Weakly-supervised deepfake localization in diffusion-generated images

Dragoș-Constantin Țânțaru
Bitdefender
dtantaru@bitdefender.com

Elisabeta Oneață
Bitdefender
eoneata@bitdefender.com

Dan Oneață
University Politehnica of Bucharest
dan.oneata@gmail.com

Abstract

The remarkable generative capabilities of denoising diffusion models have raised new concerns regarding the authenticity of the images we see every day on the Internet. However, the vast majority of existing deepfake detection models are tested against previous generative approaches (e.g. GAN) and usually provide only a "fake" or "real" label per image. We believe a more informative output would be to augment the per-image label with a localization map indicating which regions of the input have been manipulated. To this end, we frame this task as a weakly-supervised localization problem and identify three main categories of methods (based on either explanations, local scores or attention), which we compare on an equal footing by using the Xception network as the common backbone architecture. We provide a careful analysis of all the main factors that parameterize the design space: choice of method, type of supervision, dataset and generator used in the creation of manipulated images; our study is enabled by constructing datasets in which only one of the components is varied. Our results show that weakly-supervised localization is attainable, with the best performing detection method (based on local scores) being less sensitive to the looser supervision than to the mismatch in terms of dataset or generator.

1. Introduction

Image generation is improving by the day and it is arguably past the point where it is possible to perceptually distinguish between generated (fake) and real content. Generative adversarial models (GAN) [19], normalizing flows [46], denoising diffusion probabilistic models (DDPM) [54]—all provide excellent means for the creation of digital art or entertainment content. However, the advances in image generation come at the cost of also easing malicious use, *e.g.*, by altering reality or spreading misinformation. To counter these harmful effects, deepfake detection methods are developed to discriminate between fake and real samples [43, 44, 56].

Among the classes of generative models, diffusion models are emerging as the dominant paradigm [14], showcasing impressive results on a wide array of tasks including text-controlled image generation [45, 48, 51, 61] or image-to-image translation [38, 48, 50, 61]. Prior work on deepfake detection has naturally mostly considered detecting content generated by GANs [5, 20, 41, 57, 60], but the computer vision community is now starting to consider DDPMs [9, 47]. Here we continue this direction, going one step further to address the task of weakly-supervised deepfake localization.

First, we extend prior approaches to localise the manipulated area and not only label the entire image as fake or real. The binary output of the typical deepfake detection methods provides only coarse and opaque information, especially in the frequent case of local manipulations and forgeries. In this scenario, we would be much better served by a richer representation that could pinpoint which part of the image is likely to have been generated. Another benefit of localization is that it allows the end-user to take more informed decisions. For example, changing the color of one's eyes may just be an innocuous enhancement of the user's appearance, but the alteration of the movement of the lips in a video may hint towards a malicious use. Instead of deciding upfront what is deemed to be fake or real, a localization method can defer this decision to the end user, who is more informed and can tailor the method to their use case.

Second, in contrast to prior work, which addresses localization in a fully-supervised setting [26, 33, 58, 64], we consider a weakly-supervised scenario, where we assume that we only have access to image-level labels and the models are not explicitly trained for localization. This setup is motivated by the fact that generative methods are usually first developed in the context of full-image synthesis, and only then extended to the more specific cases of local editing, such as inpainting or attribute manipulation. Moreover, ground truth manipulation masks might not always be available, especially for newly developed local manipulation methods. Training a deepfake localization method in a weakly-supervised fashion (based on a global label) would allow us to be one step ahead of the potentially harmful uses involving local changes.

Our work brings the following contributions:

1. We propose a **weakly-supervised** framework for deepfake localization in images that allows to systematically uncover the importance of various factors (**model**, **supervision type**, **dataset**, **generator**) in the context of weakly-supervised localization of face manipulations.

2. We generate a detailed dataset (more than 125k images) with locally- and fully-manipulated images that allows the **disentanglement** of different factors in deepfake manipulation localization. The images are obtained using either newly introduced state-of-the-art generative models or a novel inpainting approach that incorporates a pretrained LDM [48] model in a diffusion-based inpainting method [38].

3. We provide **extensive quantitative and qualitative results** to understand the fundamental factors underlying the performance of weakly-supervised localization models. Our analysis reveals the severity of out-of-domain degradation, provides insights into the model's sensitivity to looser supervision or dataset mismatch, and quantifies the performance across multiple classes of generative models. Our code and dataset are available at https://github.com/bit-ml/dolos.

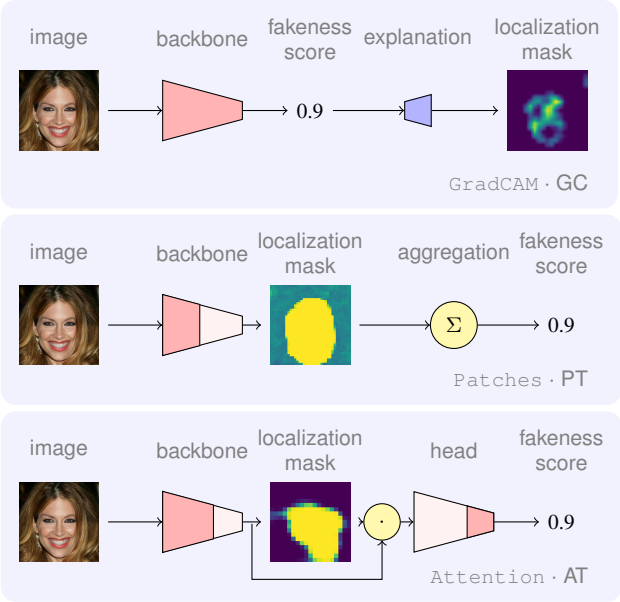

Figure 1. Overview of the three types of approaches proposed for the detection and localization of deepfakes. Each method is able to produce a fakeness score (for detection) and a mask (for localization); the mask is obtained either explicitly (for the first model) or implicitly (for the second and third models).

2. Related Work

Deepfake detection of GAN content. There is a vast and continuously-growing body of work dedicated to the detection of GAN-generated images, see [39, 43, 44, 56] for reviews. Prior research has revealed many particularities of GAN content [10, 20, 60], an important observation being the appearance of a fingerprint—an imperceptible pattern, which allows the identification of the GAN method and training dataset [41, 60]. Wang *et al*. [57] also observe that all CNN-generated images share common systematic artifacts, that can be easily picked up by a classifier, while in [20] the authors indicate that downsampling might destroy these high-frequency artifacts, which are the key to detection.

Deepfake detection of DDPM content. Preliminary works on detecting diffusion-generated images made use of high-level cues such as inconsistencies in lighting [17] or perspective distortion [18]. However, more common end-to-end detection networks were also tested on diffusion images [9, 47], focusing on the transferability across classes of generative models (from GAN to DDPM, and vice versa). The prevailing observation is that detectors trained on one type of data do not generalize well to the other, but finetuning helps.

Local manipulations. A common setup in deepfake creation is altering a person's face by reenactment, replacement, editing or synthesis using techniques known as face swap, face transfer, facial attribute manipulations or inpainting [43]. These approaches result in local manipulations and are traditionally GAN-based. Increasingly larger and more complex datasets and challenges have emerged [15, 25, 28, 31, 34, 49] and, with these, a considerable effort has been made to expose those types of fakes [1, 3, 16, 23, 39, 63]. However, actually localizing manipulations has arguably received less attention than detecting whether an image is fake or not. Works that tackle localization rely on local noise fingerprint patterns [21, 33, 40, 64], attention mechanisms [12, 13, 42] or self-consistency checks [2, 27]. Very recent, concurrent works proposed a forensic framework for general manipulation localization [21] and a hierarchical fine-grained formulation for image forgery detection [22]. Similar to us they consider diffusion-generated data with local forgeries, but differently they assume full supervision.

3. Methodology

We first describe the methods used for deepfake detection and weakly-supervised localization (§3.1). Then we detail the generative techniques that we are interested in detecting (§3.2) and the datasets generated with these methods (§3.3).

3.1. Methods for detection and localization

The task of deepfake detection consists of predicting whether an image is either real or fake. This task is usually framed as a binary classification problem and it is addressed using standard classification networks. In this paper we are interested in evaluating the capabilities of such methods in a

weakly-supervised setting: if we assume only image-level labels, can these classifiers be successfully used for *localization* of partially manipulated images?

We identify and investigate three categories of architectures suitable for weakly-supervised localization. These methods are based on either explanations (`GradCAM`), local scores (`Patches`) or attention (`Attention`) (for visual depictions see Figure 1). The first category is a general technique that given a trained classification network it uses explainability techniques to highlight the most predictive regions for the "fake" label. The other categories implicitly construct the localization maps: `Patches` produces local patch scores that are then used for classification, while `Attention` predicts an activation map that is used to pool relevant classification features. To allow for a fair comparison we fix the backbone and, in particular, we select the Xception network [8], which has been shown to yield excellent results for deepfake detection of faces [49].

The proposed methods are inspired by and build upon state-of-the art deepfake detection methods [5, 12, 49], but we further modify them as described below.

`GradCAM`. While GradCAM explanations were previously used in the deepfake detection literature [4, 53, 59, 63], they were mostly shown as qualitative results and rarely (if ever) evaluated quantitatively, in terms of how well they localize the input alterations. In this paper we aim to quantify their performance and contrast them with other weakly-supervised localization methods. Concretely, we endow the Xception [8] network with localization capabilities by applying GradCAM [52] on the activations produced by block 11, the one before the last downsampling operation.

`Patches`. We use Patch–Forensics [5], which is a truncated image classification network: it takes the feature activations after a few layers and projects them to a patch-level score using 1×1 convolutions. At train time, the loss is computed for each patch, while at test time, it produces a detection score by averaging the per-patch softmax scores. The authors experiment with two backbones (Xception [8] and ResNet [24]) and vary the number of layers that are kept. We chose the Xception backbone truncated after the second block of layers, as this combination was shown to yield good performance [5]. One advantage of `Patches` is that its output naturally corresponds to a localization map. While visualizations of the activation maps were shown in the original work, the localization performance was not quantified.

`Attention`. We start from [12] which augments an Xception [8] backbone with a learned attention mask that is used to modulate the feature maps produced by the network. The network is trained in a multi-task setting, with a loss on the full-image fakeness score and another one on the localization mask. In the weakly-supervised scenario, when no groundtruth mask is provided, the second term ensures that the maximum value of the predicted mask agrees with the image-level label. We modify the original implementation in [12] to improve the performance and stabilize the training. First, we replace the L1 loss on the mask with the binary cross-entropy loss (CE). Second, we cross-validate the weight λ that balances the two losses. Our final loss is:

$$L = \mathrm{CE}(y, \hat{y}) + \lambda \mathrm{CE}(y, \max \hat{\mathbf{m}}), \tag{1}$$

where y is the true image label, $\hat{y}$ is the fakeness score and $\hat{\mathbf{m}}$ is the estimated localization mask.

Fully-supervised localization. Along with the weakly-supervised setup we also consider the fully-supervised case to show an upper bound on the performance. Since not all considered detection methods are able to be trained in a fully-supervised setting out of the box, we modify them to accommodate this setup: for `GradCAM` we truncate after block 11 and add a fully convolutional layer as in [37]; for `Attention` we keep only the loss on the mask, otherwise the architecture remains the same; for `Patches` we maintain the same architecture, but instead of using the image label to supervise each feature prediction, we use the downsized mask as groundtruth.

3.2. Dataset generation methods

We use diffusion models to generate both full images and locally-inpainted ones.

Diffusion denoising probabilistic models (DDPM) [54] are a class of generative models trained to reverse a diffusion process. The forward diffusion process iteratively adds Gaussian noise to a sample until its distribution reaches a standard normal. The reverse denoising process gradually removes noise, producing novel samples when starting from a random image. The reverse process is implemented as a neural network (with parameters θ) that predicts the mean $\mu_\theta(\mathbf{x}_t, t)$ and covariance $\Sigma_\theta(\mathbf{x}_t, t)$ of a Gaussian distribution:

$$p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) = \mathcal{N}\left(\mathbf{x}_{t-1}; \mu_\theta(\mathbf{x}_t, t), \Sigma_\theta(\mathbf{x}_t, t)\right), \tag{2}$$

where $\mathbf{x}_t$ are images that are sequentially generated, from $t = T$ to $t = 1$.

Repaint: Inpainting with diffusion. The task of inpainting is to fill in the missing regions of an image $\mathbf{x}_0$ such that the resulting composition looks natural; the missing regions are usually specified by a binary mask $\mathbf{m}$. For inpainting with diffusion we use the approach of Lugmayr *et al.* [38], whose method performs mask-guided decoding on any pretrained DDPM. More precisely, at generation time they first sample a new image $\hat{\mathbf{x}}_t$ from the previously-generated image, $\hat{\mathbf{x}}_{t+1}$, according to Equation (2), but then they replace the values of $\hat{\mathbf{x}}_t$ outside the given mask $\mathbf{m}$ with the values of the original image encoded after t steps $\mathbf{x}_t$:

$$\hat{\mathbf{x}}_t \leftarrow \mathbf{m} \odot \hat{\mathbf{x}}_t + (1 - \mathbf{m})\mathbf{x}_t \tag{3}$$

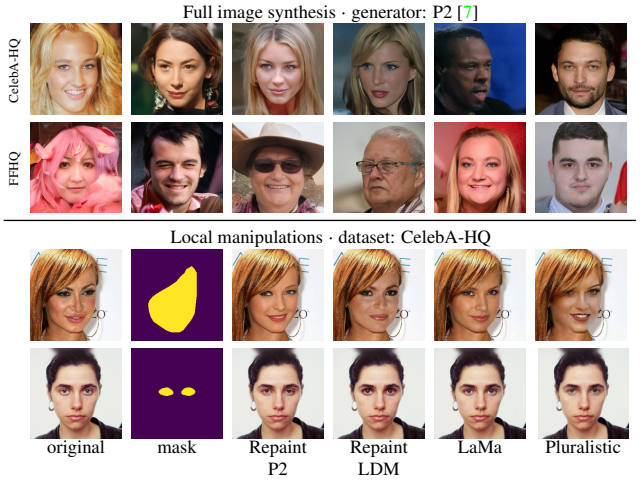

Figure 2. Examples from our generated dataset. The first two rows represent fully-generated images using P2 on CelebAHQ and FFHQ, respectively. The last two rows represent locally-inpainted images using Repaint–P2, Repaint–LDM, LaMa, Pluralistic, respectively; notice the high realism of images obtained with both large and small masks.

This procedure ensures that the values outside the mask are preserved from the original image $\mathbf{x}_0$.

Repaint–LDM: Inpainting with diffusion in the latent space. Latent diffusion models (LDM) [48] have been shown to offer a scalable approach to generating high-fidelity images. Their main idea consists of performing diffusion in the (low-dimensional) latent space of a variational autoencoder (VAE). We translate this idea to inpainting by running the Repaint scheduler (Equation 3) in the latent space, $\mathbf{x} \leftarrow \mathsf{enc}(\mathbf{x})$, of the variational autoencoder and using an appropriately downsized mask, $\mathbf{m} \leftarrow \mathsf{resize}(\mathbf{m})$. This procedure generates an (inpainted) latent code, $\hat{\mathbf{x}}$, which is then inverted to the original pixel space using the decoder of the VAE, $\mathsf{dec}(\hat{\mathbf{x}})$. Notably, this method allows us to inpaint an image using any existing pretrained LDM model. To the best of our knowledge, this approach to inpainting is novel.

3.3. Datasets

To train and evaluate our models, we use real images and two types of fake images: fully-synthesized and locally-manipulated images. The datasets are summarized in Table 1 and examples are shown in Figure 2.

Real data. We use the CelebA-HQ and FFHQ face datasets as sources of real data. CelebA-HQ [29] consists of 30k images that were selected and processed from the CelebA dataset [36]; we keep the original splits for training, validation and testing. FFHQ [30] consists of 70k PNG images that have been crawled from Flickr and automatically aligned and cropped. Both datasets are popular choices for training generative models and, consequently, are suitable choices for training deepfake detection models. We select a subset of 9k train and 900 validation images from each of the two datasets to match the number of fake images that are generated.

	Generator			Num. samples		
Type	Family	Model	Dataset	Train	Val.	Test
Real	–	–	CelebA	9k	900	900
Real	–	–	FFHQ	9k	900	–
Fake full	Diffusion	P2	CelebA	9k	1k	–
Fake full	Diffusion	P2	FFHQ	9k	1k	–
Fake local	Diffusion	Repaint–P2	CelebA	30k	3k	8.5k
Fake local	Diffusion	Repaint–P2	FFHQ	30k	3k	–
Fake local	Latent diff	Repaint–LDM	CelebA	9k	900	900
Fake local	Fourier conv	LaMa	CelebA	9k	900	900
Fake local	GAN	Pluralistic	CelebA	9k	900	900

Table 1. Details of our proposed dataset, which contains locally- and fully-generated images from multiple types of generators. The dataset is designed to allow for a principled analysis of multiple factors: manipulation type, generator, source dataset. We provide: (i) fully-generated images on CelebA-HQ and FFHQ using P2 [7]; (ii) locally-inpainted images on FFHQ using Repaint–P2 and on CelebA-HQ using Repaint–P2, Repaint–LDM, Pluralistic [62], LaMa [55] (using the same masks).

Fake data: Full-image synthesis. We use the perception-prioritized (P2) diffusion method of Choi *et al*. [7] to sample fully-synthetic images. We chose this approach because (i) the authors provide pretrained models on the two real datasets mentioned above (CelebA-HQ and FFHQ), which enable a systematic experimentation, and (ii) the models are lightweight and hence the inference is reasonably fast. For both datasets we sample 10k images: 9k for training and 1k for validation. We do not evaluate on these fully-synthesized sets, hence no test set is provided. We refer to these datasets as P2/CelebA-HQ and P2/FFHQ, respectively.

Fake data: Local manipulations. We generate two locally-manipulated datasets using the Repaint method [38] to inpaint images from the CelebA-HQ and FFHQ datasets. We use the Repaint method on top of pretrained P2 models, namely its variants trained on CelebA-HQ and FFHQ, respectively. The inpainted regions correspond to various face attributes (skin, hair, eyes, mouth, nose, glasses). For CelebA-HQ, these annotations were manually labeled and are available in the CelebAMask-HQ [29] extension of the dataset, while for FFHQ these are obtained using a pretrained face segmentation method [32]. Given an image (corresponding to the identity of a person) we generate multiple inpaintings by randomly sampling masks corresponding to these face attributes and, for the smaller parts (eyes, mouth, nose), by also dilating them with a kernel of randomly-chosed size,

but up to 15 pixels. We refer to the resulting datasets as Repaint–P2/CelebA-HQ and Repaint–P2/FFHQ; the former will represent our main test bed, while the latter is used only at training.

To be able to systematically study the importance of the generator we inpainted a subset of 9k images used in Repaint–P2/CelebA-HQ with three other methods: Repaint-LDM (ours), LaMa [55], Pluralistic [62]. Repaint-LDM adapts the Repaint method to operate in the latent space by using the LDM model [48]; LaMa is an inpainting method that uses an autonecoder with Fourier convolutions [6]; Pluralistic is a conditional variational autoencoder with adversarial loss. We have chosen these methods since they all provide pretrained models on the CelebA-HQ dataset. This allows us to inpaint the same images using the same masks, and isolate the differences attributed to the change of generator.

4. Experimental setup

Implementation details. Following the recommendation of Chai *et al*. [5], we ensure that real and fake images both follow exactly the same preprocessing steps prior to passing them through the detection methods. These steps include the input resolution and resize algorithm. Consequently, we process both CelebA-HQ and FFHQ images as they were processed for training the generator, that is, we resize them to 256 $\times$ 256 using bicubic interpolation.

Tasks and metrics. Localization is the main task that we tackle. We report intersection over union (IoU) and pixel-wise binary classification accuracy (PBCA). These metrics assume binary prediction and we use a fixed threshold of 0.5 for binarization. The detection methods generate masks of different sizes: 19 $\times$ 19 for `GradCAM` and `Attention`, 37 $\times$ 37 for `Patches`. For a fair evaluation we resize them to the size of the input image: 256 $\times$ 256.

We also report results on detection, the task of telling apart fake images from real images. We rank the images by their per-image fakeness score, which is output by each method as illustrated in Figure 1. The detection performance is then measured in terms of average precision (AP), which is a threshold-free metric.

5. Experiments

Our experiments evaluate the proposed methods with different levels of supervision, gradually changing the dataset and the generators in order to quantify their importance for localization. We investigate the performance using three main levels of supervision:

- **Setup `A` (label & full)** is a weakly-supervised setup in which we have access to fully-generated images as fakes and, consequently, only image-level labels. We use 9k fake images, fully synthesized by P2, and 9k real images from the corresponding dataset on which P2 was trained.

- **Setup `B` (label & partial)** is a weakly-supervised setup in which we have access to partially-manipulated images, but only with image-level labels (no localization information). This means that while an image may be labelled as "fake", not all of its regions are fake. We use 9k locally-modified images by Repaint–P2 and 9k real images from the corresponding training dataset.

- **Setup `C` (mask & partial)** is a fully-supervised setting, in which we have access to ground-truth localization masks of partially-manipulated images. We uses 30k locally-modified images by Repaint–P2; for this setup, no real images are used.

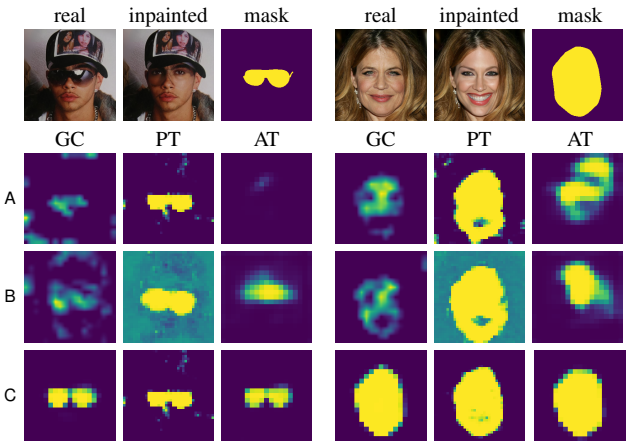

Figure 3. Soft localization maps produced by the three approaches using different levels of supervision. `Patches` can accurately detect the manipulations after having seen only fully-generated fake images (scenario `A`) or locally-inpainted images with only image-level supervision (scenario `B`). Both `Attention` and `GradCAM` struggle in scenarios `A` and `B`. All methods recover the manipulated region in the fully supervised scenario, `C`. This suggests that operating at a patch level is better suited for recovering local manipulations than either using GradCAM or attention.

To evaluate localization we use 8.5k locally-manipulated images produced by Repaint–P2/CelebA-HQ and to evaluate detection we use 900 real images from CelebA-HQ and 900 fakes from Repaint–P2/CelebA-HQ. Note that the evaluation is carried on the same data regardless of the setup. Table 1 from the supplementary material summarizes the data used in each of the three setups.

5.1. Evaluating localization abilities

We evaluate all three proposed approaches for localization in the three setups described above. To exclude other factors of variation we maintain the image generator and source

setup	sup.	generator	IoU (%)			PBCA (%)			AP (%)		
			GC	PT	AT	GC	PT	AT	GC	PT	AT
A	label	full	16.8	**64.9**	9.7	83.1	**96.7**	83.4	67.3	**95.3**	79.3
B	label	partial	21.5	**37.7**	23.2	85.1	79.8	**86.3**	94.4	**95.3**	94.4
C	mask	partial	83.7	**84.5**	70.3	96.8	**98.6**	97.6	–	–	–

Table 2. Evaluation of the three selected localization techniques (GradCAM GC, Patches PT, Attention AT) on the Repaint–P2/CelebA-HQ dataset using different levels of supervision: image-level label on full images (A), image-level label on locally manipulated images (B), and fully-supervised masks (C). We evaluate both localization (using IoU and PBCA) and detection (using AP). Patches systematically outperforms the other two methods under most of the scenarios and metrics.

dataset fixed, that is, for scenario A we train on P2/CelebA-HQ, while for scenarios B and C we use Repaint–P2/CelebA-HQ. Real data from CelebA-HQ is used in setups A and B, while for the fully supervised scenario, setup C, real data is not needed. Results for both localization and detection are shown in Table 2.

Among the selected methods, we see that Patches generally outperforms the other two approaches across multiple setups and metrics (bold values in Table 2). We see that localization performance is strong for all methods when training in the fully supervised scenario (setup C) and performance drops as we move to the two weakly supervised setups (setups A and B). Interestingly, GradCAM and Attention perform better in setup B than in setup A, while for Patches we observe the reverse trend. We believe that Patches is worse in setup B because the loss is set at patch-level, and the patch labels are inherently noisy as we use partially-manipulated images at input.

In terms of detection (the 'AP' columns in Table 2), we observe strong performance of Patches in both weakly supervised setups, A and B. Interestingly, the detection performance is good for all models in setup B. In retrospect, this is expected since for the detection task in setup B the train data matches the test data.

Figure 3 shows examples of the localization maps produced by the detection methods in all three scenarios. We notice that Patches is able to partially recover the manipulated areas even in setups A and B. In setup B we observe that due to the noisy labels the model fires also on the background regions. GradCAM and Attention struggle more in the weakly-supervised scenarios and their outputs are qualitatively different: the former seems to produce weaker activations, which are spread through irrelevant areas of the image (especially in scenario A), while the latter produces less precise localizations.

5.2. Generalization across source datasets

Generalization is a key desirable property of deepfake detectors. Here, we assess how localization is affected by datasets shifts. To this end, we design an experiment in which the training and testing data come from different source datasets, while fixing the generator and the detection method. Training is either based on fake data derived from CelebA-HQ (P2/CelebA-HQ for scenario A and Repaint–P2/CelebA-HQ for scenarios B and C) or FFHQ (P2/FFHQ for scenario A and Repaint–P2/FFHQ for scenarios B and C), while the testing is carried on Repaint–P2/CelebA-HQ.

	sup.	generator	CelebA-HQ			FFHQ		
			IoU	PBCA	AP	IoU	PBCA	AP
A	label	full	64.9	96.7	95.3	25.1	88.9	84.4
B	label	partial	37.7	79.8	95.3	23.3	64.4	75.2
C	mask	partial	84.5	98.6	–	32.3	89.2	–

Table 3. Evaluation of Patches on the Repaint–P2/CelebA-HQ dataset using two training datasets: CelebA-HQ and FFHQ. When the source dataset does not match the target one, we observe a consistent drop in performance across all scenarios. This is more evident in scenario B where only image-level supervision is available for locally-manipulated images.

Quantitative results are shown in Table 3 for all scenarios under both localization and detection metrics. We observe a consistent drop in performance across all scenarios and metrics when there is a dataset mismatch. A closer look at the soft localization maps reveals a more complete picture. The 'different' columns in Figure 4 show that training on FFHQ still produces qualitatively reasonable predictions even for small regions (nose and mouth). However, in this mismatched setting the predictions are less certain at the boundaries and the masks appears to be eroded or with holes.

To better assess the estimated localization maps we look at the how their accuracy varies with the size of the manipulated region. In Figure 5 we show the IoU score as a function of the mask area for the three setups when (i) the training dataset matches the one at test time (blue line), and (ii) is different (orange line). We observe that larger manipulations are generally easier to correctly locate (increasing IoU with area) and that the dataset mismatch results in a sizable drop in performance. However, for setup B the slopes are much steeper and the gap between the two curves is reduced. We believe that this happens because in this setup the model fires

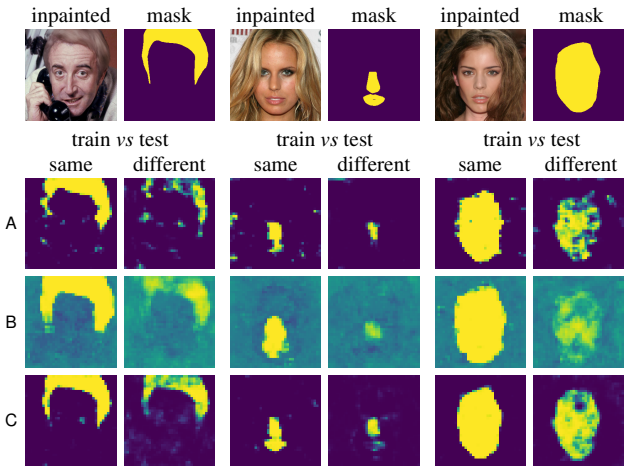

Figure 4. Soft localization maps produced by `Patches` when using the same *vs* different source datasets for training and testing. For training we use data derived form either CelebA-HQ or FFHQ, while for testing we use data derived from CelebA-HQ. When there is a dataset mismatch (the 'different' column), we observe maps that are less sharp and eroded, especially in the weakly supervised scenarios, `A` and `B`. The noisy training of scenario `B` dims the separation between real and fake regions.

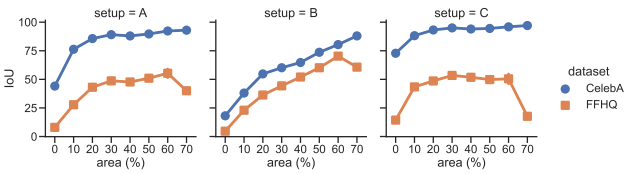

Figure 5. IoU as a function of the manipulated area (as percentage) for all three setups when changing the training dataset: CelebA-HQ (same as test; blue) or FFHQ (different from test; orange).

also on the background and, as the background takes most of the image, this region will impact most of the performance.

5.3. Generalization across generators

We evaluate to what extent localization methods trained on a particular generation method (*e.g.*, diffusion, GAN) generalize to samples produced by a different one. To this end we inpaint the CelebA-HQ dataset (using the same masks as before) with three other approaches: Latent Diffusion Model (LDM) [48], LaMa [55] and Pluralistic [62]. For the Repaint–P2 dataset we use a subset of 9k samples to match the samples from the other approaches (see Table 3). We train the `Patches` method in a fully supervised setting (scenario `C`) on each of the four datasets as well as combinations of those (using three out of the four datasets). The evaluation is carried on all four inpainted test sets.

The results for the 32 train–test combinations are given in Figure 6, while qualitative results are shown in Figure 7.

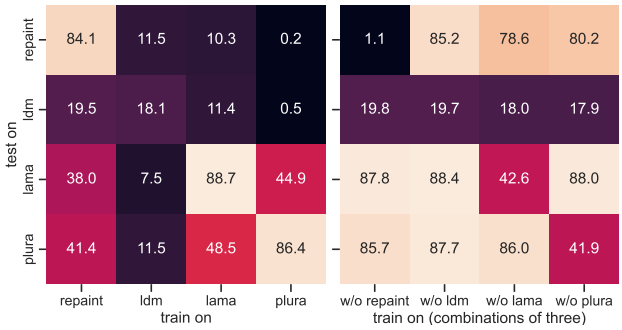

Figure 6. Localization performance (IoU) across four inpainting methods (Repaint, LDM, LaMa, Pluralistic) and their combinations. All four methods inpainted the same images from CelebA-HQ using the same masks. We used `Patches` in setup `C`.

We observe that localization works generally very well as long we test on data generated from the same model (main diagonal in the left plot). However, LDM is an exception: localization in LDM-manipulated images is more difficult since the inpainting is carried in the latent space and the decoding step "hides" the traces of the latent manipulation, akin to how image processing steps degrade detection performance [57].

When we evaluate on data coming from a different generator, the performance drops sharply (off-diagonal entries in the left plot). The transfer performance between LaMa and Pluralistic is still decent, presumably due to the particularities of the encoder–decoder approach. The diffusion model of Repaint is different from the two and makes the cross-generator transfer more challenging. Still, it appears that the transfer from diffusion to autoencoders and GANs (38.0% and 41.4%, respectively) is easier than the other way around (10.3% and 0.2%, respectively); a similar conclusion has been observed for detection [47].

Training on combinations of multiple datasets yields generally good performance on all the datasets involved at training (off-diagonal entries in the right plot). However, we do not observe a generalization benefit by using more types of generators at training (diagonal entries in the right plot *vs* off-diagonal entries in the left plot). For example, training on all generators but LDM yields an IoU of 19.7%, which is only marginally above 19.5%, what is achieved by training only on Repaint. For the other three generators, the performance is even slightly worse on combinations than the single best generator.

5.4. Performance on unseen datasets

In this section, we consider generalization in its most challenging form, by varying both the source dataset and the generation algorithm. Consequently, we evaluate on a

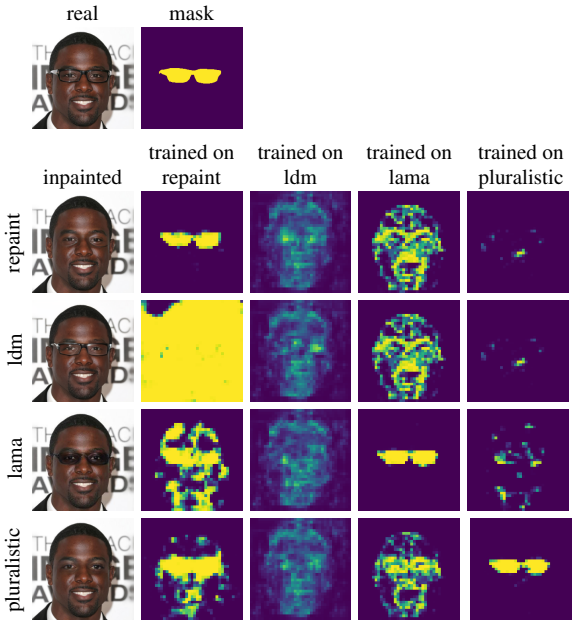

Figure 7. Qualitative results for the cross-generator evaluation using `Patches` trained in setup `C`. We observe the difficulty in generalization across generators (off-diagonal predictions) and the fact that local manipulations induced by LDM are challenging to identify (second image on the main diagonal).

different dataset, COCO Glide [21], which consists of 512 locally-edited images using a text-guided diffusion-based model. Additionally, we present results of five other existing localization methods [11, 21, 22, 35, 58], which were pretrained on different datasets, and compare their performance on our own Repaint–P2/CelebA-HQ, as well as on COCO Glide. For a comparison to `Patches`, we also fine-tune the PSCC method [35] in setup `C` on the Repaint–P2/CelebA-HQ data. The results are shown in Table 4.

We observe that the generalization performance is modest on either of the two datasets: the best out-of-domain performance on Repaint–P2/CelebA-HQ is 23.1%, obtained by TruFor, while on COCO Glide is 33.3%, obtained by PSCC. Even methods that have shown to generalize (TruFor [21]) or that have been trained specifically on diffusion images (HiFi-Net [22]) have difficulties on out-of-domain datasets.

`Patches` shows competitive results (second best in terms of IoU on COCO Glide), even if it was trained solely on faces. Interestingly, this is not the case for PSCC. While PSCC obtains top performance in-domain, on Repaint–P2/CelebA-HQ, it struggles to generaralize to COCO Glide. This behaviour suggests that overfitting is ocurring, which is not surprising given that the model capacity of PSCC (3.6M parameters) is an order of magnitude larger than the one of `Patches` (200k parameters).

	R-P2/CelebA		COCO Glide	
Method	IoU	PBCA	IoU	PBCA
MantraNet [58]	4.8	**81.9**	25.1	79.8
Noiseprint [11]	18.2	23.8	23.9	29.0
PSCC [35]	14.3	66.5	**33.3**	80.6
TruFor [21]	**23.1**	81.3	. 29.2	**81.4**
HiFi-Net [22]	0.0	81.0	2.6	3.2
Methods trained on Repaint–P2/CelebA-HQ in setup `C`				
PSCC [35]	89.0	98.8	13.3	18.4
`Patches`	84.5	98.7	30.8	64.8

Table 4. Evaluation of pretrained localization models on our Repaint–P2/CelebA-HQ and the COCO Glide dataset [21]. The grayed out results (`Patches` and PSCC on Repaint–P2/CelebA-HQ) are not directly comparable to those of other methods, since both `Patches` and PSCC are trained on Repaint–P2/CelebA-HQ. Qualitative results are available in the supplementary material.

6. Conclusions

In this paper, we investigate weakly-supervised localization in the context of diffusion-generated images of faces. We propose a framework and a dataset that allows to systematically explore the importance of different factors in model performance, such as: choice of detection method, level of supervision, dataset and type of generator used. We design a series of experiments that progressively modify the training assumptions and showed that, to a certain extent, detection of local manipulations can be performed weakly supervised, even in the most restrictive scenarios.

We summarize our findings: 1. The patch-based method consistently outperforms the other two approaches (explanations or attention) across multiple settings and metrics. 2. The detection performance in one of the weakly-supervised settings (image label & partial manipulations) is strong across all detection methods, suggesting that partially-manipulated images can be used for training deepfake classifiers. 3. Among the three types of factors (supervision, dataset, generator method), supervision seems to have a lesser impact (at least for the best performing method, `Patches`), while the generator impacts the most. 4. Localization of manipulations for latent diffusion models is very challenging even in the most optimistic scenario.

We believe that these findings can fuel research into weakly-supervised localization of deepfake manipulations with possible extensions to general-content images and to other types of local manipulations, such as face-swap, local enhancements or facial pose transfer obtained with DDPMs.

Acknowledgements. This work was supported in part by European Union's HORIZON-CL4-2021-HUMAN-01 research and innovation program under grant agreement No. 101070190 "AI4Trust".

References

[1] Darius Afchar, Vincent Nozick, Junichi Yamagishi, and Isao Echizen. MesoNet: A compact facial video forgery detection network. In *IEEE Workshop on Information Forensics and Security*, pages 1–7, 2018. 2

[2] Susmit Agrawal, Prabhat Kumar, Siddharth Seth, Toufiq Parag, Maneesh Singh, and R. Venkatesh Babu. SISL: Self-supervised image signature learning for splicing detection & localization. In *CVPRW*, pages 22–32, 2022. 2

[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 TIP*, 28(7):3286–3300, 2019. 2

[4] Aidan Boyd, Patrick Tinsley, Kevin W. Bowyer, and Adam Czajka. CYBORG: Blending human saliency into the loss improves deep learning-based synthetic face detection. In *WACV*, pages 6108–6117, January 2023. 3

[5] Lucy Chai, David Bau, Ser-Nam Lim, and Phillip Isola. What makes fake images detectable? Understanding properties that generalize. In *ECCV*, pages 103–120, 2020. 1, 3, 5

[6] Lu Chi, Borui Jiang, and Yadong Mu. Fast fourier convolution. *NeurIPS*, 33:4479–4488, 2020. 5

[7] Jooyoung Choi, Jungbeom Lee, Chaehun Shin, Sungwon Kim, Hyunwoo Kim, and Sungroh Yoon. Perception prioritized training of diffusion models. In *CVPR*, pages 11472–11481, 2022. 4

[8] François Chollet. Xception: Deep learning with depthwise separable convolutions. In *CVPR*, pages 1251–1258, 2017. 3

[9] Riccardo Corvi, Davide Cozzolino, Giada Zingarini, Giovanni Poggi, Koki Nagano, and Luisa Verdoliva. On the detection of synthetic images generated by diffusion models. In *ICASSP*, pages 1–5. IEEE, 2023. 1, 2

[10] Davide Cozzolino, Diego Gragnaniello, Giovanni Poggi, and Luisa Verdoliva. Towards universal GAN image detection. In *IEEE Visual Communications and Image Processing*, pages 1–5, 2021. 2

[11] Davide Cozzolino and Luisa Verdoliva. Noiseprint: A cnn-based camera model fingerprint. *IEEE Transactions on Information Forensics and Security*, 15:144–159, 2019. 8

[12] Hao Dang, Feng Liu, Joel Stehouwer, Xiaoming Liu, and Anil K Jain. On the detection of digital face manipulation. In *CVPR*, pages 5781–5790, 2020. 2, 3

[13] Sowmen Das, Md. Saiful Islam, and Md. Ruhul Amin. GCA-Net : Utilizing gated context attention for improving image forgery localization and detection. In *CVPRW*, pages 81–90, 2022. 2

[14] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat GANs on image synthesis. *NeurIPS*, 34:8780–8794, 2021. 1

[15] Brian Dolhansky, Joanna Bitton, Ben Pflaum, Jikuo Lu, Russ Howes, Menglin Wang, and Cristian Canton Ferrer. The deepfake detection challenge (DFDC) dataset. *arXiv preprint arXiv:2006.07397*, 2020. 2

[16] Xiaoyi Dong, Jianmin Bao, Dongdong Chen, Ting Zhang, Weiming Zhang, Nenghai Yu, Dong Chen, Fang Wen, and Baining Guo. Protecting celebrities from deepfake with identity consistency transformer. In *CVPR*, pages 9468–9478, 2022. 2

[17] Hany Farid. Lighting (in) consistency of paint by text. *arXiv preprint arXiv:2207.13744*, 2022. 2

[18] Hany Farid. Perspective (in) consistency of paint by text. *arXiv preprint arXiv:2206.14617*, 2022. 2

[19] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020. 1

[20] Diego Gragnaniello, Davide Cozzolino, Francesco Marra, Giovanni Poggi, and Luisa Verdoliva. Are GAN generated images easy to detect? A critical analysis of the state-of-the-art. In *ICME*, pages 1–6, 2021. 1, 2

[21] Fabrizio Guillaro, Davide Cozzolino, Avneesh Sud, Nicholas Dufour, and Luisa Verdoliva. TruFor: Leveraging all-round clues for trustworthy image forgery detection and localization. In *CVPR*, pages 20606–20615, 2023. 2, 8

[22] Xiao Guo, Xiaohong Liu, Zhiyuan Ren, Steven Grosz, Iacopo Masi, and Xiaoming Liu. Hierarchical fine-grained image forgery detection and localization. In *CVPR*, pages 3155–3165, 2023. 2, 8

[23] Alexandros Haliassos, Konstantinos Vougioukas, Stavros Petridis, and Maja Pantic. Lips don't lie: A generalisable and robust approach to face forgery detection. In *CVPR*, pages 5039–5049, 2021. 2

[24] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 3

[25] Yinan He, Bei Gan, Siyu Chen, Yichun Zhou, Guojun Yin, Luchuan Song, Lu Sheng, Jing Shao, and Ziwei Liu. ForgeryNet: A versatile benchmark for comprehensive forgery analysis. In *CVPR*, pages 4360–4369, 2021. 2

[26] Xuefeng Hu, Zhihan Zhang, Zhenye Jiang, Syomantak Chaudhuri, Zhenheng Yang, and Ram Nevatia. SPAN: Spatial pyramid attention network for image manipulation localization. In *CVPR*, pages 312–328, 2020. 1

[27] Minyoung Huh, Andrew Liu, Andrew Owens, and Alexei A Efros. Fighting fake news: Image splice detection via learned self-consistency. In *ECCV*, pages 101–117, 2018. 2

[28] Liming Jiang, Ren Li, Wayne Wu, Chen Qian, and Chen Change Loy. Deeperforensics 1.0: A large-scale dataset for real-world face forgery detection. In *CVPR*, pages 2889–2898, 2020. 2

[29] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of GANs for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017. 4

[30] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *CVPR*, pages 4401–4410, 2019. 4

[31] Hasam Khalid, Shahroz Tariq, Minha Kim, and Simon S. Woo. FakeAVCeleb: A novel audio-video multimodal deepfake dataset. In *NeurIPS Datasets and Benchmarks Track*, 2021. 2

[32] Cheng-Han Lee, Ziwei Liu, Lingyun Wu, and Ping Luo. MaskGAN: Towards diverse and interactive facial image manipulation. In *CVPR*, 2020. 4

[33] Ang Li, Qiuhong Ke, Xingjun Ma, Haiqin Weng, Zhiyuan Zong, Feng Xue, and Rui Zhang. Noise doesn't lie: Towards universal detection of deep inpainting. In *IJCAI*, pages 786–792, 2021. 1, 2

[34] Yuezun Li, Xin Yang, Pu Sun, Honggang Qi, and Siwei Lyu. Celeb-DF: A large-scale challenging dataset for deepfake forensics. In *CVPR*, pages 3207–3216, 2020. 2

[35] Xiaohong Liu, Yaojie Liu, Jun Chen, and Xiaoming Liu. Pscc-net: Progressive spatio-channel correlation network for image manipulation detection and localization. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(11):7505–7517, 2022. 8

[36] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *ICCV*, pages 3730–3738, 2015. 4

[37] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, pages 3431–3440, 2015. 3

[38] Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models. In *CVPR*, pages 11461–11471, June 2022. 1, 2, 3, 4

[39] Asad Malik, Minoru Kuribayashi, Sani M Abdullahi, and Ahmad Neyaz Khan. Deepfake detection for human face images and videos: A survey. *IEEE Access*, 10:18757–18775, 2022. 2

[40] Hannes Mareen, Dante Vanden Bussche, Fabrizio Guillaro, Davide Cozzolino, Glenn Van Wallendael, Peter Lambert, and Luisa Verdoliva. Comprint: Image forgery detection and localization using compression fingerprints. In *ICPR*, pages 281–299, 2022. 2

[41] Francesco Marra, Diego Gragnaniello, Luisa Verdoliva, and Giovanni Poggi. Do GANs leave artificial fingerprints? In *IEEE Multimedia Information Processing and Retrieval*, pages 506–511, 2019. 1, 2

[42] Ghazal Mazaheri, Niluthpol Chowdhury Mithun, Jawadul H. Bappy, and Amit K. Roy-Chowdhury. A skip connection architecture for localization of image manipulations. In *CVPRW*, pages 119–129, 2019. 2

[43] Yisroel Mirsky and Wenke Lee. The creation and detection of deepfakes: A survey. *ACM Computing Surveys*, 54(1):1–41, 2021. 1, 2

[44] Thanh Thi Nguyen, Quoc Viet Hung Nguyen, Dung Tien Nguyen, Duc Thanh Nguyen, Thien Huynh-The, Saeid Nahavandi, Thanh Tam Nguyen, Quoc-Viet Pham, and Cuong M Nguyen. Deep learning for deepfakes creation and detection: A survey. *Computer Vision and Image Understanding*, 223:103525, 2022. 1, 2

[45] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with CLIP latents. *arXiv preprint arXiv:2204.06125*, 2022. 1

[46] Danilo Rezende and Shakir Mohamed. Variational inference with normalizing flows. In *ICML*, pages 1530–1538. PMLR, 2015. 1

[47] Jonas Ricker, Simon Damm, Thorsten Holz, and Asja Fischer. Towards the detection of diffusion model deepfakes. *arXiv preprint arXiv:2210.14571*, 2022. 1, 2, 7

[48] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, pages 10684–10695, 2022. 1, 2, 4, 5, 7

[49] Andreas Rossler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. Faceforensics++: Learning to detect manipulated facial images. In *CVPR*, pages 1–11, 2019. 2, 3

[50] Chitwan Saharia, William Chan, Huiwen Chang, Chris Lee, Jonathan Ho, Tim Salimans, David Fleet, and Mohammad Norouzi. Palette: Image-to-image diffusion models. In *ACM SIGGRAPH*, pages 1–10, 2022. 1

[51] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. In *NeurIPS*, volume 35, pages 36479–36494, 2022. 1

[52] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-CAM: Visual explanations from deep networks via gradient-based localization. In *CVPR*, pages 618–626, 2017. 3

[53] Kaede Shiohara and Toshihiko Yamasaki. Detecting deepfakes with self-blended images. In *CVPR*, pages 18720–18729, 2022. 3

[54] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *ICML*, volume 37, pages 2256–2265, 2015. 1, 3

[55] Roman Suvorov, Elizaveta Logacheva, Anton Mashikhin, Anastasia Remizova, Arsenii Ashukha, Aleksei Silvestrov, Naejin Kong, Harshith Goka, Kiwoong Park, and Victor Lempitsky. Resolution-robust large mask inpainting with fourier convolutions. In *WACV*, pages 2149–2159, 2022. 4, 5, 7

[56] Luisa Verdoliva. Media forensics and deepfakes: An overview. *IEEE Journal of Selected Topics in Signal Processing*, 14(5):910–932, 2020. 1, 2

[57] Sheng-Yu Wang, Oliver Wang, Richard Zhang, Andrew Owens, and Alexei A. Efros. CNN-generated images are surprisingly easy to spot... for now. In *CVPR*, June 2020. 1, 2, 7

[58] Yue Wu, Wael AbdAlmageed, and Premkumar Natarajan. ManTra-Net: Manipulation tracing network for detection and localization of image forgeries with anomalous features. In *CVPR*, pages 9543–9552, 2019. 1, 8

[59] Ying Xu, Kiran Raja, Luisa Verdoliva, and Marius Pedersen. Learning pairwise interaction for generalizable deepfake detection. In *WACV*, pages 672–682, 2023. 3

[60] Ning Yu, Larry S. Davis, and Mario Fritz. Attributing fake images to GANs: Learning and analyzing GAN fingerprints. In *ICCV*, October 2019. 1, 2

[61] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *ICCV*, pages 3836–3847, 2023. 1

[62] Chuanxia Zheng, Tat-Jen Cham, and Jianfei Cai. Pluralistic image completion. In *CVPR*, pages 1438–1447, 2019. 4, 5, 7

[63] Peng Zhou, Xintong Han, Vlad I Morariu, and Larry S Davis. Two-stream neural networks for tampered face detection. In *CVPRW*, pages 1831–1839, 2017. 2, 3

[64] Peng Zhou, Xintong Han, Vlad I Morariu, and Larry S Davis. Learning rich features for image manipulation detection. In *CVPR*, pages 1053–1061, 2018. 1, 2