and easy to detect by the human eyes. We wanted to create a dataset with meaningful forgeries and challenging for the human eye by removing most of the traces that an automatically generated forgery could contain. This goes from generating a proper segmentation of the object to select where to paste it in the final composite image. We believe that some mistakes (strong edges of the forged element) could introduce a bias in learning algorithms and wanted to address this issue.

B. Annotations

One advantage of automatically generated forgeries is that we can provide precise annotations for each image. In our dataset, each generated forged image is accompanied by diverse annotations to give further information on the tampering process.

1) General information: for each image, a detailed JSON file is provided. In this file, every operation made on the ground truth images are listed. Parameters used by each operation are detailed.

2) Localization: every image is also accompanied by one or more ground truth binary masks. One binary mask serves to localize the forgery under the $probe\_mask$ directory. For splicing, copy-move, face morphing and swapping, a binary mask under the $donor\_mask$ directory gives the localization of the source. Object-removal has a binary mask under the $inpaint\_mask$ directory which localize what has been filled by the inpainting algorithm.

IV. Automating forgery creation

A. Segmenting meaningful objects

To produce meaningful forgeries, we took advantages of MSCOCO dataset [13]. They collected more than 300,000 non-iconic images from Flickr. Afterwards, they defined 91 object categories of objects and annotated all the images.

TABLE I: Number of images per category in DEFACTO

Forgeries	Copy-Move	Inpainting	Splicing	Morphing
Images	19000	25000	105000	80000

Those annotations include the segmentation of the objects that we use as a base to produce our forgeries. The raw segmentation annotation cannot be used directly to generate a forgery as they are not precise enough (Fig. 1a). They need to be processed to obtain more suitable segmentation.

B. Refining segmentation

MSCOCO has over 2,000,000 object instances. Refining all annotations or even a small part by hand was not possible. We employed an alpha matting technique to refine the masks. Alpha matting consists in finding the foreground of an image

$$I = F\alpha + B(1-\alpha) \tag{1}$$

where I is the image, F the foreground, B the background and α the *alpha matte*.
This equation cannot be solved without any prior information. Thus, alpha matting techniques rely on a user input to define areas that are known to be part of the foreground F and areas that are known to be part of the background B. Those inputs can go from simple lines to what is called a trimap.

Trimap defines three areas: the foreground, the background and an unknown area (Fig. 1b).

Based on the given foreground and background areas, the alpha matting algorithm automatically affects a value to every unknown pixel to produce the final *alpha matte*.

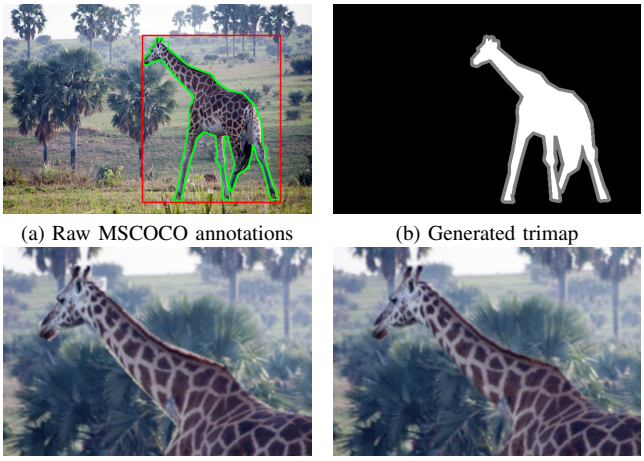
We used MSCOCO raw segmentation to construct the trimaps. First the foreground region is obtained by applying a morphological erosion to the raw MSCOCO mask to make sure that no background pixel is added to the foreground region. The unknown region is obtained by applying a morphological dilation to the raw mask (Fig. 1b). We then use a modified version of [14] to produce the *alpha matte*. This *alpha matte* is finally used to produce a much more convincing segmentation of the objects. This allows us to produce forgeries that are more pleasant (Fig. 1d).

Having a good segmentation of the objects is a first step toward the automatic generation of meaningful forgeries. Though it is not enough, removing or copying those objects in a random manner would most certainly produce bad results. For each category of forgery, a set of rules had to be applied to maximize the chances of producing good results.

C. Object Removal

There are many categories of inpainting algorithms, we used an exemplar-based inpainting method [15] as they have proved to be effective in various conditions and are still in common retouching software.

Inpainting methods are more efficient if the subject to remove is on a relatively simple background (Fig. 2). To produce convincing forgeries, we excluded objects for which the background was too complex. A border region is extracted by dilating the raw MSCOCO mask, and the standard deviation is computed within this region. Objects for which the standard deviation was below a fixed threshold were kept.



(a) Raw MSCOCO annotations (b) Generated trimap

(c) Copy-move without alpha matting (d) Copy-move with alpha matting

Fig. 1: MSCOCO mask refinement

(a) Ground truth image (b) Inpainted image

Fig. 2: Example of inpainting

D. Copy-move

For copy-move, the raw MSCOCO annotations were first used to produce the forgeries. Resulting images were almost systematically unpleasing due to strong visual artifacts on the objects borders (Fig. 1c). For this reason we had to use alpha matting as described in IV-B to obtain much better results in most cases (Fig. 1d). At first, objects would be copied and paste randomly within the image. This would cause the creation of images that were often semantically incorrect (*e.g.* people walking in the sky ...). To reduce chances of producing those kinds of images, we constrained the location of the forgery to be on the same axis as the source object (Fig. 3a and 3b). Decision to stay on the x or y axis is based on the $width$ and $height$ of the object. If $width > height$ then the object is duplicated on the x axis otherwise it is duplicated on the y axis. As for the object removal, we only kept objects on a fairly simple surroundings. This is to prevent to copy-move an object that is too tightly coupled with its context. For instance, in the MSCOCO dataset, a person and his backpack would be annotated separately, thus copying the person would not produce a good forgery as the backpack would be missing. Those rules allowed us to produce convincing copy-move forgeries (Fig. 3).

E. Splicing

Splicing is arguably the most complex forgery to generate automatically. When creating a splicing, an object from an image is pasted onto a new one. When manually creating the splicing, we can make sure that the target image has a similar point of view, lighting condition and so on. Those are essential to make a realistic splicing. Unfortunately, it is extremely difficult to assess all those elements automatically, thus it is extremely hard to produce good splicing forgeries. To address this issue, we decided to limit the MSCOCO categories used to produce splicing forgeries. We kept categories for which the objects appearance does not vary or depend too much on the point of view and the context. For example, a person could be standing, sitting, running ... and for each case, the appearance would vary a lot depending on the point of view to make it almost impossible to generate a convincing forged image. But if we take a sports ball, which is spherical, it will always look about the same and will be more easily spliced onto another image. Or if we take a bird which is either standing or flying, cutting and pasting it onto an image already containing birds has reasonable chances of producing an acceptable result. Objects are either pasted on an object of the same category or randomly pasted in a relatively smooth area to avoid pasting an object onto another one (Fig. 4b).

(a) Copy-move of the kite on the y axis (b) Copy-move of the bird on the x axis

Fig. 3: Example of copy-move

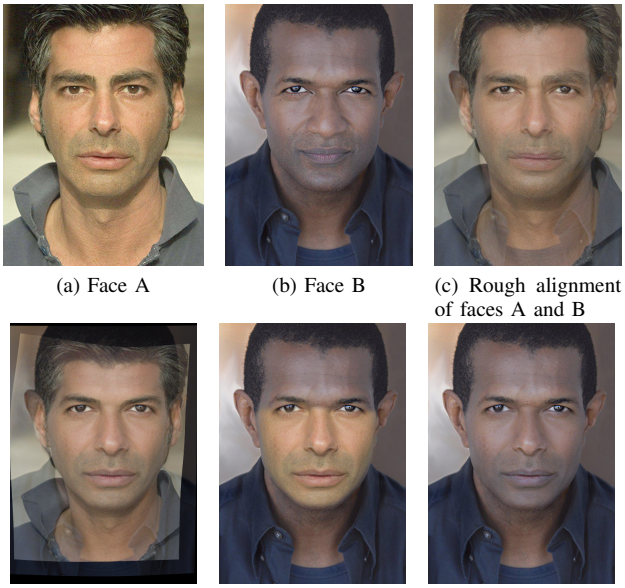
F. Face Morphing

Image morphing was originally used to produce smooth transitions between many images. One of the first notable appearances was in the Michael Jackson music video "Black or White" where the method described in [16] was used to morph peoples together. Nowadays, face morphing as received particular attention among the forensic community [17]. As shown in [18], Automatic Border Control systems are very vulnerable to face morphing attack. Thus we decided to include such forgeries into our dataset. We gathered public figure portraits on IMDB website and selected 200 front facing actors with a relatively neutral expression as a base to generate our face morphing forgeries.

The complete face morphing process can be seen in Fig. 5. Given two faces A and B, we used Dlib [19] to extract a set of facial landmarks. Thanks to those landmarks the two are first roughly align with respect to their eyes (Fig. 5c). A weighted average by a factor $\alpha \in [0, 1]$ of the two sets of landmarks is computed, and the two faces are warped [20] to this weighted average to precisely align them (Fig. 5d). Finally, the two faces are alpha blended using the same factor α. A local RGB scaling is performed to better take the skin tone into account. The factor α allows us to decide which face's biometrical traits are more visible. A α value of one produces what is commonly called a face swapping.

(a) Ground truths (b) Tampered

Fig. 4: Example of splicing



(a) Face A (b) Face B (c) Rough alignment of faces A and B

(d) Precise alignment of faces A and B (e) Without color balance (f) With color balance

Fig. 5: Automatic Face Morphing creation

V. Experiments

A. Baseline Models

We consider these methods [21]–[28] known from current state-of-the-art to assess our dataset (see Table II). All implementations of hand-crafted methods that we have used are provided by Zampoglou et al. [29]. For supervised methods, we assess four well-known network architectures in state-of-the-art to perform classification task. We use the code provided by the authors [25]–[28] for evaluation.

1) Evaluation Metric:

We use pixel-level F1 score, Area Under Curve (AUC) and Matthews Correlation Coefficient (MCC) as evaluation parameters for the methods proposed above. For a fair comparison, we adjust the prediction threshold to achieve a binary prediction mask and report the best score on all dataset. We perform AUC comparison by assigning the confidence score for each pixel in the potentially altered areas.

TABLE II: Hand-crafted and Learning-based methods

Hand-crafted Methods	**Technical Description**
ELA [21]	JPEG compression error
NOI1 [22]	Noise-based method using high-pass wavelet coefficients
CFA1 [23]	Camera's filter array patterns
DCT [24]	JPEG Blocking artifacts
Learning-based Methods	**Technical Description**
Bayar et al. [25]	Constrained Convolutional layer applied as an adaptive kernel
Rahmouni et al. [26]	Extract Statistical Features within a CNN framework
RGB-N [27]	RGB and Noise streams merged with bilinear pooling
EXIF-Consistency [28]	Self-Consistency method based on metadata

2) Implementation Details:

All the experiments are performed on one Nvidia TITAN Xp Graphics Card. To ensure fairness, all methods are trained exactly on NIMBLE 2017. All hyperparameters (learning rate, number of iterations, and weight of the regularization term) have been fixed by reproducing what was stated in original papers. Also when required, images have been resized [25]–[28], to match the size of the network input layer. We test those architectures on all our large scale dataset.

3) Experimental environment for training:

In data preparation, we split Nimble 2017 [10] dataset in half using scikit-learn model selection library to construct train and validation sets. These subsets are chosen randomly. For supervised methods [25]–[28], we train the models on Nimble 2017 and evaluate on DEFACTO dataset. For others [21]–[24], we directly test them on our dataset. We train all the models for 90k epochs. We stop the training process if the validation accuracy remains unchanged for 15 consecutive epochs.

4) Comparison on our dataset :

We report and discuss experimental results of our dataset by comparing 8 methods (Hand-crafted and Learned-based methods), with 3 performance metrics F1, AUC and MCC. We consider assessing one metric at a time. Notably, all methods based on hand-crafted features achieve lower performance. This is due to the fact that they all focus on specific forgery artifacts that contain only partial information for localization, which limits their performance. From the quantitative results shown in Table III, RGB-N [27] model achieves the best performance across the supervised baselines, which were trained on NIST 2017. Furthermore, performance declines, especially for shallow CNN architectures [25]. The method [28] seeks for similarities between patches and use deeper CNN with intensive training. Besides that and from state-of-the-art, it seems that this latter behaves well with uncompressed datasets containing only splicing techniques. The CNN [26] method integrates statistical feature extraction and finds the most appropriate features for an efficient boundary. The method [27] is effective for capturing the global context rather than the nearest pixels, which helps to capture manipulated areas. Finally, Deep Neural Networks have been proven as more efficient in detecting manipulations with fairly reliable results.

TABLE III: Pixel-level F1 score, *AUC* and *MCC* comparison on DEFACTO dataset.

Methods	ELA [21]	DCT [24]	NOI1 [22]	CFA1 [23]	**RGB-N** [27]	BAYAR-16 [25]	RAHMOUNI17 [26]	EXIF-Consistency [28]
F1	0.202	0.236	0.267	0.242	**0.544**	0.405	0.417	0.401
AUC	0.246	0.261	0.293	0.273	**0.579**	0.418	0.435	0.413
MCC	0.128	0.138	0.197	0.130	**0.519**	0.389	0.401	0.375

VI. Conclusion

In this paper we proposed a new publicly available dataset for the study of image manipulations. We automatically generated a large number of forged images (over 200000 images) and reviewed them manually to select the most realistic ones. To our knowledge this is the first dataset containing a large amount of manipulated face images that we consider as a growing problem and we hope that it can serve as a benchmark for future research in the field. As alternative prospects, we also consider two scenarios, the first one will consist in evaluating the complexity of DEFACTO by testing several hand-crafted and learning-based methods. For the second one, we will compare with other existing datasets in state-of-the-art.

References

[1] Lilei Zheng, Ying Zhang, and Vrizlynn L.L. Thing. A survey on image tampering and its detection in real-world photos. *Journal of Visual Communication and Image Representation*, 58:380–399, January 2019.

[2] Tian-Tsong Ng and Shih-Fu Chang. A data set of authentic and spliced image blocks. Technical report, Columbia University, June 2004.

[3] Yu-feng Hsu and Shih-fu Chang. Detecting Image Splicing using Geometry Invariants and Camera Characteristics Consistency. In *2006 IEEE International Conference on Multimedia and Expo*, pages 549–552, Toronto, ON, Canada, July 2006. IEEE.

[4] 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, Beijing, China, July 2013. IEEE.

[5] I. Amerini, L. Ballan, R. Caldelli, A. Del Bimbo, and G. Serra. A SIFT-Based Forensic Method for Copy–Move Attack Detection and Transformation Recovery. *IEEE Transactions on Information Forensics and Security*, 6(3):1099–1110, September 2011.

[6] V. Christlein, C. Riess, J. Jordan, C. Riess, and E. Angelopoulou. An evaluation of popular copy-move forgery detection approaches. *IEEE Transactions on Information Forensics and Security*, 7(6):1841–1854, Dec 2012.

[7] D. Tralic, I. Zupancic, S. Grgic, and M. Grgic. Comofod — new database for copy-move forgery detection. In *Proceedings ELMAR-2013*, pages 49–54, Sep. 2013.

[8] Bihan Wen, Ye Zhu, Ramanathan Subramanian, Tian-Tsong Ng, Xuanjing Shen, and Stefan Winkler. Coverage – a novel database for copy-move forgery detection. In *IEEE International Conference on Image processing (ICIP)*, pages 161–165, 2016.

[9] Davide Cozzolino, Giovanni Poggi, and Luisa Verdoliva. Copy-move forgery detection based on PatchMatch. In *2014 IEEE International Conference on Image Processing (ICIP)*, pages 5312–5316, Paris, France, October 2014. IEEE.

[10] Haiying Guan, Mark Kozak, Eric Robertson, Yooyoung Lee, Amy N. Yates, Andrew Delgado, Daniel Zhou, Timothee Kheyrkhah, Jeff Smith, and Jonathan Fiscus. MFC Datasets: Large-Scale Benchmark Datasets for Media Forensic Challenge Evaluation. In *2019 IEEE Winter Applications of Computer Vision Workshops (WACVW)*, pages 63–72, Waikoloa Village, HI, USA, January 2019. IEEE.

[11] Markos Zampoglou, Symeon Papadopoulos, and Yiannis Kompatsiaris. Detecting image splicing in the wild (WEB). In *2015 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, pages 1–6, Turin, Italy, June 2015. IEEE.

[12] Silvan Heller, Luca Rossetto, and Heiko Schuldt. The PS-Battles Dataset - an Image Collection for Image Manipulation Detection. *arXiv:1804.04866 [cs]*, April 2018. arXiv: 1804.04866.

[13] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft COCO: Common Objects in Context. *arXiv:1405.0312 [cs]*, May 2014. arXiv: 1405.0312.

[14] Kaiming He, Christoph Rhemann, Carsten Rother, Xiaoou Tang, and Jian Sun. A global sampling method for alpha matting. In *CVPR 2011*, pages 2049–2056, Colorado Springs, CO, USA, June 2011. IEEE.

[15] M. Daisy, P. Buyssens, D. Tschumperle, and O. Lezoray. A smarter exemplar-based inpainting algorithm using local and global heuristics for more geometric coherence. In *2014 IEEE International Conference on Image Processing (ICIP)*, Paris, France, October 2014. IEEE.

[16] Thaddeus Beier and Shawn Neely. Feature-based image metamorphosis. *SIGGRAPH Comput. Graph.*, 26(2):35–42, July 1992.

[17] Andrey Makrushin and Andreas Wolf. An Overview of Recent Advances in Assessing and Mitigating the Face Morphing Attack. page 5, 2018.

[18] Matteo Ferrara, Annalisa Franco, and Davide Maltoni. The magic passport. In *IEEE International Joint Conference on Biometrics*, pages 1–7, Clearwater, FL, USA, September 2014. IEEE.

[19] Davis E. King. Dlib-ml: A machine learning toolkit. *Journal of Machine Learning Research*, 10:1755–1758, 2009.

[20] F.L. Bookstein. Principal warps: thin-plate splines and the decomposition of deformations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(6):567–585, June 1989.

[21] Neal Krawetz. A Picture's Worth: Digital Image Analysis and Forensics. Technical report, Black Hat Briefings USA, 2017.

[22] Babak Mahdian and Stanislav Saic. Using noise inconsistencies for blind image forensics. *Image and Vision Computing*, 27(10):1497–1503, September 2009.

[23] 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, October 2012.

[24] Shuiming Ye, Qibin Sun, and Ee-Chien Chang. Detecting Digital Image Forgeries by Measuring Inconsistencies of Blocking Artifact. In *Multimedia and Expo, 2007 IEEE International Conference on*, pages 12–15, Beijing, China, July 2007. IEEE.

[25] Belhassen Bayar and Matthew C. Stamm. 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 - IH&MMSec '16*, pages 5–10, Vigo, Galicia, Spain, 2016. ACM Press.

[26] Nicolas Rahmouni, Vincent Nozick, Junichi Yamagishi, and Isao Echizen. Distinguishing computer graphics from natural images using convolution neural networks. In *2017 IEEE Workshop on Information Forensics and Security (WIFS)*, pages 1–6, Rennes, December 2017. IEEE.

[27] Peng Zhou, Xintong Han, Vlad I. Morariu, and Larry S. Davis. Learning Rich Features for Image Manipulation Detection. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1053–1061, Salt Lake City, UT, USA, June 2018. IEEE.

[28] Minyoung Huh, Andrew Liu, Andrew Owens, and Alexei A. Efros. Fighting Fake News: Image Splice Detection via Learned Self-Consistency. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, *Computer Vision – ECCV 2018*, volume 11215, pages 106–124. Springer International Publishing, Cham, 2018.

[29] Markos Zampoglou, Symeon Papadopoulos, Yiannis Kompatsiaris, Ruben Bouwmeester, and Jochen Spangenberg. Web and social media image forensics for news professionals, 2016.