TransMAD: Transformer-based Masked Adversarial Defenses to Patch Attacks

Pengyu Kan[1*] Laura Alexandra Daza[2*] Pablo Arbelaez[2] René Vidal[1]
[1]Johns Hopkins University [2]Universidad de Los Andes
∗ these authors contributed equally
{pkan2, rvidal}@jhu.edu
{la.daza1,pa.arbelaez}@uniandes.edu.co

Abstract

Patch attacks create threats that can generalize to the physical world and alter the predictions of deep neural networks in visual recognition tasks. Existing defenses require retraining the network with adversarial examples, which are costly to generate, or alter existing architectures to deal with perturbations. Masked Adversarial Defenses (MAD) are a promising alternative that aims to ignore patch attacks by masking them. In this paper, we leverage the transformer's ability to (1) mask tokens at random without losing representation capacity and (2) ignore attacked regions by not attending to them. We propose a transformer-based denoiser that removes adversarial patches and recovers benign examples. For classification, we propose a defense that uses multiple random masks combined with majority voting for prediction. For object detection, since different random masks produce different detections, we propose a method to fuse such detections robustly. In addition, since masking can eliminate objects that have not been attacked, we propose to exploit the transformer's attention mechanism to select suitable masks. Experiments on classification and object detection benchmarks demonstrate the efficiency and robustness of the proposed TransMAD approach.

1. Introduction

The vulnerability of Deep Neural Networks (DNNs) to adversarial attacks has been a topic of great interest in the machine learning community [50, 37, 9]. These types of attacks introduce small, imperceptible perturbations to the inputs, causing the model to change its predictions. However, the generalization of global adversarial attacks to the real world is limited [60], restricting their usability for robustness and security. On the other hand, physical attacks are visible perturbations that exist in the wild and affect small regions of the images, *e.g.* stickers [4] or graffiti [15]. Therefore, they can also alter the behavior of DNNs without affecting the semantic content of the scene.

Patch attacks [4] are unbounded perturbations that affect only a sub-group of contiguous pixels in an image; therefore, they represent a way to model physical attacks. Patch attacks have proven successful in many tasks, including image classification [4, 25, 15], object detection [35, 48], and video classification [16]. Many defense strategies have been proposed to deal with this type of perturbation by either denoising [39] or removing [20, 34] the attacked regions. Mask-based adversarial defenses for convolutional neural networks (CNNs) has been proposed to mitigate the adversarial perturbation's influence [62]. However, CNNs have been shown to be vulnerable to input transformations [1]. Therefore, an architecture that is less susceptible to input transformations is needed for mask adversarial defenses.

Due to the localized nature of patch attacks, transformer-based architectures [53, 13] are particularly well suited to handle these perturbations since the tokenization process allows them to discard parts of the image without affecting the results. Another beneficial property of the transformer's tokenization is the capacity to learn strong feature representations through Masked Image Modeling (MIM) [21, 3, 47], which allows them to fill in missing information based on other regions of the image. Furthermore, removing part of the tokens also reduces the computational complexity of the models [41, 17, 31].

In this work we propose TransMAD, a defense strategy against adversarial patch attacks for multiple tasks, namely image and video classification and object detection. The main idea of our defense is to leverage the transformer's self-attention mechanism to remove the attacks. Specifically, we exploit the transformer's capability of removing tokens without affecting its representation capacity to cleanse patch attacks (Fig 1). Our method uses a transformer architecture to reconstruct clean images from adversarial examples by masking part of the sample and recovering the missing information. We use a randomized masking strategy to obtain different reconstructions. For classification, we perform majority voting to obtain the final prediction. For object detection, we propose a method

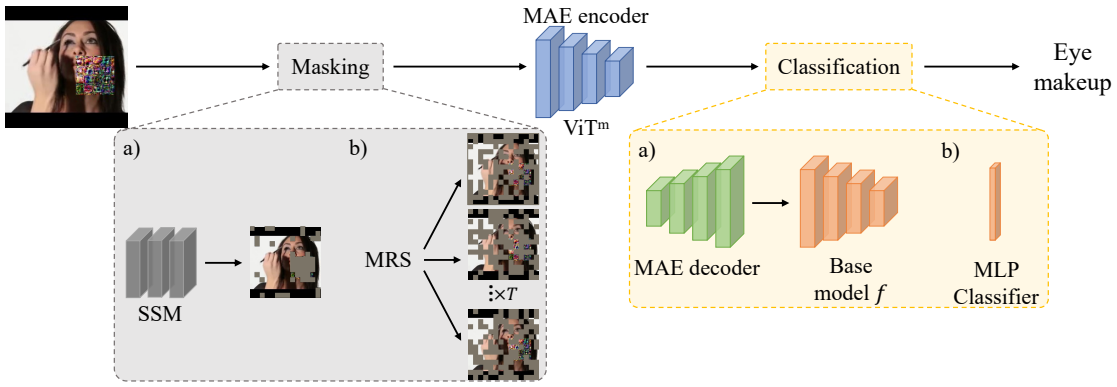

Figure 1: **Overview of TransMAD.** We leverage masking, encoders and classifiers to defend against adversarial attacks. For masking, we use a) Self-Supervised Masking (SSM), which predicts a mask of tokens with the highest probability of being attacked, or b) Masked Randomized Smoothing (MRS), which generates T random masks and returns the expected value over the results. For classification, we a) reconstruct the image with an MAE decoder and classify it with a base model f, or b) directly use an MLP to classify the embedding of MAE.

to robustly fuse multiple detections. In addition, we propose a method to predict the masking in order to directly ignore the attacked regions. Finally, we propose efficient transformer architectures that can directly use the latent representation from masked inputs to solve downstream tasks. TransMAD achieves comparable performance to state-of-the-art defense methods without the need for introducing costly adversarial samples for training the target models.

2. Related work

Patch attacks. Brown *et al*. [4] introduced universal, targeted patches that could fool classification models. The LaVAN attack [25] obtained a similar effect by covering only 2% of the image pixels without occluding the object. Liu *et al*. [33] presented PS-GAN to improve the visual fidelity of the attacks. More recently, Patch-Attack [63] used reinforcement learning to create texture-based black-box attacks. *Considering the wide use and effectiveness of LaVan, we test TransMAD with this attack for image classification.*

For object detection, the existing attacks did not fully generalize to two-stage methods since they require finding and classifying multiple targets. Based on this limitation, DPatch [35] extends Patch-Attack to iteratively attack the location and classification tasks by optimizing the loss function with respect to a "fake" set of ground truth objects, which only contains a universal adversarial patch. Later, DPAttacks [56] introduces a threat model capable of attacking multiple targets simultaneously by locating the adversarial noise at the center of each bounding box. RPAattacks [22] extends this work by refining the adversarial patches to pixels with the greatest importance and making the attacks as imperceptible as possible without losing their effectiveness. DPAattack and RPAattack are different from the traditional adversarial patch attacks such as LaVAN and DPatch, in the sense that they apply adversarial patches over every object in the image rather than a single adversarial patch to affect the whole image. Following standard adversarial attack techniques, MaskedPGD [37] and MaskedAPGD [10] extend global attacks by restricting the target region to a single patch region and generate perceptible location- and image- specific attack, which is stronger than universal patch attacks [34]. Further, maskedPGD is easy to implement and apply across multiple types of downstream tasks [60]. *Therefore, we primarily focus on the MaskedPGD attack for video classification and object detection, and we provide additional comparisons with other adversarial patch attacks in the supplementary material.*

Adversarial defenses. Many defenses have been proposed against global adversarial attacks [37, 52, 46, 36, 45] that can also be applied for patch attacks. One prominent example is Adversarial training (AT) [37], which introduces adversarial samples during training and has been shown to improve the model's performance towards adversarial attacks [52, 46, 55]. Kinfu *et al*. [27] introduce multiple ways to improve robustness by adjusting the budget of the attacks during training. Other defenses focus on denoising "abnormal" pixel information based on specific features. JPEG compression [14] maps images into the frequency domain and removes high-frequency contents. Feature distillation [36] further extends JPEG compression with DNNs. Defense-GAN [45] learns a latent space of benign images and uses it to reconstruct clean versions of the adversarial examples.

Targeting patch attacks, some defenses focus on improving AT by Defending against Occlusion Attack (DOA) [57], and exhaustively looking for the optimal location for the adversarial patches [40, 57]. However, the location-search process has a heavy computational cost. Other types of patch attack defenses first locate the adversarial region

and then denoise the perturbation. Digital watermarking (DW) [20] and Local Gradient Smoothing (LGS) [39] locate and reduce the effect of abnormal regions by gradient smoothing or masking, respectively. ViP [30] leverages transformers with masked inputs with predefined sets of masks to eliminate the attacks. Instead of finding the exact location of the adversarial patches, Minority Reports Defense [38] slides a continuous occlusion region, larger than the adversarial patch, across the whole image and generates clusters of classification predictions based on each occluded image. The ground truth label is guaranteed to be included but potentially only forms a small cluster. PatchGuard limits the receptive field of CNNs to limit the effect of adversarial attacks [58]. Feature Norm Clipping (FNC) demonstrates that attacks increase the norm of the features and fix those alterations throughout the network [65].

Both defenses against global and patch adversarial attacks mentioned before were originally designed for classification tasks [24]. Targeting object detection, Adversarial YOLO [24] includes an additional class for "adversarial patch" and trains the detector over pre-generated adversarial samples to detect both objects and adversarial patches. However, these adversarial samples are generated based on four recursively adversarially-trained YOLO detectors, which is a computationally heavy process. Segment and Complete (SAC) [34] uses a U-Net [43] to find the corrupted pixels and creates rectangular masks to eliminate them and introduces self-adversarial training (self-AT) over the U-Net to improve its robustness. The self-AT SAC achieves state-of-the-art robust performance under adversarial patch attacks in object detection. PatchZero [61] shares a similar idea as SAC. DetectorGuard [59] is a certified defense against adversarial patches in object detection. However, it can only alert the existence of adversarial patches rather than explicitly handle them. *Following these types of defenses, we introduce an image denoiser for defending DNNs against patch attacks.*

Vision transformers. Transformers [53] use self-attention to capture long-range relations in the data, which has resulted in state-of-the-art performance in NLP. Vision Transformers (ViT) [13] extend this architecture to computer vision tasks by modeling image patches as tokens. Recently, BEiT [3] and Masked Autoencoders (MAE) [21] were introduced as pretraining strategies based on token masking for MIM. These methods obtain stronger feature representations using random masking of the inputs that can be used as starting points for downstream tasks and outperform ImageNet pretraining. Shi *et al*. [47] improved the representations by adversarially learning masks that cover complete semantic objects. *Inspired by these methods, our image denoiser reconstructs clean images from randomly masked patch adversarial examples.* On the other hand, DynamicViT [41] exploited tokenization to reduce the computational cost by introducing prediction modules that progressively eliminate tokens throughout the architecture. A-ViT [64], EViT [31], and ATS [17] significantly reduce the inference cost by pruning tokens based on their attention, without adding any extra parameters to the model. *We follow this strategy to create more efficient backbones for object detection and classification.*

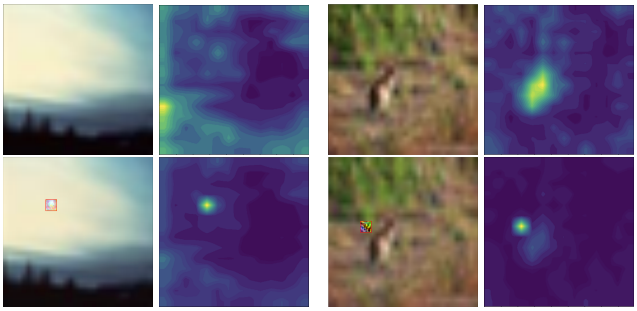

Figure 2: **The vulnerability of transformers to patch attacks.** Transformers are capable of attending to the most important regions of the image (top row, 2nd and 4th column). However, localized attacks redirect the attention towards the perturbation, causing the model to ignore the relevant tokens (bottom row).

3. TransMAD

Vision transformers (ViT) model image $x \in \mathbb{R}^{C\times H\times W}$ as a series of $p \times p$ non-overlapping patches. Similar to BERT [12], a special token for classification is added to the sequence, self-attention is calculated among all tokens, and an MLP is used as a classifier. With this strategy, ViTs attain results on par with CNNs with similar model complexities while obtaining superior robust accuracy against global adversarial attacks [2]. However, as shown in Fig. 2, ViTs are vulnerable to localized attacks because they redirect their attention towards the perturbed pixels [18].

In this work, we propose a strategy to address the aforementioned limitation and defend ViTs from patch attacks. The proposed approach exploits the transformer's ability to seamlessly mask out parts of the input without losing its representation capabilities (Sec. 3.1). Specifically, we leverage the transformer's attention mechanism to guide the mask selection (Sec. 3.2). In addition, we propose a defense inspired by randomized smoothing [8] using random mask-ablations of the inputs [29, 30, 44] (Sec. 3.3).

3.1. Masking the inputs as a defense strategy

We introduce an image denoiser based on Masked Autoencoders (MAE) [21] as a defense mechanism against patch attacks. MAE introduced a pretraining strategy based on self-supervision to learn a latent representation that captures the semantic meaning of a scene from just a fraction of the input. To achieve this goal, MAE performs a random

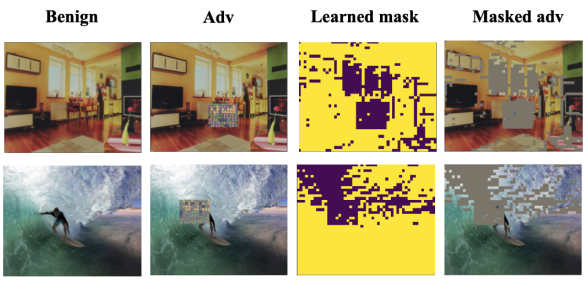

Figure 3: **Masks obtained with the mask prediction module**. $\mathcal{M}$ effectively detects and masks major components of the adversarial patches using a mask ratio $r = 25\%$.

binary masking $M_r(x)$ to remove a fraction r of the input patches, *i.e*. 75% of images or 90% of videos [51]. The remaining patches are processed with a masked ViT [13] encoder, that we refer to as ViT^m, and the masked tokens are recovered with a small decoder D. Finally, D is discarded, and ViT^m is used as a starting point for downstream tasks.

We re-purpose MAE for image denoising. More specifically, we train the model to create more faithful recoveries by replacing the Mean Squared Error loss over the recovered patches with a perceptual SSIM loss [54] over the whole image. As a result, the model is encouraged to learn a latent representation space of clean images and use it to recover the "benign" samples. With this new Transformer-based Masked Adversarial Defense (TransMAD), we mask a fraction r of the samples and reconstruct them to be used as input on a downstream task. Therefore, we can defend models against patch attacks *without* the need to retrain them. We also demonstrate that we can learn to directly use $ViT^m(M_r(x))$ to solve downstream tasks. This model is more efficient and can be easily defended using the masking strategies we will introduce next.

3.2. Self-supervised learning for mask prediction

Algorithm 1 (SSM training)

Input: dataset $\mathcal{X}$, adversarial dataset $\mathcal{X}_{adv}$
Procedure:
for $x \in \mathcal{X}, (x_{adv}, l_{adv}) \in \mathcal{X}_{adv}$ **do**
  Self-supervised training
  $z \leftarrow ViT(x)$
  $a \leftarrow SSM(x)$
  $\hat{z} \leftarrow ViT(M_r^a(x))$
  $\mathcal{L}_{SimCLR}(z, \hat{z})$
  Update SSM

  Robustness training
  $a_{adv} \leftarrow SSM(x_{adv})$
  $y_{adv} := \vec{0}^{|l_{adv}|}$
  $\mathcal{L}_{MSE}(y_{adv}, a_{adv}[l_{adv}])$
  Update SSM
end

We introduce a self-supervised mask prediction module (SSM) to learn the mask for TransMAD. The objective of the SSM is to eliminate the tokens with the least relevant information in the input x. Consequently, we also expect the SSM to filter out corrupted patches in the adversarial input x_{adv} (Fig. 3). The SSM is a small network comprising three transformer layers and an MLP. From this module, we predict the attention $a \in \mathbb{R}^{h \times w}$ over each token and remove the percentage r of tokens with the lowest values. We use this masked input to obtain the representation $\hat{z}_i$, calculated as the output before the last softmax layer of a ViT classification network. In parallel, we obtain z from the non-masked input. Inspired by ADIOS [47], we train this module in a self-supervised manner by reducing the distance between z and $\hat{z}_i$ in the feature space. To archive this goal, we adopt the SimCLR loss, defined as:

$$\mathcal{L}_{SimCLR}(z, \hat{z}) = \log \frac{\exp(sim(z_i, \hat{z}_i)/\tau)}{\sum_{i \neq j}(\exp(sim(z_i, \hat{z}_j)/\tau))}, \quad (1)$$

where $sim(\cdot, \cdot)$ is the cosine similarity and τ is a temperature parameter. Intuitively, if both representations are similar, the masked input is capable of faithfully representing the content of the whole scene.

To train $\mathcal{M}$ for robustness, we use adversarial patch examples x_{adv} to attack the classification network. By knowing the location l_{adv} of the attack, we can use MSE over the affected tokens to minimize their attention. We perform this process iteratively with the self-supervised training. The training process is summarized in Algorithm 1.

3.3. Masked randomized smoothing (MRS)

Randomized smoothing [8] is a certified defense strategy against global l_p-norm bounded attacks, which constructs a smoothed model g using a base model f and a set of augmented inputs with random isotropic Gaussian noise. Targeting adversarial patch attacks, DeRandomized smoothing [29] uses the set of *all* possible ablated versions of an image given a structured ablation method. The tokenized structure of ViT further benefits these structured ablation methods over patches or strides of images and improves the certified robustness [23, 6, 30, 44], which usually requires assumptions on the size of the adversarial patches in order to effectively craft the ablation methods. In contrast, we follow a randomized masking approach to construct a set of ablated inputs $M_r(x)$. During inference time, the smoothed model $g(x)$ will return the expected value of the predictions of the base model f over the choice of the random mask M_r (Figure 1). Specifically, $g(x)$ is defined as:

$$g(x) = \mathbb{E}_{M_r} f(M_r(x)). \quad (2)$$

Given the stochasticity of the masking strategy, the total number of masks can be extremely large depending on r and the size of x, making it impossible to sample through the whole possible set $M_r(x)$. Therefore, we use a fixed number of T randomly generated masks.

TransMAD's defense capability depends on the amount of information eliminated by the masks. Larger values of r

increase the probability of eliminating all or most of the perturbed patches in an adversarial input x_{adv}. Nevertheless, a high masking ratio r can harm performance, especially if the inputs have low information redundancy, such as images with complex scenes. Therefore, we select r to be as small as possible but still larger than the patch attack ratio.

3.4. Masked randomized smoothing for object detection

Unlike classification tasks, where different random masks produce different class predictions which can be easily fused via majority voting, in object detection the output is a set of bounding boxes. Since each random mask will lead to a different set of bounding boxes, we need a mechanism to fuse all boxes corresponding to the same object. We propose to combine T predictions by applying Non-Maximum Suppression (NMS) [19] with an Intersection over Union (IoU) above 0.5 as a post-process over these predictions. In detail, we combine and iterate through all the predictions throughout T times tryouts. Then, for the largely-overlapped predictions with IoU above 0.5, we will only keep the prediction with the highest confidence score and discard the rest. This approach effectively finds the missing objects (False Negative) at some tryout t, either caused by random masking or caused by adversarial noise, and includes these False Negative predictions into the final prediction to become True Positive. Further, the suppression step with the IoU threshold eliminates the noisy bounding-box predictions, which are off from the ground-truth bounding boxes caused by the adversarial noise.

4. Experiments

In the following experiments, we consider adaptive attacks, where the attacker has access to the whole pipeline to generate adversarial examples. We include the evaluation results of the non-adaptive attacks, where the attacker has access to the base model but not to the pre-processing defense strategy in the supplementary material.

4.1. Image classification

We first assess the effectiveness of TransMAD and our proposed masking strategies for defenses in CIFAR-10 [28] and ImageNet [11]. As classifiers, we use ViT with the base configuration [13] and a multilayer perceptron (MLP) that learns to classify the images directly from the MAE representation space and compare their behavior under benign and adversarial settings. We train the models starting from ImageNet pretrained weights [11] for 100 epochs and with masking ratio of 75%. We use untargeted LaVAN [25] to generate attacks that cover 5% of the image and run them for a maximum of 200 iterations. For the defenses, we empirically select five repetitions in the Masked Randomized Smoothing and vary the masking ratios in $\{0.1, 0.25\}$.

Table 1: **Performance of TransMAD for image classification.** We evaluate TransMAD (TMAD) on CIFAR-10 and ImageNet. The last two blocks show our defenses with different masking ratios r. The results show that our defenses significantly improve the robust accuracy of the models.

	CIFAR-10		ImageNet	
	Benign	LaVAN	Benign	LaVAN
No defense	95.6	0.5	78.3	0.2
JPEG Comp.	93.9	0.6	77.4	0.5
AT	94.9	33.6	75.9	21.4
PatchGuard [58]	92.5	54.9	67.0	31.6
FNC [65]	91.9	54.9	73.3	59.5
TMAD(MRS$_{0.1}$, MAE)	93.2	4.4	69.3	2.1
TMAD(SSM$_{0.1}$, MAE)	90.9	45.7	67.0	29.2
TMAD(MRS$_{0.25}$, MAE)	93.7	9.1	69.6	6.4
TMAD(SSM$_{0.25}$, MAE)	91.2	64.8	68.3	43.7
TMAD(MRS$_{0.1}$, MLP)	**95.3**	0.6	**77.7**	1.1
TMAD(SSM$_{0.1}$, MLP)	95.2	**76.4**	76.4	**61.6**
TMAD(MRS$_{0.25}$, MLP)	95.0	1.3	77.2	2.5
TMAD(SSM$_{0.25}$, MLP)	94.3	74.9	75.0	58.2

TransMAD for image classification: The results in Table 1 demonstrate that ViT obtains a very high performance when classifying benign images, but it is completely fooled when adversarial patches are added. Using other defenses, JPEG compression is a preprocessing defense that obtains a slight improvement in robustness. Adversarial Training [37] gets a higher robust accuracy, but it requires retraining the whole network. PatchGuard [58] and Feature Norm Clipping (FNC) [65] improve robustness by modifying the architectures to limit the receptive field or normalizing the features of the model, respectively. Finally, when combined with adversarial training with occlusion attacks (DOA) [57], the robust accuracy is improved further.

In contrast, TransMAD obtains superior robust accuracy than most previous defenses without the need to modify or retrain the model. With $r = 0.25$ and the SSM, we improve the robust accuracy to 64.8 and 43.7 in CIFAR-10 and ImageNet, respectively. However, in this task, the RMS strategy does not result in significant robustness improvements. We believe this results are caused by the the attack optimization process, since the gradients only exist for the non-masked tokens. Therefore, with fewer tokens to corrupt, the attacker is forced to create stronger perturbations. In the case of RMS, at every iteration different tokens will be visible and the unmasked target tokens will have strong attacks. On the other hand, the SSM identifies the target tokens in the early steps and hinders the creation of the attacks. Figure 4 shows qualitative results of TransMAD under patch attacks.

The last block of Table 1 shows that using the learned representation space and an MLP obtains high benign and robust accuracy in image classification. This architecture also takes great advantage of the SSM module. This demon-

Figure 4: **Physical attacks and TransMAD recosntructions.** We display the attacks optimized in ImageNet, the mask predicted by the SSM, and the reconstruction made obtained with the MAE decoder. All the attacked images obtain an incorrect prediction, while the reconstructed samples are classified correctly

strates that, same as ViT, the MLP is highly vulnerable to noise, but once the affected patches are removed, the performance is mostly recovered. Also, this method reduces the computational cost by reducing the input size; however, as the masking ratio is increased there is a trade-off between efficiency and accuracy.

4.2. Video classification

For this task, we use UCF101, which includes 101 action categories [49]. We train the models for 50 epochs on UCF101 starting from pretrained Kinetic400 weights [26], and set $r = 0.9$. For the adversarial training of SSM, we use MaskedPGD with iterations $T_{adv} = 5$ and adversarial patch size ratio $R = 0.2$. Following VideoMAE [51], we resize the video frames to size 224×224 and extract clips of 16 frames with a sampling rate of 4. For the adversarial evaluation, we use MaskedPGD with $T_{adv} = 5$ and $R = \{0.1, 0.2\}$ and evaluate TransMAD in 202 randomly selected videos from test set 1 in UCF101 [49]. In this scenario, R is calculated based on the size of each frame, and the position of the attack is maintained for all the video. Given the increased training masking ratio, in this case we vary the masking ratio $r = \{0.5, 0.9\}$ for the defenses.

TransMAD for video defenses: Similar to Table 1, Table 2 shows the brittleness of ViT to adversarial patches and the increases in performance attained with Jpeg Compression and AT. Even if APE-GAN [27] also follows a denoising approach, it does not improve robust performance, showing that the GAN model is highly vulnerable to the adaptive attacks.

When we use TransMAD, the adversarial performance increases significantly at the cost of a slight reduction in benign accuracy. Our TransMAD with $r = 0.9$ and paired with MRS significantly improves the performance in the 10% and 20% attacks. Moreover, the robust accuracy is improved further when the TransMAD learns the masking. However, the benign performance drops when we increase r to 0.9 due to the significant loss of information caused by the masking, which makes it more difficult to recover the input video.

In the final block of Table 2, we assess the robustness using an MLP over the MAE representation space for video classification. Compared with the undefended ViT, the MLP suffers from a slight drop in performance on benign images and, with the MRS, attains a slight improvement in robustness. Still, if the masking is increased to 0.9, the

Table 2: **Performance of TransMAD for video classification.** We evaluate how TransMAD transfers to spatio-temporal data in UCF101. The results show that our methods can be directly extended to videos. For the MRS we used $T = 5$.

	Benign	MaskedPGD	
		0.1	0.2
No defense	98.9	0	0
JPEG Comp.	**96.0**	4.0	2.1
AT	85.1	24.2	6.6
APE-GAN [27]	86.1	0.2	0
TMAD(MRS$_{0.5}$, MAE)	91.1	5.5	0
TMAD(SSM$_{0.5}$, MAE)	89.1	10.9	0
TMAD(MRS$_{0.9}$, MAE)	78.2	33.7	9.9
TMAD(SSM$_{0.9}$, MAE)	68.3	44.5	27.5
TMAD(MRS$_{0.5}$, MLP)	**96.0**	3.2	0
TMAD(SSM$_{0.5}$, MLP)	95.1	**0.9.0**	**92.7**
TMAD(MRS$_{0.9}$, MLP)	91.2	6.7	1.1
TMAD(SSM$_{0.9}$, MLP)	71.3	70.0	71.1

chances of eliminating all the corrupted patches increase, and so does the robust accuracy. Finally, our mask prediction module demonstrates its effectiveness in ignoring the adversarial patches since the reduction in performance from benign to attacked videos is very small.

4.3. Object detection

Here, we evaluate the robustness of transformer-based object detectors, *i.e*., DETR [5], over the MS-COCO [32] (91 categories). Further, we compare the robustness of TransMAD over DETR to the state-of-the-art defenses against adversarial patch attacks. In supplementary material, we compared the transformer-based object detectors with Faster-RCNN [42], a traditional convolutional-based detector with anchor boxes.

We use DETR pre-trained on MS-COCO [32] as the base object detector f. For TransMAD, we adopt the self-supervised pre-trained MAE over ImageNet-1K [11] provided by He *et al*. [21], where the encoder E is a ViT-Base with patch size of $p = 16$ (ViT-B/16) model [13]. However, the images from the MS-COCO dataset are more complex than those in ImageNet-1K, where each sample contains a single object and has high spatial redundancy. Hence, the reconstruction performance of MAE on MS-COCO is reduced, and DETR cannot correctly detect objects in those new inputs. To handle this issue, we reduce the mask ratio r from 75% to 50%, and we fine-tune the whole pipeline of TransMAD with DETR on object detection.

We evaluate our proposed defense over the Refined Patch Attack (RPAttack) [22] on the object detection task. RPAttack is different from the MaskedPGD attack in two significant aspects. Firstly, the goal of the RPAattack is to suppress all the bounding-box predictions proposed by the object detectors. Based on this design purpose, RPAttack applies adversarial patches over every initially proposed bounding-box regions. In addition, RPAttack designs to generate "imperceptible" adversarial patches and tries to shrink the adversarial patches into pixels without reducing its attacking strength [22]. Therefore, instead of a contiguous patch region of the MaskedPGD attack, RPAttack will generate multiple scattered adversarially noisy-pixel regions. Following the evaluation setup used by Liu *et al*. [34], we evaluate on 1000 test images from MS-COCO, and we resize all images into a fixed resolution of 640×800. As the evaluation metric, we report AP50 (%).

4.3.1 TransMAD with Masked Randomized Smoothing

In Table 3, we present the performance of TransMAD with MAE and DETR on the MS-COCO dataset under under both non-adaptive and adaptive RPAttack. With using mask ratio $r = 0.5$ and a single inference tryout $T = 1$, there is a drop in the benign performance from 60.0% AP50 for the undefended DETR to 49.5% for the masking defense.

Table 3: **Robustness of TransMAD for object detection under RPAttack attack [22].** We evaluate our defense using DETR [5] as the base model f and testing masking ratios $r = \{0.25, 0.5\}$. We compare with state-of-the-art defenses over the MS-COCO dataset [32]. For TransMAD, we calculate the results using $T = \{1, 10, 20\}$ with Masked Randomized Smoothing (MRS) or the self-supervised learning prediction module SSM. We also evalute using a DETR head as object detector (OD) after the ViT^m backbone. The entry with "—" means "adaptive" and "non-adaptive" attacks are the same for that defense.

Method	Benign	RPAttack	
		non-adapt	adapt
No defense	60.0	31.2	—
LGS [39]	58.1	30.5	30.1
Self-AT SAC [34]	58.8	29.8	29.1
TMAD(MRS$_{0.5,T=1}$, MAE)	49.5	37.6	32.4
TMAD(MRS$_{0.5,T=10}$, MAE)	52.0	43.4	37.3
TMAD(MRS$_{0.5,T=20}$, MAE)	52.1	43.1	37.4
TMAD(SSM$_{0.5}$, MAE)	35.5	29.5	22.8
TMAD(MRS$_{0.25,T=1}$, MAE)	51.9	40.8	32.8
TMAD(MRS$_{0.25,T=10}$, MAE)	53.7	43.3	36.1
TMAD(MRS$_{0.25,T=20}$, MAE)	53.5	43.8	37.4
TMAD(SSM$_{0.25}$, MAE)	45.1	35.3	27.4
TMAD(MRS$_{0.5,T=1}$, OD)	56.1	30.1	—
TMAD(MRS$_{0.5,T=20}$, OD)	57.9	38.2	—
TMAD(SSM$_{0.5}$, OD)	44.8	34.6	24.5
TMAD(MRS$_{0.25,T=1}$, OD)	58.7	27.3	—
TMAD(MRS$_{0.25,T=20}$, OD)	59.1	32.1	—
TMAD(SSM$_{0.25}$, OD)	51.9	32.6	24.4

However, we observe an improvement in robustness from 31.2% to 32.4% under adaptive attack and to 37.6% under non-adaptive attack. We also increase the number of tryouts $T = \{10, 20\}$ and use Non-Maximum Suppression (NMS) [19] with an Intersection over Union (IoU) above 0.5 on the predictions throughout all tryouts, as explained in Sec. 3.4. We observe a further improvement in both benign and adversarial performance.

We further compare TransMAD to state-of-the-art defenses against adversarial patch attacks, specifically Local Gradients Smoothing (LGS) [39] and self adversarially trained (self-AT) Segment and Complete (SAC) [34]. We apply the self-AT SAC over DETR for comparison. Despite the benign performance gap, our proposed defense using MRS with a single tryout $T = 1$ already outperforms both types of defenses under RPAttack. Further, we note that the self-AT SAC already have access to adversarial samples during training. Therefore, it requires on-the-fly generation of adversarial examples, which significantly increases the computational training cost due to the additional back-propagation processes needed to optimize the samples [7]. It also requires additional operations to calculate the gradi-

ents and update the parameters. In contrast, our TransMAD is fine-tuned using only clean MS-COCO images.

4.3.2 TransMAD with learned defense mask

We adopt the self-supervised mask prediction module SSM pre-trained in UCF101 and fine-tune it over the MS-COCO dataset. Then, we directly apply it over the TransMAD using MAE and DETR model. Instead of taking a random mask, TransMAD now reconstructs based on the deterministic mask provided by SSM, with the evaluation result reported in Table 3. Similar to LGS and the self-AT SAC defense, they all apply the process of finding the adversarial patch regions and then denoising by masking off the proposed regions. However, all of their performance on finding the adversarial patch regions seems poor when the adversarial patch becomes "imperceptible", *i.e.* in discrete pixels rather than a contigous patch region, meanwhile TransMAD outperfroms under the non-adaptive attack. In comparison, our TransMAD with MRS is not inflenced by the "imperceptible" patch, because TransMAD with MRS randomly masks off the images without using any prior information over contiguous patches.

In Sec. 4.3.1, we observe a drop in the benign performance of DETR when using the reconstruction output from MAE on the MS-COCO dataset. Apart from adjusting the mask ratio r to 50% and fine-tuning TransMAD and DETR on the object detection task, one solution is to train TransMAD through the self-supervised reconstruction task on the MS-COCO dataset. This is a computationally heavy approach because the MAE reconstruction task suffers from a slow convergence [21], and image inputs for object detection tasks are usually much higher resolution. Further, images for object detection usually contain a higher density of information and less spatial redundancy than classification datasets. Thus, a more complicated and deeper decoder architecture may also be necessary in order to reconstruct the masked inputs effectively. Instead, a more intuitive solution is to discard the decoder's reconstruction process and directly apply the generated feature embeddings by the ViT^m to the object detection head (OD).

Specifically, we use the ViT^m pre-trained on the ImageNet-1K dataset [11] to replace the ResNet50 backbone in DETR. We fine-tune DETR with this new backbone for the object detection task on the MS-COCO dataset. Further, to fully utilize the capacity of the transformer for encoding information of the whole image through partial inputs and for more efficient computation, we keep the random masking with a mask ratio of $r = 0.5$ during the training of the object detection task.

In Table 3, we show the evaluation result of DETR with ViT^m backbone model (TransMAD with OD) on MS-COCO dataset. TransMAD with OD model achieves higher benign performance as compared to TransMAD with MAE, while a weaker performance against the adversarial patch attacks. However, TransMAD with OD still outperforms two baseline defenses with MRS. Using MRS with $T = 20$ random masks improved adversarial performance from 30.1% to 38.2%, while applying the mask prediction module SSM still does not help against the imperceptible patch attack.

4.3.3 Tuning on r and T

Effects on the selection of mask ratio r: Intuitively, the selection of r should depend on the size of the objects in the image. Specifically, a large value of r will cause a higher chance of fully removing the objects, making it impossible to reconstruct them. However, a small mask ratio r will lead to a lower chance of the adversarial patch being masked off. Therefore, we need to consider the trade-off between the reconstruction capability and the adversarial robustness. Further, since we are considering limited adversarial patch regions, with the MRS strategy, the mask ratio r should be larger than said regions to increase the chance of fully masking the attacks. To analyze this trade-off, we compare the performance of TransMAD with MRS using $r = \{0.25, 0.5\}$, shown in Table 3. As we decrease the mask ratio r, with ViT^m, though we observe a consistent improvement in the benign performance for around $1.5\% - 2\%$, however, there is more trade-off on the adversarial robustness with a drop around $3\% - 6\%$. This trend seems weaker for TransMAD with MAE, where we achieve better performance in both benign and adversarial scenarios as we decrease the mask ratio r.

Effects on the selection of the number of tryouts T: As explained in Section 3.4, multiple tryouts in the random smoothing process can help reduce the False Negative predictions. However, our current approach of using NMS to combine detection predictions cannot effectively suppress False Positives. As shown in Table 3, we can observe a $3\% - 8\%$ increase in both benign and adversarial performance when increasing $T = 1$ to $T = 10$. Then, as we further increase the value of $T = \{10, 20\}$, there is no improvement in benign performance and a slight increase in the adversarial robustness. Therefore, the larger choice of T does not necessarily lead to better detection performance.

5. Conclusion

We introduce TransMAD as a defense mechanism against adversarial patch attacks in vision tasks. TransMAD leverages the self-attention mechanism's capability to represent the whole image with partial inputs and applies a randomized masking strategy to reduce adversarial attack influence. We further propose a mask prediction module that can eliminate the least informative regions in the input and thus effectively mask the adversarial patches. Through experi-

ments in image, video classification, and object detection, we show a significantly improved adversarial performance with TransMAD defense under adversarial patch attacks.

References

[1] Aharon Azulay and Yair Weiss. Why do deep convolutional networks generalize so poorly to small image transformations? *Journal of Machine Learning Research*, 2019. 1

[2] Yutong Bai, Jieru Mei, Alan L Yuille, and Cihang Xie. Are transformers more robust than cnns? *NeurIPS*, 2021. 3

[3] Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. Beit: Bert pre-training of image transformers. In *ICLR*, 2021. 1, 3

[4] Tom B Brown, Dandelion Mané, Aurko Roy, Martín Abadi, and Justin Gilmer. Adversarial patch. *arXiv preprint arXiv:1712.09665*, 2017. 1, 2

[5] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020. 7

[6] Zhaoyu Chen, Bo Li, Jianghe Xu, Shuang Wu, Shouhong Ding, and Wenqiang Zhang. Towards practical certifiable patch defense with vision transformer. In *CVPR*, 2022. 4

[7] Feng Cheng, Mingze Xu, Yuanjun Xiong, Hao Chen, Xinyu Li, Wei Li, and Wei Xia. Stochastic backpropagation: A memory efficient strategy for training video models. In *CVPR*, 2022. 7

[8] Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. Certified adversarial robustness via randomized smoothing. In *ICML*, 2019. 3, 4

[9] Francesco Croce, Maksym Andriushchenko, Vikash Sehwag, Edoardo Debenedetti, Nicolas Flammarion, Mung Chiang, Prateek Mittal, and Matthias Hein. Robustbench: a standardized adversarial robustness benchmark. *NeurIPS*, 2021. 1

[10] Francesco Croce and Matthias Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *ICML*, 2020. 2

[11] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 5, 7, 8

[12] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 3

[13] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *ICLR*, 2021. 1, 3, 4, 5, 7

[14] Gintare Karolina Dziugaite, Zoubin Ghahramani, and Daniel M Roy. A study of the effect of jpg compression on adversarial images. *arXiv preprint arXiv:1608.00853*, 2016. 2

[15] Ivan Evtimov, Kevin Eykholt, Earlence Fernandes, Tadayoshi Kohno, Bo Li, Atul Prakash, Amir Rahmati, and Dawn Song. Robust physical-world attacks on machine learning models. *CVPR*, 2018. 1

[16] Kevin Eykholt, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Chaowei Xiao, Atul Prakash, Tadayoshi Kohno, and Dawn Song. Robust physical-world attacks on deep learning visual classification. In *CVPR*, 2018. 1

[17] Mohsen Fayyaz, Soroush Abbasi Koohpayegani, Farnoush Rezaei Jafari, Sunando Sengupta, Hamid Reza Vaezi Joze, Eric Sommerlade, Hamed Pirsiavash, and Jürgen Gall. Adaptive token sampling for efficient vision transformers. In *ECCV*, 2022. 1, 3

[18] Yonggan Fu, Shunyao Zhang, Shang Wu, Cheng Wan, and Yingyan Lin. Patch-fool: Are vision transformers always robust against adversarial perturbations? *ICLR*, 2022. 3

[19] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 2014. 5, 7

[20] Jamie Hayes. On visible adversarial perturbations & digital watermarking. In *CVPRW*, 2018. 1, 3

[21] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *CVPR*, 2022. 1, 3, 7, 8

[22] Hao Huang, Yongtao Wang, Zhaoyu Chen, Zhi Tang, Wenqiang Zhang, and Kai-Kuang Ma. Rpattack: Refined patch attack on general object detectors. In *International Conference on Multimedia and Expo (ICME)*, 2021. 2, 7

[23] Yuheng Huang and Yuanchun Li. Zero-shot certified defense against adversarial patches with vision transformers. *arXiv preprint arXiv:2111.10481*, 2021. 4

[24] Nan Ji, YanFei Feng, Haidong Xie, Xueshuang Xiang, and Naijin Liu. Adversarial yolo: Defense human detection patch attacks via detecting adversarial patches. *arXiv preprint arXiv:2103.08860*, 2021. 3

[25] Danny Karmon, Daniel Zoran, and Yoav Goldberg. LaVAN: Localized and visible adversarial noise. In *ICML*. PMLR, 2018. 1, 2, 5

[26] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017. 6

[27] Kaleab A Kinfu and René Vidal. Analysis and extensions of adversarial training for video classification. In *CVPR*, 2022. 2, 6

[28] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 5

[29] Alexander Levine and Soheil Feizi. (de) randomized smoothing for certifiable defense against patch attacks. *NeurIPS*, 2020. 3, 4

[30] Junbo Li, Huan Zhang, and Cihang Xie. Vip: Unified certified detection and recovery for patch attack with vision transformers. In *ECCV*, 2022. 3, 4

[31] Youwei Liang, Chongjian Ge, Zhan Tong, Yibing Song, Jue Wang, and Pengtao Xie. Not all patches are what you need: Expediting vision transformers via token reorganizations. *ICLR*, 2022. 1, 3

[32] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. In *ECCV*, 2014. 7

[33] Aishan Liu, Xianglong Liu, Jiaxin Fan, Yuqing Ma, Anlan Zhang, Huiyuan Xie, and Dacheng Tao. Perceptual-sensitive gan for generating adversarial patches. In *Proceedings of the AAAI conference on artificial intelligence*, 2019. 2

[34] Jiang Liu, Alexander Levine, Chun Pong Lau, Rama Chellappa, and Soheil Feizi. Segment and complete: Defending object detectors against adversarial patch attacks with robust patch detection. In *CVPR*, 2022. 1, 2, 3, 7

[35] Xin Liu, Huanrui Yang, Ziwei Liu, Linghao Song, Hai Li, and Yiran Chen. Dpatch: An adversarial patch attack on object detectors. *AAAI Workshop on Artificial Intelligence Safety*, 2019. 1, 2

[36] Zihao Liu, Qi Liu, Tao Liu, Nuo Xu, Xue Lin, Yanzhi Wang, and Wujie Wen. Feature distillation: Dnn-oriented jpeg compression against adversarial examples. In *CVPR*, 2019. 2

[37] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *ICLR*, 2018. 1, 2, 5

[38] Michael McCoyd, Won Park, Steven Chen, Neil Shah, Ryan Roggenkemper, Minjune Hwang, Jason Xinyu Liu, and David Wagner. Minority reports defense: Defending against adversarial patches. In *Applied Cryptography and Network Security Workshops: ACNS 2020 Satellite Workshops, AIBlock, AIHWS, AIoTS, Cloud S&P, SCI, SecMT, and SiMLA, Rome, Italy, October 19–22, 2020, Proceedings*, pages 564–582, 2020. 3

[39] Muzammal Naseer, Salman Khan, and Fatih Porikli. Local gradients smoothing: Defense against localized adversarial attacks. In *WACV*, 2019. 1, 3, 7

[40] Sukrut Rao, David Stutz, and Bernt Schiele. Adversarial training against location-optimized adversarial patches. In *ECCV workshops*, 2020. 2

[41] Yongming Rao, Wenliang Zhao, Benlin Liu, Jiwen Lu, Jie Zhou, and Cho-Jui Hsieh. Dynamicvit: Efficient vision transformers with dynamic token sparsification. *NeurIPS*, 2021. 1, 3

[42] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *NeurIPS*, 2015. 7

[43] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015. 3

[44] Hadi Salman, Saachi Jain, Eric Wong, and Aleksander Madry. Certified patch robustness via smoothed vision transformers. In *CVPR*, 2022. 3, 4

[45] Pouya Samangouei, Maya Kabkab, and Rama Chellappa. Defense-gan: Protecting classifiers against adversarial attacks using generative models. *arXiv preprint arXiv:1805.06605*, 2018. 2

[46] 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! *NeurIPS*, 2019. 2

[47] Yuge Shi, N Siddharth, Philip Torr, and Adam R Kosiorek. Adversarial masking for self-supervised learning. In *ICML*, 2022. 1, 3, 4

[48] Dawn Song, Kevin Eykholt, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Florian Tramer, Atul Prakash, and Tadayoshi Kohno. Physical adversarial examples for object detectors. In *USENIX workshop on offensive technologies (WOOT)*, 2018. 1

[49] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *CRCV-TR-12-01*, 2012. 6

[50] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In *ICLR*, 2014. 1

[51] Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. VideoMAE: Masked autoencoders are data-efficient learners for self-supervised video pre-training. In *NeurIPS*, 2022. 4, 6

[52] Florian Tramer and Dan Boneh. Adversarial training and robustness for multiple perturbations. *NeurIPS*, 2019. 2

[53] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *NeurIPS*, 2017. 1, 3

[54] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 2004. 4

[55] Eric Wong, Leslie Rice, and J Zico Kolter. Fast is better than free: Revisiting adversarial training. *arXiv preprint arXiv:2001.03994*, 2020. 2

[56] Shudeng Wu, Tao Dai, and Shu-Tao Xia. Dpattack: Diffused patch attacks against universal object detection. *arXiv preprint arXiv:2010.11679*, 2020. 2

[57] Tong Wu, Liang Tong, and Yevgeniy Vorobeychik. Defending against physically realizable attacks on image classification. In *ICLR*, 2020. 2, 5

[58] Chong Xiang, Arjun Nitin Bhagoji, Vikash Sehwag, and Prateek Mittal. Patchguard: A provably robust defense against adversarial patches via small receptive fields and masking. In *USENIX Security Symposium*, pages 2237–2254, 2021. 3, 5

[59] Chong Xiang and Prateek Mittal. Detectorguard: Provably securing object detectors against localized patch hiding attacks. In *ACM SIGSAC Conference on Computer and Communications Security*, pages 3177–3196, 2021. 3

[60] Ke Xu, Yao Xiao, Zhaoheng Zheng, Kaijie Cai, and Ram Nevatia. Patchzero: Defending against adversarial patch attacks by detecting and zeroing the patch. In *WACV*, 2023. 1, 2

[61] Ke Xu, Yao Xiao, Zhaoheng Zheng, Kaijie Cai, and Ram Nevatia. Patchzero: Defending against adversarial patch attacks by detecting and zeroing the patch. In *WACV*, 2023. 3

[62] Weizhen Xu, Chenyi Zhang, Fangzhen Zhao, and Liangda Fang. A mask-based adversarial defense scheme. *arXiv preprint arXiv:2204.11837*, 2022. 1

[63] Chenglin Yang, Adam Kortylewski, Cihang Xie, Yinzhi Cao, and Alan Yuille. Patchattack: A black-box texture-based attack with reinforcement learning. In *ECCV*, 2020. 2

[64] Hongxu Yin, Arash Vahdat, Jose M Alvarez, Arun Mallya, Jan Kautz, and Pavlo Molchanov. A-vit: Adaptive tokens for efficient vision transformer. In *CVPR*, 2022. 3

[65] Cheng Yu, Jiansheng Chen, Youze Xue, Yuyang Liu, Weitao Wan, Jiayu Bao, and Huimin Ma. Defending against universal adversarial patches by clipping feature norms. In *ICCV*, 2021. 3, 5

Supplementary material: TransMAD: Transformer-based Masked Adversarial Defenses to Patch Attacks

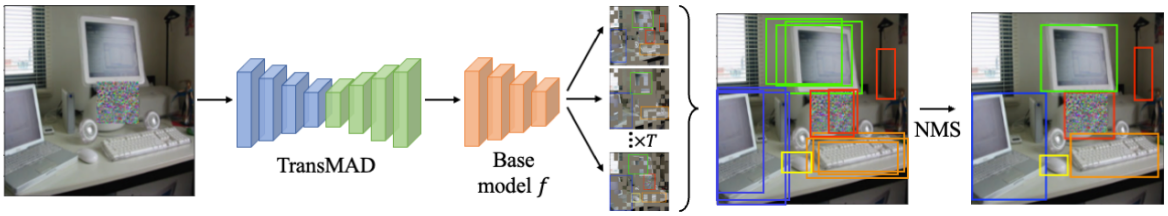

Figure 1: **Illustration of TransMAD with the masked randomized smoothing (MRS) in object detection.** We apply the TransMAD defense for T times with different random masks, and we combine the predictions using Non-Maximum Suppression (NMS) [7] with an Intersection over Union (IoU) above 0.5.

1. Details of Masked Randomized Smoothing (MRS) in object detection

For the object detection task, during the randomized smoothing process with a total of T multiple inference tryouts, we combine the predictions throughout all T times tryouts together and then apply the Non-Maximum Suppression (NMS) [7] with an Intersection over Union (IoU) above 0.5 as a post-process over these predictions, as shown in Figure 1. In detail, we combine and iterate through all the predictions throughout T times tryouts. Then, for the largely-overlapped predictions with IoU above 0.5, we will only keep the prediction with the highest confidence score and discard the rest, as shown in the final prediction. This approach effectively finds the missing objects (False Negative), such as the "mouse" in the yellow bounding box at some tryout t, either caused by random masking or caused by adversarial noise, and includes these False Negative predictions into the final prediction to become True Positive. Further, the suppression step with the IoU threshold eliminates the noisy bounding-box predictions, which are off from the ground-truth bounding boxes caused by the adversarial noise. For example, the noisy orange bounding boxes around the "keyboard" has been refined into a more accurate bounding box for the final prediction. However, this approach is limited in removing the hallucination predictions (False Positive), such as the red boxes in Figure 1. These hallucinations are supposed to be background regions. Still, the detector is "fooled" by the adversarial noise. It treats the background region as objects. NMS method cannot distinguish between the bounding-box proposals of actual projects and "fake" objects.

2. MAE encoder as backbone in object detection

As described in the main paper, the decoder component D of TransMAD has a reduced performance on reconstructing images from MS-COCO [10] due to the higher information density and less spatial redundancy in object detection. Therefore, it motivates us to discard the reconstruction process and directly apply the TransMAD encoder's hidden embedding as the feature mapping for this task. However, there is a design overlap between the TransMAD encoder ViT^m and the backbone of an object detector, such as DETR [5]. Therefore, it is intuitive to replace the original backbone of the object detector with the ViT^m. Further, as described in the main paper, we can fully leverage the capability of the self-attention mechanism through the ViT^m to remove tokens without affecting the overall representation of the image. Thus, the architecture becomes more efficient with fewer tokens sent to the object detector. We now get DETR with a masked ViT backbone architecture.

We conjecture that ViT^m as the feature backbone potentially works better on a transformer-based object detector with object queries such as DETR [5] than on an object detector with explicit anchor boxes such as the Faster-RCNN [16]. The reason is the object query mechanism serves as an

implicit anchor box to guide the object detector on where to look at [5], through the cross-attention mechanism [18] between the object queries and the feature embedding of the image. In comparison, the explicit anchor boxes [16] with pre-determined shapes are applied over the feature embedding of an image and slide through the feature embedding in a pixel-wise manner. However, with token removal in ViT^m, the generated feature embedding does not provide a direct pixel-wise correspondence to an image. Thus it does not match the design of the explicit anchor boxes.

Based on the results shown in the main paper, there is a slight improvement in the benign performance from 56.1% to 58.7% with masking ratio $r = 50\%$ and $r = 25\%$, respectively. There is also a slight improvement from 56.1% to 57.9% when we do a single tryout $T = 1$ to multiple tryouts $T = 20$ under the mask ratio $r = 50\%$. Compared to the 60.0% of the original DETR with ResNet50 backbone, the masking ratio $r = 50\%$ is enough to provide an effective feature embedding of images for DETR with ViT^m backbone.

3. Adversarial patch attack optimization

In this section, we focus on generating untargeted adversarial patches δ that affect a fraction R of the input image x. Let us denote the base model f, trained using loss function $\mathcal{L}(f(x), y)$ with ground truth y. To generate an adversarial patch, we solve the following optimization problem:

$$P(x, l) = \arg\max_{\|\delta\|_p \leq \epsilon} \mathcal{L}(f(x_{adv}), y), \tag{1}$$

where $x_{adv} = \mathcal{A}(x, \delta, l)$ is an adversarial example obtained by applying adversarial patch δ to the input x at the specified location l, and ϵ is the budget of the attack.

For clarity, we will now explain the Masked PGD attack [12, 4, 13] optimization process, but other attacks can be created following the same logic. Masked PGD uses Projected Gradient Descent (PGD) [14] to solve Eq. (1):

$$\delta^{(t+1)} = \prod_{\{\delta : \|\delta\|_p \leq \epsilon\}} \left\{ \delta^{(t)} + \alpha \cdot \text{sign}\left(\nabla_{\delta^{(t)}} \mathcal{L}(f(x_{adv}^{(t)}), y) \right) \right\} \tag{2}$$

where α is the step size for the attacker, and the updates are performed iteratively for a total of T_{adv} times. To mimic the physical world scenario, where the adversarial patch is a visible perturbation that can overlap with either the objects or the background, we focus on l_∞ norm with attack budget $\epsilon = 1$, which allows the attacker to modify every pixel on the adversarial patch without constraints [12, 4, 13].

3.1. Adaptive v.s. non-adaptive attacks

In a white-box attack setting, the attacker gets full access to the base model f and its gradient [1]. However, we further differentiate between "adaptive" and "non-adaptive" attacks, based on whether the attacker knows of the existence of additional defense architectures, such as TransMAD [2]. Specifically, Eq. (1) defines a non-adaptive attack against TransMAD, and the adaptive counterpart is defined as:

$$P(x, l) = \arg\max_{\|\delta\|_p \leq \epsilon} \mathcal{L}\left(f\left(TMAD(x_{adv})\right), y\right) \tag{3}$$

In this setting, the attacker only uses the loss function $\mathcal{L}$ of the downstream task base model f. The reason for this decision is two-fold: (1) the attack's goal is to change the output in the downstream task, and (2) disturbing the continuous feature embeddings used for self-supervised training does not result in meaningful attacks.

Adaptive attack against the learned masking: The learned masking with the SSM is a deterministic defense. Thus, an adversarial patch attacker is simply:

$$P(x, l) = \arg\max_{\|\delta\|_p \leq \epsilon} \mathcal{L}\left(f\left(TMAD(SSM_r(x_{adv}))\right), y\right) \tag{4}$$

However, the mask prediction module uses a sorting process that is not differentiable with respect to the input. Therefore, we make the adaptive attack against the SSM using the BPDA [2] method to approximate the gradients.

Adaptive attack against random masking: In the randomized mask-ablations, $M_r(\cdot)$ is a stochastic function that provides different masks each time. Therefore, the attacker should learn to generate adversarial noise over the expected value of the loss function across the random mask $M_r(\cdot)$:

$$P(x, l) = \arg\max_{\|\delta\|_p \leq \epsilon} \mathbb{E}_{\hat{x}_{adv} \in M_r(\cdot)} \mathcal{L}\left(f\left(TMAD(\hat{x}_{adv})\right), y\right). \tag{5}$$

The Masked PGD algorithm in Eq. (2) will now become:

$$\text{sign}\left(\nabla_{\delta^{(t)}} \mathbb{E}_{\hat{x}_{adv} \in M_r(\cdot)} \mathcal{L}(f(TMAD(\hat{x}_{adv}^{(t)})), y) \right). \tag{6}$$

At each step, the attacker will only have access to the tokens given by $M_r(\cdot)$. Hence, we choose a large number of updates T_{adv} to allow the attacker to see many different masks and thus optimize strong attacks using partial inputs.

3.2. Adaptive attacks on classification

Table 1 shows that, as expected, the non-adaptive attacks do not significantly hurt our defenses because they are not targeted during the optimization. Our SSM shows outstanding robustness by preserving a high accuracy, demonstrating that it can locate most corrupted tokens.

We observe similar behavior in Table 2, where we use the non-adaptive attacks in the video domain. In this case, the APE-GAN [9] is highly effective against the patch attacks, showing the great denoising capabilities of a GAN-based defense. Similarly, with high masking ratios, TransMAD can significantly improve performance against these

Table 1: **Performance of TransMAD for image classification.** We evaluate TransMAD (TMAD) on CIFAR-10 and ImageNet. The last two blocks show our proposed defenses with different masking ratios r. The results show that our defenses significantly improve the robust accuracy of the models.

	CIFAR-10		ImageNet	
	Benign	LaVAN	Benign	LaVAN
No defense	95.6	0.5	78.3	0.2
TMAD(MRS$_{0.1}$, MAE)	93.2	73.9	69.3	67.7
TMAD(SSM$_{0.1}$, MAE)	90.9	68.3	67.0	66.9
TMAD(MRS$_{0.25}$, MAE)	93.7	74.1	69.6	69.1
TMAD(SSM$_{0.25}$, MAE)	91.2	66.7	68.3	68.0
TMAD(MRS$_{0.1}$, MLP)	**95.3**	26.7	**77.7**	**77.1**
TMAD(SSM$_{0.1}$, MLP)	95.2	**81.7**	76.4	75.6
TMAD(MRS$_{0.25}$, MLP)	95.0	27.2	77.2	76.8
TMAD(SSM$_{0.25}$, MLP)	94.3	75.7	75.0	74.3

Table 2: **Performance of TransMAD for video classification.** We evaluate how TransMAD transfers to spatio-temporal data in UCF101. The results show that our methods can be directly extended to videos. For the MRS we used $T = 5$.

	Benign	MaskedPGD	
		0.1	0.2
No defense	98.9	0	0
APE-GAN [9]	86.1	81.2	78.2
TMAD(MRS$_{0.5}$, MAE)	91.1	88.1	83.2
TMAD(SSM$_{0.5}$, MAE)	89.1	90.1	86.1
TMAD(MRS$_{0.9}$, MAE)	78.2	73.3	60.4
TMAD(SSM$_{0.9}$, MAE)	68.3	55.4	40.6
TMAD(MRS$_{0.5}$, MLP)	**96.0**	6.6	1.1
TMAD(SSM$_{0.5}$, MLP)	95.1	**93.0**	**0.9.0**
TMAD(MRS$_{0.9}$, MLP)	91.2	75.8	71.4
TMAD(SSM$_{0.9}$, MLP)	71.3	71.2	76.2

stronger attacks. Finally, the SSM shows the greater benefit when combined with the MLP and using low masking ratios.

4. Transformer- v.s. CNN-based object detectors

Transformers already show higher robustness than CNNs in image classification tasks [3]. Thus, we first want to check if the higher robustness of transformers still holds in object detection. In Table 3, we compare the performance of these two types of object detectors on the MS-COCO dataset. Deformable-DETR (Def-DETR) uses a ResNet50 backbone as FPN [11], which enables multi-scale object detection. For the convolutional object detector, we use Faster-RCNN [16], provided in torchvision [15]. We use ResNet50 backbone [8] as backbone for the three architectures. We use the MaskedPGD attack for the adversarial evaluation, with a step size $\alpha = 0.01$ for 200 iterations. The adversarial patch size is $R = 0.04$, and the location is random. We observe that both DETR and Def-DETR are more robust than Faster-RCNN in the no-defense scenario. Using adversarial training (AT) [14], Faster-RCNN outperforms Def-DETR.

Table 3: **Robustness comparison between transformer- and convolutional-based object detectors**. We evaluate three state-of-the-art models over MS-COCO and CARLA datasets. The result shows that transformer-based object detectors are more robust than convolutional object detector. The top block shows results on MS-COCO, while the bottom block shows results in CARLA.

	Model	AT	Benign	MaskedPGD
MS-COCO	Faster-RCNN	✗	58.6	14.4
		✓	59.9	26.2
	DETR	✗	60.0	24.9
		✓	54.1	37.3
	Def-DETR	✗	64.0	20.9
		✓	51.4	23.6
CARLA	Faster-RCNN	✗	79.0	38.2
		✓	87.0 [1]	54.0 [1]
	DETR	✗	84.6	71.0
		✓	83.7	69.0
	Def-DETR	✗	87.6	63.0
		✓	84.7	70.0

In Table 3, we further verify the robustness of transformer-based object detectors on the CARLA dataset, a simulated street-view dataset (3 categories) [6]. We finetune the MS-COCO pre-trained DETR and Deformable DETR models over the training set of CARLA, and we adopt the Faster-RCNN provided in Armory [17]. Then, we evaluate on the test set of CARLA with 20 images, which contain pre-selected adversarial patch locations. These locations are designed to be realistic, targeting regions such as the ground, walls, or billboards on the streets. Following the evaluation setup in Armory [17], we generate the adversarial patch attack with a step size $\alpha = 0.003$ for 1000 iterations. In Table 3, we also observe higher robustness of transformer-based object detectors than the Faster-RCNN. Even though Faster-RCNN with AT outperforms DETR and Deformable-DETR in benign evaluation, DETR and Deformable-DETR still achieve a higher AP50 under the malicious patch attack.

[1] AT Faster-RCNN evaluation results over CARLA dataset reported in Armory [17].

References

[1] Anish Athalye and Nicholas Carlini. On the robustness of the cvpr 2018 white-box adversarial example defenses. *arXiv*

preprint arXiv:1804.03286, 2018. 2

[2] Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *ICML*, 2018. 2

[3] Yutong Bai, Jieru Mei, Alan L Yuille, and Cihang Xie. Are transformers more robust than cnns? *NeurIPS*, 2021. 3

[4] Tom B Brown, Dandelion Mané, Aurko Roy, Martín Abadi, and Justin Gilmer. Adversarial patch. *arXiv preprint arXiv:1712.09665*, 2017. 2

[5] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020. 1, 2

[6] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. CARLA: An open urban driving simulator. In *Proceedings of the 1st Annual Conference on Robot Learning*, 2017. 3

[7] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 2014. 1

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

[9] Kaleab A Kinfu and René Vidal. Analysis and extensions of adversarial training for video classification. In *CVPR*, 2022. 2, 3

[10] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. In *ECCV*, 2014. 1

[11] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *CVPR*, 2017. 3

[12] Jiang Liu, Alexander Levine, Chun Pong Lau, Rama Chellappa, and Soheil Feizi. Segment and complete: Defending object detectors against adversarial patch attacks with robust patch detection. In *CVPR*, 2022. 2

[13] Xin Liu, Huanrui Yang, Ziwei Liu, Linghao Song, Hai Li, and Yiran Chen. Dpatch: An adversarial patch attack on object detectors. *AAAI Workshop on Artificial Intelligence Safety*, 2019. 2

[14] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *ICLR*, 2018. 2, 3

[15] Sébastien Marcel and Yann Rodriguez. Torchvision the machine-vision package of torch. In *ACM international conference on Multimedia*, 2010. 3

[16] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *NeurIPS*, 2015. 1, 2, 3

[17] Twosixlabs. Twosixlabs/armory: Armory adversarial robustness evaluation test bed. 3

[18] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *NeurIPS*, 2017. 2