ONLY FOR YOU: DEEP NEURAL ANTI-FORWARDING WATERMARK PRESERVES IMAGE PRIVACY

Anonymous authors
Paper under double-blind review

ABSTRACT

In recent decades, messaging apps (e.g., Facebook Messager, Whatsapp, Wechat, Snapchat) have expanded exponentially, where a huge amount of private image sharing takes place daily. However, within these apps, the possible unauthorised or malicious image forwarding among users poses significant threats to personal image privacy. In specific situations, we hope to send private and confidential images (e.g., personal selfies) in an '*only for you*' manner. Given limited existing studies on this topic, for the first time, we propose the Deep Neural Anti-Forwarding Watermark (DeepRAFT) that enables media platforms to check and block any unauthorised forwarding of protected images through injecting *non-fragile and invisible* watermarks. To this end, we jointly train a DeepRAFT encoder and scanner, where the encoder embeds a confidentiality stamp into images as watermarks, and the scanner learns to detect them. To ensure that the technique is robust and resistant to tampering, we involve a series of stochastic concatenated data augmentations and randomized smoothing (a scalable and certified defense) towards both common image corruptions (e.g., rotation, cropping, color jitters, defocus blur, perspective warping, pixel noise, JPEG compression) and adversarial attacks (i.e., under both black and white box settings). Experiments on Mirflickr and MetFaces datasets demonstrate that DeepRAFT can efficiently and robustly imbue and detect the anti-forwarding watermark in images. Moreover, the trained DeepRAFT encoder and scanner can be easily transferred in a zero-shot manner even with a significant domain shift. We release our code and models to inspire studies in this anti-forwarding area at `link.available.upon.acceptance.`

1 INTRODUCTION

Over the past decades, online messaging apps, such as Facebook Messager, Whatsapp, Wechat, Snapchat, have been becoming essential tools for people's work and life. Billions of people use these platforms daily to send images to other users for the purpose of sharing life and business cooperation.

In some cases, a lot of shared images (e.g., private self-portrait and photos of non-discloseable business documents) are confidential. From the perspective of users, these private/confidential images are expected to be anti-forwarded to unauthorised receivers with the goal of privacy protection. For instance, a couple or close friends share private photos among themselves but do not intent for the media to be propagated outside the group. Currently, such privacy protections are mainly achieved by none-technological tools, such as business law Mantelero (2017) and personal trust Saeri et al. (2014). Existing data-privacy related studies Beigi & Liu (2020); Jiang et al. (2021) mainly aim to avoid the sensitive/private information leaking or abuse in the usage or publication of data, which can be deemed as the data privacy conflict between the data provider and the platform who utilize these data to train machine learning models. In contrast, our work mainly focuses on the privacy protection for data transmissions among different users on a media platform.

To prohibit unauthorized image transmissions among users, for the first time, we propose the Deep Neural Anti-Forwarding Watermark (DeepRAFT) as shown in Figure 1. Specifically, when a user intends to share an image to others on a media platform, he/she can choose whether turning on or off DeepRAFT. If DeepRAFT is turned on as shown by the green pipeline, the DeepRAFT encoder will add imperceptible watermarks on the protected images, accordingly any unexpected further

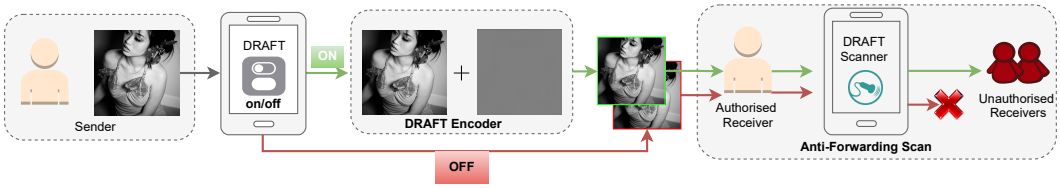

Figure 1: Pipelines of deep neural privacy-preserving watermark framework. Sample image is from MirFlickr dataset Huiskes et al. (2010).

transmissions to unauthorised receivers will be stopped by the DeepRAFT scanner. In contrast, if DeepRAFT is tuned off (i.e., the red pipeline), images will be directly sent to the receiver and they can be forwarded by the receiver to other parties. Although plenty of studies have worked on image watermarking Zhu et al. (2018); Tancik et al. (2020), none of them is for anti-forwarding purpose.

With the goal of anti-forwarding, we design an end-to-end structure of DeepRAFT as shown in Figure 2. Specifically, the encoder is structured to embed imperceptible/subtle watermarks into protected images, where the imperceptibility is regularized by a residual regularization loss and LPIPS perceptual loss Zhang et al. (2018). Meanwhile, the DeepRAFT scanner is designed as a binary classifier that learns to distinguish watermarked and non-watermarked images. More importantly, to make DeepRAFT robust and resilient towards possible image editing/corruptions (e.g., rotation, cropping, color jitters, defocus blur, perspective warping, pixel noise, JPEG compression), we involve a stochastic concatenation based data augmentation during training. Besides, the corruptions/edit may come from the malicious party trying to circumvent DeepRAFT to propagate private media. This inspires us going step further to improve the adversarial robustness of the scanner through randomized smoothing due to its scalability. We validate our adversarial robustness on five white- and black-box attacks, including auto-attack Croce & Hein (2020), Auto-PGD Croce & Hein (2020), square-attack Andriushchenko et al. (2020), PGD and FGSM.

The main contributions of this paper can be summarized as follows.

- To the best of knowledge, this is the first study that investigates the anti-forwarding problem to protect personal data privacy when sharing images on media platforms. This opens up a plethora of novel research questions for data privacy that have not yet been studied in machine learning.
- We propose DeepRAFT, an end-to-end training framework with 1) an encoder that adds imperceptible watermark on protected images, and 2) a scanner that learns to detect the watermark. Moreover, we jointly train the encoder and the scanner with introducing a stochastic concatenation of the data augmentations that mimics both electronic and physical image corruptions in the real-world. Moreover, we take a step further to preemptively take care of the robustness towards malicious adversarial attacks by training a randomly smoothed detector.
- Extensive experiments on Mirflickr and MetFaces datasets showcase that DeepRAFT can not only accurately detect whether an image should be anti-forwarded, but also be substantially robust towards common image corruptions and black- & white-box adversarial attacks. Moreover, we surprisingly find that our trained DeepRAFT encoder and scanner can be transferred in a zero-shot manner where significant domain shift exists.

2 Related Work

The closely related studies to this paper are about hiding data in an image, which includes image watermarking, steganography and adversarial attacks. Moreover, given our aim is to protect the users' data privacy on a media platform, we also show the difference between this work and previous studies about data privacy.

Image Watermarking. Research along adding watermarks on images has a long history. Earlier works Braudaway (1997) mainly focus on improving the robustness towards possible image manipulations (e.g., JPEG compression) and human imperceptibility. Further studies thereafter explore how to improve the invisibility of embedded watermark through log-polar frequency domain. There are also efforts Nakamura et al. (2006); Pramila et al. (2012) on improving the effectiveness of added watermarks in the wild, such as using on mobile apps. In addition, to better alleviate the impact from re-photography (i.e., with perspective warping), there are studies that specifically investigate

the printer-camera Pramila et al. (2018) and display-camera Fang et al. (2018); Wengrowski et al. (2016) transformation. However, most of previous approaches in watermarking are mainly based on hand designed pipeline.

Although there are approaches Tancik et al. (2020); Zhu et al. (2018); Sharma et al. (2019); Wang et al. (2021) in recent years that learn how to insert watermarks automatically in an end-to-end manner, they mainly focus on information transmission instread of anti-forwarding. For instance, Zhu et al. (2018) proposed to hide specific messages in images, from which the model learns how to reconstruct the original message. Tancik et al. (2020) proposed to hide arbitrary hyperlink bit-strings into images, thereby the model can recognise this hyperlink with the goal of information transformation. Wang et al. (2021) investigated how to generate fake watermarked images for circumvention, which is still different from our anti-forwarding goal. These approaches can be converted to do anti-forwarding, but they are not designed explicitly to do so, thus making our encoder design more efficient as shown in Section 3.2. There are also studies in steganography that hide data in images using encoder-decoder based deep learning models Baluja (2017); Hayes & Danezis (2017); Tang et al. (2017); Wengrowski & Dana (2019). Many of them assume perfect digital image transmission, thus the possible image perturbation (e.g., random noise) and editing (e.g., rotation, cropping, color jitters) may cause the well trained model less efficient.

Compared with previous studies, our work investigates more comprehensive types of image corruptions/perturbation as shown in Section 3.4. Specifically, StegaStamp Tancik et al. (2020) did not consider image rotation and cropping. HiDDeN Zhu et al. (2018) examines most of our corruptions yet without color jitters and physical perspective change. It is more noteworthy that *none* of previous studies considers the case of adversarial attacks by malicious actors. In contrast, we carefully investigate the adversarial robustness of DeepRAFT as shown in Section 3.5. Lastly, compared to many image watermarking studies for embedding random messages Tancik et al. (2020); Zhu et al. (2018) into images during encoder training, our anti-forwarding encoder structure does involve such randomness; this makes our pipeline relatively easier to train.

Adversarial Attack. In computer vision, adversarial attack Szegedy et al. (2014) aims to find out particular pixel perturbations that mostly degrade the performance of well trained models within restricted neighborhoods of original images. For instance, the projected gradient descent (PGD) Madry et al. (2018b) attack starts from a random perturbation and iteratively updates it to minimise the accuracy of the original model. Although our approach is also adding pixel perturbations on images, we take a different and positive view, i.e., we aim to protect the user privacy instead of discovering the vulnerability.

Besides that, we also maintain the robustness of DeepRAFT against adversarial attacks. Namely, assuming our DeepRAFT has been deployed, the attacker may generate adversarial examples that escape from the DeepRAFT scanner, thereby forwarding the anti-forwarded images. There are plenty of methods for improving adversarial robustness, such as adversarial training Shafahi et al. (2019), interval bound propagation Mirman et al. (2018); Zhang et al. (2019) and randomized smoothing Cohen et al. (2019). In this paper we utilize randomized smoothing due to its scalibility towards high dimensional dataset, e.g., Mirflickr with 400×400 dimension.

Data Privacy. There have been many studies on protecting data privacy Beigi & Liu (2020); Jiang et al. (2021) in machine learning. However, most of them focus on the problem of possible exposes of user information when a dataset is used for model training Liu et al. (2021). Namely, such data privacy aims to solve the issue that malicious parties may use deployed deep models to retrieve desired sensitive information. For instance, Zhang et al. (2021) proposed to proactively transfer original data into adversarial data with quasi-imperceptible perturbations before releasing them. There are also plenty of studies on differential privacy Dwork et al. (2014); Abadi et al. (2016) of deep learning model, which aims to publicly share a dataset while preserving information about individuals in the dataset. In summary, previous privacy protection approaches aim at a different scope compared with our case, since our focus is on the privacy protection among users when sharing images on media platforms. Although there are approaches (e.g., iPrivacy Yu et al. (2016)) that try to identify sensitive information when users sharing images, they are based on a fixed and predefined rule for identifying sensitivity and determining whether to allow forwarding thereafter. However, our approach can choose arbitrary images to be anti-forwarded, thus being more general and flexible.

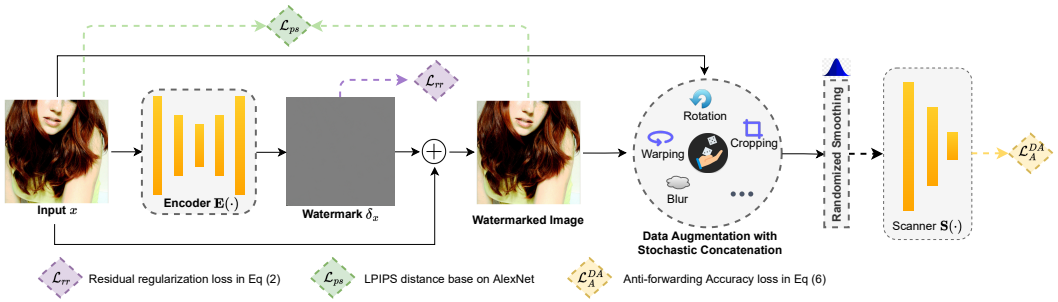

Figure 2: General training framework of DeepRAFT.

3 DEEP NEURAL ANTI-FORWARDING WATERMARK (DEEPRAFT)

First, we formally define the anti-forwarding problem, since it has rarely studied in literature. Thereby, we design an encoder-scanner framework for DeepRAFT that enables adding imperceptible watermark (on arbitrarily selected images) and being detected by the scanner in an end-to-end manner. During training, we involve a stochastic concatenation of a series of differentiable image corruptions in order to mimic both digital image editing and real-world transmissions, e.g., color jitters, cropping, defocus blur, random noise, rotation, perspective warping that simulates physical displaying-imaging pipeline, and JPEG compression. Besides common image corruptions, we go step further to consider the robustness towards adversarial attacks via adversarial training Madry et al. (2018a) and randomized smoothing Cohen et al. (2019).

3.1 ANTI-FORWARDING PROBLEM SETTING

The training framework of DeepRAFT is shown in Figure 2. In general, the goal of anti-forwarding is to distinguish non-watermarked images and watermarked images (i.e., generated by DeepRAFT encoder) without degrading/distorting the image. A more specific problem setting is defined as follows.

Definition 1 (Anti-Forwarding) *Given an image $x \in \mathcal{X}$, the DeepRAFT encoder $\mathbf{E} : \mathcal{X} \rightarrow \mathcal{X}'$ generates a watermarked image $x + \delta_x, \delta_x \leq \alpha$, where α is the threshold for visibility. Thereby, the DeepRAFT scanner $\mathbf{S} : \mathcal{X} \rightarrow \mathcal{Y}$ learns to distinguish x and $x + \delta_x$, where $\mathcal{Y}$ is a binary space. This therefore enables $x + \delta_x$ to be anti-forwarded but x not.*

To this end, $\mathbf{E}$ and $\mathbf{S}$ are optimized jointly to minimize the loss function $\mathcal{L}(\mathbf{E}, \mathbf{S})$:

$$\mathcal{L}(\mathbf{E}, \mathbf{S}) = \mathbb{E}_{x \sim \mathcal{X}} \Big[\underbrace{\mathcal{L}_P(x, \mathbf{E}(x))}_{\textbf{Watermark Imperceptibility}} + \underbrace{\mathcal{L}_A(\mathbf{S}(x), \mathbf{S}(\mathbf{E}(x))}_{\textbf{Anti-Forwarding Accuracy}} \Big], \mathbf{E}(x) = x + \delta_x, \tag{1}$$

where $\mathcal{L}_P(\cdot)$ is the imperceptibility loss that minimizes the visibility of the added watermark δ_x. $\mathcal{L}_A(\cdot)$ is the anti-forwarding accuracy loss that enables the scanner $\mathbf{S}$ to distinguish x and $x + \delta_x$. Note that the encoder does not involve a random vector for generating watermark δ_x as what previous approaches have done. This makes our pipeline relatively much easier to train. More details of the training loss and model structure are shown as follows.

3.2 MODEL STRUCTURE AND LOSS DESIGN

Encoder Model. The encoder $\mathbf{E}$ is a U-Net Ronneberger et al. (2015) based model that generates a watermark δ_x solely based on the original image x (e.g., with dimension $3 \times 400 \times 400$). The output (i.e., watermarked image $x + \delta_x$) from $\mathbf{E}$ is a $3 \times 400 \times 400$ tensor as well. Accordingly, the visibility of δ_x can be controlled by simultaneously regularizing δ_x from different perspectives (e.g., RGB space, YUV space and deep feature space) as shown below.

Watermark Imperceptibility Loss. The watermark imperceptibility loss $\mathcal{L}_P(\cdot)$ aims to enforce minimal perceptual corruptions from δ_x. To this end, we introduce two loss functions, viz., 1) the

residual regularization loss $\mathcal{L}_{rr}(\cdot)$ that generally minimizes the magnitude of δ_x; 2) the perceptual similarity loss $\mathcal{L}_{ps}(\cdot)$ in deep feature space that computes the average learned perceptual image patch similarity (LPIPS) distance Zhang et al. (2018) between x and $x + \delta_x$. Specifically, $\mathcal{L}_{rr}(\cdot)$ is calculated as the sum of l_1 distance in RGB space and l_2 distance in YUV space Levin et al. (2004), where the transformation $\mathcal{T}_{yuv}$ separates the color (Y channel) and brightness (U and V channels) of a source image. In addition, the LPIPS distance is calculated by a weighted Euclidean distance between *deep features* of images, where features are obtained from ImageNet-pretrained AlexNet Krizhevsky et al. (2012) and the weights are fit to align with human perceptual similarity judgments. In sum, $\mathcal{L}_P(\cdot)$ is formulated as:

$$\begin{aligned} \mathcal{L}_P(x, \mathbf{E}(x)) &= \mathcal{L}_{rr}(x, \mathbf{E}(x)) + \mathcal{L}_{ps}(x, \mathbf{E}(x)), \\ \mathcal{L}_{rr}(x, \mathbf{E}(x)) &= \|\mathbf{E}(x) - x\|_1 + \|\mathcal{T}_{yuv}(\mathbf{E}(x)) - \mathcal{T}_{yuv}(x)\|_2 \end{aligned} \tag{2}$$

Details of calculating $\mathcal{L}_{ps}(x, \mathbf{E}(x)$ follow the implementation [1] in Zhang et al. (2018).

Scanner Model. The scanner $\mathbf{S}$ is a network trained to distinguish the original image x and watermarked image $x + \delta_x = \mathbf{E}(x)$. To consider both digital and physical image corruptions (as shown in Section 3.4) that may appear, a stochastic concatenation of different differentiable image transformations are introduced. After that, the transformed images are fed into the scanner network (i.e., a set of convolution and fully connected layers) that outputs a sigmoid activated value $y = \mathbf{S}(x), y \in \mathcal{Y}$.

Anti-Forwarding Accuracy Loss. The anti-forwarding accuracy loss $\mathcal{L}_A(\cdot)$ utilizes binary cross entropy $\mathcal{I}(\cdot)$, viz.,

$$\mathcal{L}_A(\mathbf{S}(x), \mathbf{S}(\mathbf{E}(x))) = \mathcal{I}(\mathbf{S}(x), y_{nw}) + \mathcal{I}(\mathbf{S}(\mathbf{E}(x)), y_w), \tag{3}$$

where y_{nw} and y_w are labels for non-watermarked and watermarked images, respectively. The above definition and settings could enable the basic anti-forwarding ability (i.e., watermark being invisible and distinguishable) of DeepRAFT. However, in real-world scenarios, there may exist digital image editing (e.g., rotation, cropping, color jitters) or natural image corruptions (e.g., random noise, perspective change when taking a photo), which would degrade the performance of DeepRAFT. Moreover, the uses or malicious party of a media platform may apply transformations/perturbations to remove embedded yet unknown watermarks. To understand and mitigate their impact, we investigate such robustness as follows.

3.3 Threat Model

To comprehensively take care of the robustness of DRAFT, we consider that the threat comes from two aspects, i.e., common image corruptions and adversarial attacks. To this end, following most deployed machine learning model's settings, we assume that the media platform keeps the watermark encoder as secrete, i.e., the users can query the encoder API yet without accessibility of its internal model. Moreover, the platform can introduce random keys and abnormal query checking to avoid possible malicious encoder model theft.

Under such setting, the threat of watermarking removal mainly comes from the scanner perspective. To improve the robustness on common image corruptions/editing, we utilize a stochastic concatenation based image augmentation (i.e., illustrated in Section 3.4). To increase the adversarial robustness, we involve randomized smoothing (as shown in Section 3.5) to train a smoothed scanner. Such robustness operations together with other model settings are kept unknown for external parties.

3.4 Robustness Towards Common Image Corruptions/Editing

To investigate the robustness of DeepRAFT, we consider 7 common image corruptions (i.e., color jitters (CLJ), cropping (CRP), defocus blur (DFB), random noise (RDN), rotation (RTT), perspective warping (PSW), JPEG compression (JPEG)) and the concatenation of all image corruptions (CCA). Compared to previous studies Zhu et al. (2018); Tancik et al. (2020), our investigated image corruptions are more comprehensive to reflect the real world scenario as analyzed in Section 2. Examples of all image corruptions are shown in Figure 6. Specifically, the rotation layer rotates the image by angle. The cropping layer crops a random portion of image and resizes it to the original size. The color jitters layer randomly changes the brightness, contrast, saturation and hue of an image.

[1] https://github.com/richzhang/PerceptualSimilarity

The perspective warp layer performs a random perspective transformation of the given image with a given probability, which aims to mimic the displaying-and-imaging scenario that people take a photo of a displayed anti-forwarded image. The defocus blur layer reflects the inaccurate autofocus when displaying-and-imaging happens. The random noise layer adds a random Gaussian noise (with standard deviation σ) on a source image. The JPEG compression simulates the change of digital image storing format that introduces numerical corruptions, which is commonly used to reduce the amount of data that needs to record an image. Specifically, JPEG compresses an image by calculating the discrete cosine transform of each 8×8 block in the image and quantizing the obtained coefficients as their nearest integers. Implementation of JPEG compression follows from DiffJPEG Mirman et al. (2018). Other corruptions are mounted on transforms in Torchvision Marcel & Rodriguez (2010).

Note that a plain training cannot generalize to these image corruptions. To improve such generalization towards different corruptions, we introduce a stochastic contention process $\mathcal{T}_c(x, p)$ that sequentially manipulates an image x using each corruption with a probability p. Such stochastic concatenation of different corruptions is motivated by the fact that training is possibly biased towards a specific corruption. Since we found that the plain training could naturally resist the perturbation from random Gaussian noise, we only utilize the other 6 image corruptions shown in Figure 6 besides random noise in the training stage. Therefore, the corresponding anti-forwarding accuracy loss $\mathcal{L}_A^{DA}(\cdot)$ with data augmented training becomes

$$\mathcal{L}_A^{DA}(\mathbf{S}(x), \mathbf{S}(\mathbf{E}(x))) = \mathcal{L}_A(\mathbf{S}(x), \mathbf{S}(\mathbf{E}(x))) + \mathcal{I}(\mathbf{S}(\mathcal{T}_c(x, p)), y_{nw}) + \mathcal{I}(\mathbf{S}(\mathcal{T}_c(\mathbf{E}(x), p)), y_w). \quad (4)$$

3.5 Smoothed Scanner To Resist Adversarial Attacks

Besides the common image corruptions mentioned above, we also consider the impact from adversarial attack. To improve the adversarial robustness of scanner, we utilize randomized smoothing due to its scalibility toward high dimensional dataset. Specifically, the scheme of training a smoothed scanner $\tilde{\mathbf{S}}(x)$ is: first, we train the scanner $\mathbf{S}(x)$ with Gaussian data augmentation on input image x at variance σ^2; then we utilize $\mathbf{S}(x)$ to create a new, "smoothe" scanner $\tilde{\mathbf{S}}(x)$; finally $\tilde{\mathbf{S}}(x)$ returns the prediction which $\mathbf{S}(x)$ is most likely to return when x is corrupted by isotropic Gaussian noise with variance σ^2. In sum, the smoothed scanner can be formulated as

$$\mathbb{E}_{x \in \mathcal{X}} \tilde{\mathbf{S}}(x) = \mathbb{E}_{x \in \mathcal{X}} \left[\mathbf{S}(x + \epsilon) \right], \epsilon \in \mathcal{N}(0, \sigma^2 \mathbb{I}). \quad (5)$$

Alternatively stated of Eq. (5), the smoothed scanner $\tilde{\mathbf{S}}(x)$ classifies whether x is a watermarked image under the sampling of $\mathbf{S}(x + \epsilon)$. Accordingly, an adversarial attack $x + \delta_x$ towards $\mathbf{S}(\cdot)$ is less harmful for $\tilde{\mathbf{S}}(\cdot)$, because $\tilde{\mathbf{S}}(\cdot)$ does not only focus on $x + \delta_x$ itself, but also on its weighted neighborhood. This enables $\tilde{\mathbf{S}}(\cdot)$ to eliminate/alleviate the impact from adversarial perturbations.

3.6 The Training Scheme of DeepRAFT

In sum, the training loss for DeepRAFT is shown as

$$\begin{aligned} \mathcal{L}(\mathbf{E}, \mathbf{S}) = \mathbb{E}_{x \sim \mathcal{X}} \{ & \lambda_p \left(\|\mathbf{E}(x) - x\|_1 + \|\mathcal{T}_{yuv}(\mathbf{E}(x)) - \mathcal{T}_{yuv}(x)\|_2^2 \right) + \mathcal{L}_{ps}(x, \mathbf{E}(x)) \\ & + \lambda_a \left[\mathcal{I}(\mathbf{S}(x + \epsilon), y_{nw}) + \mathcal{I}(\mathbf{S}(\mathbf{E}(x + \epsilon)), y_w) \right] \\ & + \lambda_{aug} \left[\mathcal{I}(\mathbf{S}(\mathcal{T}_c(x, p)), y_{nw}) + \mathcal{I}(\mathbf{S}(\mathcal{T}_c(\mathbf{E}(x)), p), y_w) \right] \}, \end{aligned} \quad (6)$$

where λ_p and λ_{ps} are weights for two watermark imperceptibility losses in Eq. (2); λ_a and λ_{aug} are weights for accuracy loss and corruption based augmented accuracy loss, respectively in Eq. (4). In summary, for the training loss in Eq. (6), there are several settings as shown below that we find useful for faster convergence. (1) The training is more sensitive to image variation of azimuth, i.e., rotation and perspective warping. For successful training, the strength of such data augmentation should be increased gradually. (2) JPEG compression is naturally non-differentiable. We involve an approximate yet differentiable JPEG compression Shin & Song (2017) during training. (3) We find that using L_1 distance loss term in $\mathcal{L}_{rr}(x, \mathbf{E}(x))$ is important for perceptual similarity.

4 Experiments

4.1 Experiment Settings. We evaluate the effectiveness of DeepRAFT on two commonly used benchmark datasets, MirFlickr Huiskes & Lew (2008) and MetFaces Karras et al. (2020). We start

Table 2: Objective evaluations (PSNR, SSIM and LPIPS) on watermark invisibility. ↑ and ↓ indicate a larger and smaller value is preferred, respectively. Baseline results are from Tancik et al. (2020).

	PSNR ↑	SSIM ↑	LPIPS ↓
Baluja Baluja (2017)	24.61	0.926	0.256
Hidden[native] Zhu et al. (2018)	31.07	0.940	0.070
Hidden Zhu et al. (2018)	24.55	0.775	0.202
LFM Wengrowski & Dana (2019)	20.89	0.910	0.315
StegaStamp Tancik et al. (2020)	27.25	0.927	0.194
DeepRAFT [Plain]	**40.7188** $\pm$ 0.9228	**0.9968** $\pm$ 0.0003	**0.0006** $\pm$ 0.0001
DeepRAFT [Plain+DA+RS]	*32.8442* $\pm$ 0.3287	*0.9758* $\pm$ 0.0019	*0.0096* $\pm$ 0.0008

with the training and evaluation on MirFlickr dataset, where the evaluation has considered 7 common image corruptions (i.e., rotation, cropping, color jitters, perspective warp, random noise, defocus blur and JPEG compression) and their concatenation. Moreover, we also evaluate the robustness of the well trained DeepRAFT scanner towards adversarial attacks, where both multiple black-box and white-box attacks are involved. More importantly, we evaluate the transfer-ability by directly applying the DeepRAFT encoder and scanner trained on MirFlickr to METAFACES dataset (i.e., sampled with resolution 400×400) in a zero-shot manner. We train our model on NVIDIA-A100 GPU with batchsize 64 and training step 2×10^5. The weights (i.e., λ_p, λ_{ps}, λ_a and λ_{aug}) in Eq. 6 are intially set to be 1 and we found they work well as shown below. The weight λ_{aug} is linearly increased from 0 to 1, starting at step $5e^3$ and ending at $5e^4$. We set up learning rate, dataset split, and optimizer related settings, according to protocols in StegaStamp Tancik et al. (2020). In evaluating the invisibility of added watermark, we provide both objective evaluations and demo showcases, where the objective evaluation metrics are based on peak signal-to-noise ratio (PSNR), structural similarity index metric (SSIM) and LPIPS distance (i.e., as shown in Eq. 2).

4.2 Anti-Forwarding Detection Accuracy. First, we evaluate the accuracy of DeepRAFT on original images from MirFlickr dataset across 2560 images. Note that existing works Baluja (2017); Zhu et al. (2018); Wengrowski & Dana (2019); Tancik et al. (2020) on image watermarking are mainly for message hiding. For the anti-forwarding purpose, the message reconstruction accuracy can be converted to represent as the accuracy of binary anti-forwarding classification accuracy, because a 100% message reconstruction can represent a successful anti-forwarding detection. The results of baselines are implemented by Tancik et al. (2020).

Table 1: Image watermark detection accuracy.

Method	**Accuracy** ↑
Baluja Baluja (2017)	51%
Hidden Zhu et al. (2018)	65%
LFM Wengrowski & Dana (2019)	93%
StegaStamp Tancik et al. (2020)	99%
DeepRAFT [Plain]	100%
DeepRAFT [Plain+DA]	100%
DeepRAFT [Plain+DA+RS]	99.98%

The results and comparisons are shown in Table 1, where abbreviations are explained as: DA (with data augmention), RS (with randomized smoothing). From results, we achieve 100% accuracy on plain training and 99.98% plain training with DA and RS, which suggests an outstanding performance of anti-forwarding detection on clean images. Our results are better than all the evaluated baselines, expecially for Baluja (2017) and Zhu et al. (2018). Moreover, comparisons among our methods indicate that introducing data augmentation and randomized smoothing has limited impact on the accuracy on clean data. Besides that, our approach also outperforms existing work from the watermark imperceptibility perspective, which is illustrated as follows.

4.3 Watermark Invisibility Evaluations.

To evaluate and compare the invisibility of added watermark, we use three metrics, viz., PSNR, SSIM and LPIPS distance. The results and comparisons are shown in Table 2, where our approach consistently outperforms previous baselines. Especially on LPIPS distance, our results are significantly smaller than that from previous baselines. However, compared with the invisibility of plain training (i.e., 40.7188 of PSNR, 0.9968 of SSIM and 0.0006 of LPIPS), adding randomized smoothing and data augmentation did slightly degrade the invisibility, but the corresponding scores (i.e., 32.8442 of PSNR, 0.9758 of SSIM and 0.0096 of LPIPS) still outperform previous approaches.

This would be contributed by three reasons. 1) We use a random message free design of the encoder structure, which makes the training with less randomness, thus being easier to converge. 2) The ob-

Table 3: Robustness towards APGD, square attack and auto-attack. Values represent accuracy percentage from 0 to 100.

	Auto-PGD (White-box)					Square-Attack (Black-box)					Auto-Attack (Adaptive)				
Strength ϵ	$\frac{2}{255}$	$\frac{4}{255}$	$\frac{8}{255}$	$\frac{16}{255}$	$\frac{32}{255}$	$\frac{2}{255}$	$\frac{4}{255}$	$\frac{8}{255}$	$\frac{16}{255}$	$\frac{32}{255}$	$\frac{2}{255}$	$\frac{4}{255}$	$\frac{8}{255}$	$\frac{16}{255}$	$\frac{32}{255}$
Plain	0	0	0	0	0	84.38	15.66	0	0	0	0	0	0	0	0
RS(σ=0.25)	86.80	58.46	20.44	0.40	0	99.22	99.22	92.23	90.70	68.16	86.80	58.55	22.61	21.48	17.77
RS(σ=0.5)	97.04	91.76	71.22	50.03	6.60	100	100	99.98	99.26	91.12	97.04	91.76	72.54	50.25	5.43
RS(σ=0.75)	94.66	91.67	79.97	50.26	49.90	100	100	98.48	99.23	96.18	94.66	91.57	80.00	50.28	50.00
DA+RS (σ=0.25)	93.96	33.85	8.17	5.27	5.27	99.69	98.98	95.12	84.78	62.98	94.89	84.10	72.40	76.45	63.63

jective of scanner (i.e., binary classification) is much simpler than previous message reconstruction, which simplifies the training pipeline as well. 3) The LPIPS distance explicitly appears as a loss function, hence the performance is improved more significantly than the other two metrics. 4) The training based on DA and RS makes the training more difficult, thus sacrificing some watermark invisibility to balance the accuracy loss.

We also compare the difference between original images and their corresponding watermarked image through demo showcases. One demo is shown in Figure 3, but more can be seen in Appendix. From the figure, very limited difference can be observed. Moreover, the add watermarks can reflect the semantics contained in the corresponding original images. This suggests that DeepRAFT learns how to smartly hide the watermark according to the semantic representation. To further verify the robustness of DeepRAFT, we evaluate the performance on different image corruptions and adversarial attacks as follows.

Clean Watermark Watermarked

Figure 3: Demo of watermark invisibility.

4.4 Robustness Towards Adversarial Attack. We investigate the robustness of DeepRAFT toward five adversarial attacks (i.e., auto-PGD Croce & Hein (2020), square attack Andriushchenko et al. (2020), auto-attack Croce & Hein (2020), PGD, FGSM) and random Gaussian noise. All attacks are based on L_{inf} setting with $\epsilon \in \{2/255, 4/255, 8/255, 16/255, 32/255\}$. We compare the model by plain training with that trained by randomized smoothing. In randomized smoothing, we use standard deviation $\sigma \in \{0.25, 0.5, 0.75\}$[2].

The results and comparisons are shown in Table 3 and 5 (see appendix). In general, the model using plain training setting is quite vulnerable to adversarial attacks, viz., the accuracy degrades to 0 on most cases. In contrast, with utilizing randomized smoothing, the robustness of DeepRAFT has been significantly improved. Specifically, we also have several interesting findings. 1) The results with randomized smoothing on square attack show some robustness, viz., 84.38% and 15.66% on $\epsilon = \frac{2}{255}$ and $\epsilon = \frac{4}{255}$, respectively. However, when ϵ keeps increasing, the accuracy decreases to 0 as well. This is different from other two attacks, where the accuracies under auto-PGD and auto-attack are always 0. This indicates that the black box attack in this scenario is weaker than the gradient based PGD attack and adaptive auto-attack. 2) The watermark detection accuracy on each attack consistently decreases with the increase of ϵ. This aligns with the fact that stronger attack deliveries stronger performance degradation. 3) Auto-attack is the ensemble of auto-PGD and square attacks, thus the performance mostly follows the worse one of the two. 4) Auto-PGD attack is worse than canonical PGD attack when comparing Table 3 and Table 5. This is mainly caused by the difference of steps, where auto-PGD utilizes an adaptive step size (i.e., calculated by $ceil(log_2(400)) = 9$), while PGD uses a step size 40. 5) Both the plain trained and randomized smoothing trained model are robust to random Gaussian noise, which suggests the very limited threat posed by random pixel perturbations to the security of DeepRAFT.

4.5 Robustness Towards Common Image Corruptions. The robustness towards common image corruptions (i.e., rotation, cropping, color jitters, perspective warp, random noise, defocus blur and JPEG compression) are evaluated on each corruption independently and their concatenation as a whole. The results are shown in the first two rows in Table 4. Specifically, the color jitter randomly

[2]Models will be provided in code release. We find that the training does not converge using $\sigma = 1$.

Table 4: Evaluation on common image corruptions/editing and *zero-short transfer*.

	Clean	CLJ	DFB	CRP	RTT	PSW	JPEG	CCA
MirFlickr [Plain]	100%	48.98%	49.90%	52.23%	50.24%	48.78%	53.08%	47.68%
MirFlickr [Plain+DA]	99.98%	99.94%	99.98%	99.96%	99.44%	98.96%	99.76%	97.80%
MirFlickr [Plain+DA+RS]	99.83%	99.46%	99.81%	88.68%	98.62%	99.73%	99.83%	89.11%
MetFaces Transfer Plain + DA	99.95%	99.91%	100%	100%	99.71%	100%	99.87%	97.37%

changes the brightness, contrast, saturation and hue in the scale from 0 to 0.4; the defocus blur utilizes a Gaussian kernel with size $(3, 7)$ and sigma $1, 3$; the cropping randomly selects a portion (scaled from 0.5 to 1) and upsampled to the original dimension 400×400; the rotation degree is randomly sampled from $[-45, 45]$; the perspective warping distorted an image with a random distortion scale sampled from $[0, 0.4]$; the JPEG compression follows the implementation from DiffJPEG Mirman et al. (2018) with quality uniformly sampled from $50, 100$. The random dropout of each image corruption during training is 25%.

In general from Table 4, although different image corruptions are added, all detection accuracies are larger than 98%. This suggests that DeepRAFT scanner can still robustly detect whether a corrupted image contains watermark that is added by DeepRAFT encoder. In particular, compared to other image corruptions, DeepRAFT is more sensitive toward rotation with accuracy 99.43% and perspective warping with accuracy 98.96%. When all image corruptions are concatenated together, the accuracy drops to 97.80%, which are reasonable due to their impact accumulation. Same phenomenon is also observed when DA and RS are both involved.

4.5 Zero-Shot Transfer. We investigate the transferablity of DeepRAFT by directly applying models trained on MirFlickr dataset to MetFaces dataset in a *zero-short manner*. The example images from MetFaces dataset are shown in Figure 4(b) in Appendix. Since the MetFaces dataset is collected from art paintings, which has a clear distribution shift with MirFlickr dataset in Figure 4(a). There are totally 1336 images (preprocessed to dimension 400×400) in MetFaces. We compare the clean accuracy and accuracies under image corruptions in Table 4. Although the directly transferred DeepRAFT model is not trained on MetFaces dataset, it still achieves a remarkable detection accuracy even with different image corruptions. This suggests that DeepRAFT can be successfully transferred to unseen image data; thus being flexible for future deployment in the real platform. Meanwhile, the robustness towards common image corruptions is transferred successfully as well. For instance, under the corruptions of DFB, CRP and PSW, the transferred accuracy achieves 100%.

4.7 Discussions and Limitations. Although our DeepRAFT framework works efficiently with a high accuracy, there are still challenges for broad deployment in the wild. 1) The current anti-forwarding scheme works with an assumption that different media platforms share the same DeepRAFT system, which will largely depend on the collaboration from different companies. The good news is that different countries and regions have been proposing data privacy and security related disciplines Hagendorff (2020), which makes it promising that different companies could follow a same trustworthy privacy protection system. 2) The robustness is regularized based on several yet still limited image corruptions in simulation. For real deployment, further explorations towards its vulnerability are needed, including more types of corruptions and real-world testing. 3) The training is quite struggling when random azimuth variations (e.g., cropping and perspective transform) is introduced, which need future work on make the training on such data augmentation more stable.

5 Conclusion

We have presented the DeepRAFT framework that is trained in an end-to-end manner to enable media platforms to check and block any unauthorised forwarding through injecting *non-fragile and invisible* watermarks. The DeepRAFT encoder and scanner are jointly trained, where the encoder embeds a confidentiality stamp into images as watermarks, and the scanner learns to detect them even with a stochastic concatenation of data augmentations (i.e., rotation, cropping, color jitters, defocus blur, perspective warping, pixel noise, JPEG compression). Moreover, we improve the adversarial robustness of DeepRAFT by involving randomized smoothing. Experiments on MirFlickr and Metfaces datasets indicate that our model can not only efficiently and robustly detect whether an image should be anti-forwarded, but also be easily transferred in a zero-short scenario. Therefore, this work opens up a plethora of new research questions of anti-forwarding privacy protection that have not yet been investigated in machine learning.

REFERENCES

Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, pp. 308–318, 2016.

Maksym Andriushchenko, Francesco Croce, Nicolas Flammarion, and Matthias Hein. Square attack: a query-efficient black-box adversarial attack via random search. In *European Conference on Computer Vision*, pp. 484–501. Springer, 2020.

Shumeet Baluja. Hiding images in plain sight: Deep steganography. *Advances in Neural Information Processing Systems (NeurIPS)*, 30, 2017.

Ghazaleh Beigi and Huan Liu. A survey on privacy in social media: Identification, mitigation, and applications. *ACM Transactions on Data Science*, 1(1):1–38, 2020.

Gordon W Braudaway. Protecting publicly-available images with an invisible image watermark. In *Proceedings of international conference on image processing*, volume 1, pp. 524–527. IEEE, 1997.

Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. Certified adversarial robustness via randomized smoothing. In *International Conference on Machine Learning (ICML)*, pp. 1310–1320. PMLR, 2019.

Francesco Croce and Matthias Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *International Conference on Machine Learning (ICML)*, pp. 2206–2216. PMLR, 2020.

Cynthia Dwork, Aaron Roth, et al. The algorithmic foundations of differential privacy. *Found. Trends Theor. Comput. Sci.*, 9(3-4):211–407, 2014.

Han Fang, Weiming Zhang, Hang Zhou, Hao Cui, and Nenghai Yu. Screen-shooting resilient watermarking. *IEEE Transactions on Information Forensics and Security*, 14(6):1403–1418, 2018.

Thilo Hagendorff. The ethics of ai ethics: An evaluation of guidelines. *Minds and Machines*, 30(1): 99–120, 2020.

Jamie Hayes and George Danezis. Generating steganographic images via adversarial training. *Advances in Neural Information Processing Systems (NeurIPS)*, 30, 2017.

Mark J Huiskes and Michael S Lew. The mir flickr retrieval evaluation. In *Proceedings of the 1st ACM international conference on Multimedia information retrieval*, pp. 39–43, 2008.

Mark J Huiskes, Bart Thomee, and Michael S Lew. New trends and ideas in visual concept detection: The mir flickr retrieval evaluation initiative. In *Proceedings of the international conference on Multimedia information retrieval*, pp. 527–536, 2010.

Honglu Jiang, Jian Pei, Dongxiao Yu, Jiguo Yu, Bei Gong, and Xiuzhen Cheng. Applications of differential privacy in social network analysis: a survey. *IEEE Transactions on Knowledge and Data Engineering*, 2021.

Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training generative adversarial networks with limited data. *Advances in Neural Information Processing Systems (NeurIPS)*, 33:12104–12114, 2020.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems (NeurIPS)*, 25, 2012.

Anat Levin, Dani Lischinski, and Yair Weiss. Colorization using optimization. In *ACM SIGGRAPH*, pp. 689–694. 2004.

Bo Liu, Ming Ding, Sina Shaham, Wenny Rahayu, Farhad Farokhi, and Zihuai Lin. When machine learning meets privacy: A survey and outlook. *ACM Computing Surveys (CSUR)*, 54(2):1–36, 2021.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *International Conference on Learning Representations (ICLR)*, 2018a.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *International Conference on Learning Representations (ICLR)*, 2018b.

Alessandro Mantelero. From group privacy to collective privacy: towards a new dimension of privacy and data protection in the big data era. pp. 139–158, 2017.

Sébastien Marcel and Yann Rodriguez. Torchvision the machine-vision package of torch. In *Proceedings of the 18th ACM international conference on Multimedia*, pp. 1485–1488, 2010.

Matthew Mirman, Timon Gehr, and Martin Vechev. Differentiable abstract interpretation for provably robust neural networks. In *International Conference on Machine Learning (ICML)*, pp. 3578–3586. PMLR, 2018.

Takao Nakamura, Atsushi Katayama, Masashi Yamamuro, and Noboru Sonehara. Fast watermark detection scheme from camera-captured images on mobile phones. *International Journal of Pattern Recognition and Artificial Intelligence*, 20(04):543–564, 2006.

Anu Pramila, Anja Keskinarkaus, and Tapio Seppänen. Toward an interactive poster using digital watermarking and a mobile phone camera. *Signal, Image and Video Processing*, 6(2):211–222, 2012.

Anu Pramila, Anja Keskinarkaus, and Tapio Seppänen. Increasing the capturing angle in print-cam robust watermarking. *Journal of Systems and Software*, 135:205–215, 2018.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pp. 234–241. Springer, 2015.

Alexander K Saeri, Claudette Ogilvie, Stephen T La Macchia, Joanne R Smith, and Winnifred R Louis. Predicting facebook users' online privacy protection: Risk, trust, norm focus theory, and the theory of planned behavior. *The Journal of social psychology*, 154(4):352–369, 2014.

Ali Shafahi, Mahyar Najibi, Mohammad Amin Ghiasi, Zheng Xu, John Dickerson, Christoph Studer, Larry S Davis, Gavin Taylor, and Tom Goldstein. Adversarial training for free! *Advances in Neural Information Processing Systems (NeurIPS)*, 32, 2019.

Kartik Sharma, Ashutosh Aggarwal, Tanay Singhania, Deepak Gupta, and Ashish Khanna. Hiding data in images using cryptography and deep neural network. *Journal of Artificial Intelligence and Systems*, 1(1):143–162, 2019.

Richard Shin and Dawn Song. Jpeg-resistant adversarial images. In *NIPS 2017 Workshop on Machine Learning and Computer Security*, volume 1, 2017.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *International Conference on Learning Representations (ICLR)*, 2014.

Matthew Tancik, Ben Mildenhall, and Ren Ng. Stegastamp: Invisible hyperlinks in physical photographs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2117–2126, 2020.

Weixuan Tang, Shunquan Tan, Bin Li, and Jiwu Huang. Automatic steganographic distortion learning using a generative adversarial network. *IEEE Signal Processing Letters*, 24(10):1547–1551, 2017.

Ruowei Wang, Chenguo Lin, Qijun Zhao, and Feiyu Zhu. Watermark faker: towards forgery of digital image watermarking. In *2021 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1–6. IEEE, 2021.

Eric Wengrowski and Kristin Dana. Light field messaging with deep photographic steganography. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1515–1524, 2019.

Eric Wengrowski, Wenjia Yuan, Kristin J Dana, Ashwin Ashok, Marco Gruteser, and Narayan Mandayam. Optimal radiometric calibration for camera-display communication. In *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 1–10. IEEE, 2016.

Jun Yu, Baopeng Zhang, Zhengzhong Kuang, Dan Lin, and Jianping Fan. iprivacy: image privacy protection by identifying sensitive objects via deep multi-task learning. *IEEE Transactions on Information Forensics and Security*, 12(5):1005–1016, 2016.

Huan Zhang, Hongge Chen, Chaowei Xiao, Sven Gowal, Robert Stanforth, Bo Li, Duane Boning, and Cho-Jui Hsieh. Towards stable and efficient training of verifiably robust neural networks. *International Conference on Learning Representations (ICLR)*, 2019.

Peng-Fei Zhang, Yang Li, Zi Huang, and Hongzhi Yin. Privacy protection in deep multi-modal retrieval. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 634–643, 2021.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 586–595, 2018.

Jiren Zhu, Russell Kaplan, Justin Johnson, and Li Fei-Fei. Hidden: Hiding data with deep networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 657–672, 2018.

6 APPENDIX

(a) Images from MirFlickr Huiskes & Lew (2008) (b) Images from MetFaces Karras et al. (2020) datset

Figure 4: Example images from MirFlickr and MetFaces dataset

Table 5: Robustness towards PGD, FGSM and random gaussian noise. Values represent accuracy percentage from 0 to 100.

	PGD (40-steps)					FGSM					Gaussian Noise				
Strength ϵ	$\frac{2}{255}$	$\frac{4}{255}$	$\frac{8}{255}$	$\frac{16}{255}$	$\frac{32}{255}$	$\frac{2}{255}$	$\frac{4}{255}$	$\frac{8}{255}$	$\frac{16}{255}$	$\frac{32}{255}$	$\frac{2}{255}$	$\frac{4}{255}$	$\frac{8}{255}$	$\frac{16}{255}$	$\frac{32}{255}$
Plain	0	0	0	0	0	0	0	0	0	0	99.92	99.92	99.92	99.93	98.82
RS(σ=0.25)	86.80	58.46	20.44	0.40	0	99.22	99.22	92.23	90.70	68.16	86.80	58.55	22.61	21.48	47.77
RS(σ=0.5)	67.46	63.75	57.20	50.52	29.27	67.75	64.39	58.70	52.12	46.38	99.83	99.82	99.82	99.80	99.74
RS(σ=0.75)	52.58	52.04	51.21	50.30	49.90	52.95	52.73	52.25	51.45	50.33	84.76	84.74	84.70	84.39	83.23
DA+RS (σ=0.25)	94.14	46.18	9.07	4.84	4.53	94.74	70.25	23.84	10.02	12.39	99.83	99.83	99.83	99.82	99.81

Figure 5: Additional demos of watermark invisibility by comparing original images, added watermarks and resulting watermarked images

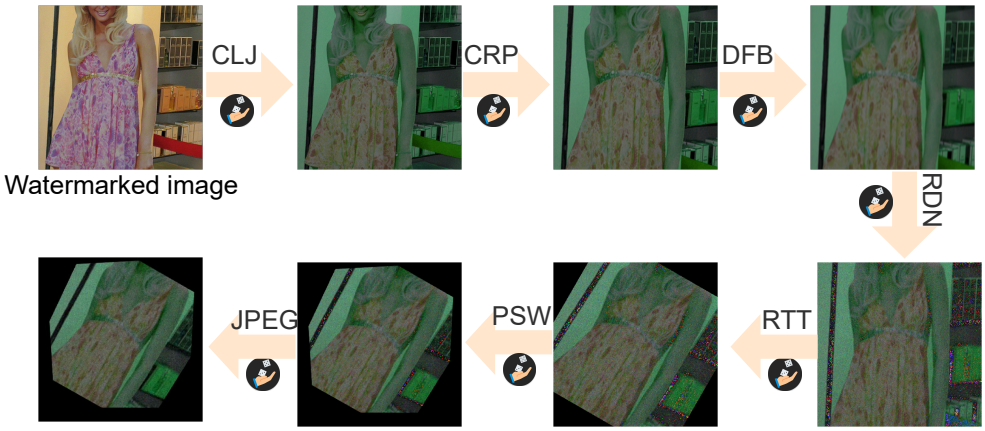

Figure 6: Showcase of the accumulation from different image corruptions.

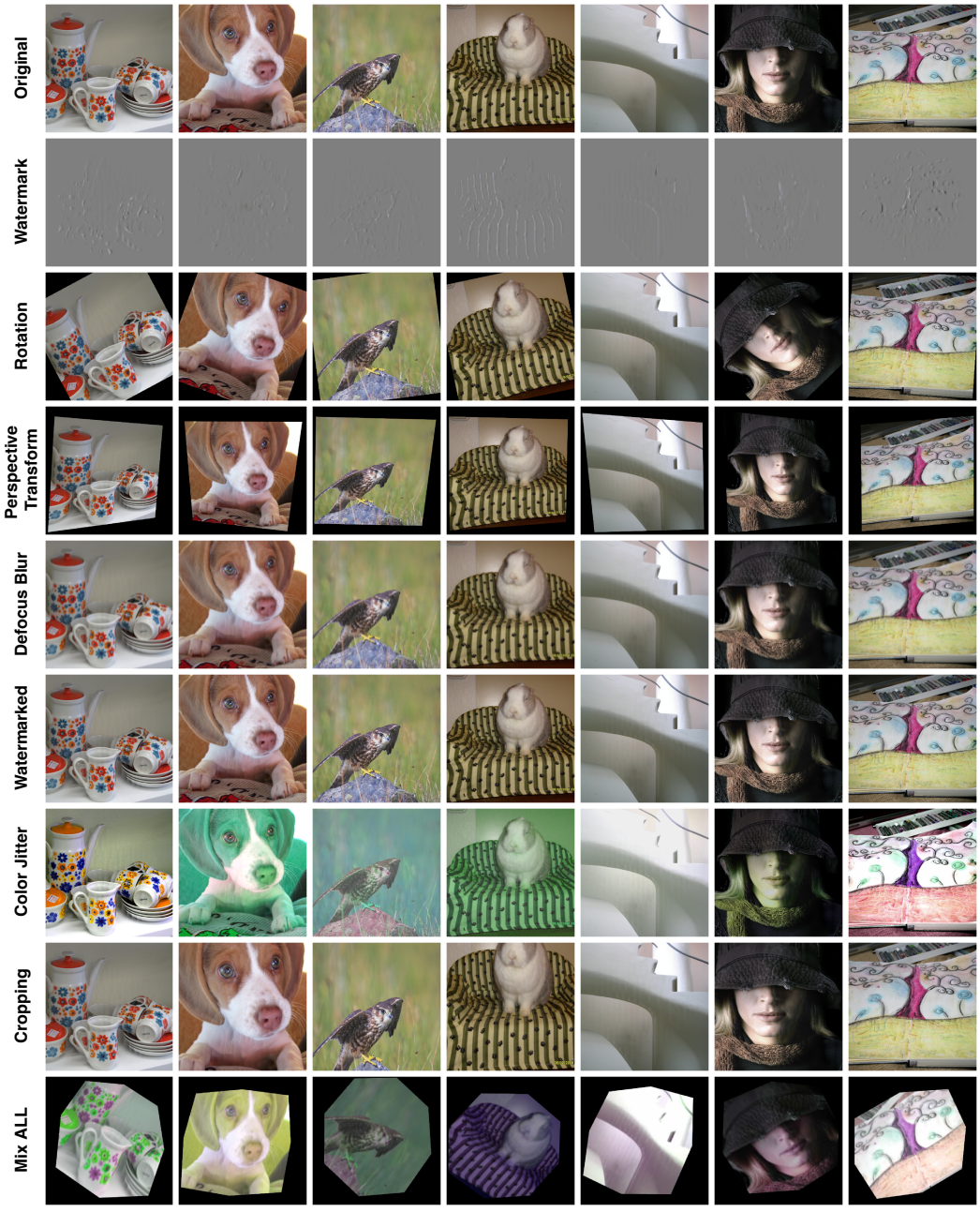

Figure 7: Additional visualization examples of DeepRAFT watermarked images and the corresponding image corruptions. The mixed in the last row indicates all image corruptions are concatenated together.