Supplementary Document
for
HiCert: Toward Patch Robustness Certification and Detection for Deep Learning Systems Beyond Consistent Samples

Supplementary Section A
Definition of certified detection

There are different schools of thought on the definition of certified detection in the literature. Some works [21]–[23] aim to detect all harmful samples (i.e., inferring a violation of $f(x') = y_0$), while others [19], [20], [24] aim to detect the change in the prediction label from the benign sample x (i.e., inferring a violation of $f(x') = f(x)$). The latter kind underestimates the attacker's ability (e.g., the attacker may know y_0 in producing x' [10], [12]), making them insensitive to detecting those harmful samples without changing the prediction label perturbed from incorrectly predicted benign samples, such that $f(x) = f(x') \neq y_0$. Some works [22], [23] further require a certified detection defender silent on certified samples, but some others [41] do not, or they [24] only require a defender silent on a proper subset of their certified samples with the guarantee by making the defender compatible in semantics with certified recovery. Our adopted definition goes for worst-case detection (detecting all harmful samples) with the least assumption on benign samples (leaving whether a benign sample should be in a particular detection state unspecified).

Supplementary Section B
A Case Study on the Ineffectiveness of Peers for Incorrectly Predicted Benign Samples

For incorrectly predicted ImageNet samples, we performed a case study with MAE and three different sizes of the patch (32, 64, 96 pixels) under the same experimental settings as the experiment for answering RQ1 in Section V. PG++ with all five values (from 0.5 to 0.9) of τ in the experiment cannot certify any sample out of all 8751 incorrectly predicted samples in the ImageNet dataset, and D_{OMA} can certify only 1 sample (the file index is `n01751748/ILSVRC2012_val_00002154`).

Supplementary Section C
Special case of HiCert

When $\tau = 0$, HiCert is reduced to D_{OMA}. This is because the set $\{f_{conf}(x_{\text{M}}) \mid \text{M} \in \mathbb{M}_{\mathbb{P}}, f(x_{\text{M}}) \neq y_0\}$ becomes empty ($\emptyset$) if a given sample x with the true label y_0 is consistent (i.e., $[\text{OMA}(x, y_0) = \textit{True}]$), thereby obtaining $v(x) = [\max \emptyset < 0] = [-\infty < 0] = \textit{True}$. Meanwhile, $w(\hat{x})$ is reduced to $[\{\hat{x}_{\text{M}} \mid \text{M} \in \mathbb{M}_{\mathbb{P}}, f(\hat{x}_{\text{M}}) \neq f(\hat{x})\} \neq \emptyset]$, where the set $\{\hat{x}_{\text{M}} \mid \text{M} \in \mathbb{M}_{\mathbb{P}}, f(\hat{x}_{\text{M}}) \neq f(\hat{x})\}$ becomes non-empty only if the input sample $\hat{x}$ has a label difference, i.e., $[\text{OMA}(\hat{x}, f(\hat{x})) = \textit{False}]$.

On the other hand, when $\tau = 1$, HiCert is reduced to a trivial detection defender that certifies every benign sample x (i.e., $[\max\{f_{conf}(x_{\text{M}}) \mid \text{M} \in \mathbb{M}_{\mathbb{P}}, f(x_{\text{M}}) \neq y_0\} < \tau]$ always holds) and warns every input sample $\hat{x}$ (i.e., $[\min\{f_{conf}(\hat{x}_{\text{M}}) \mid \text{M} \in \mathbb{M}_{\mathbb{P}}, f(\hat{x}_{\text{M}}) = f(\hat{x})\} < \tau]$ always holds).

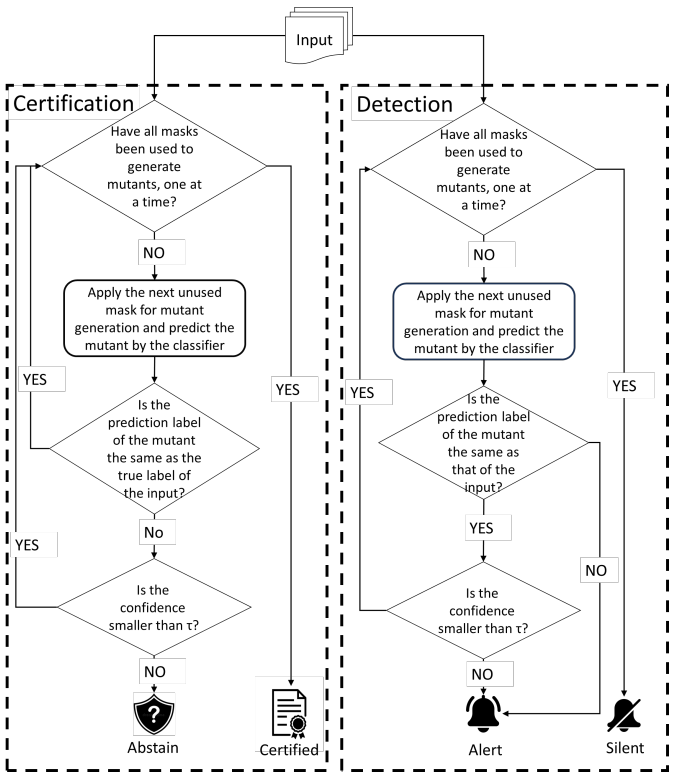

Fig. 11. The flowcharts of HiCert on certification and detection processes.

Supplementary Section D
Flowcharts of HiCert

The flowcharts of HiCert are shown in Fig. 11, illustrating two separate certification and detection processes in HiCert. In both flows, HiCert iteratively generates mutants and checks their prediction labels and confidence. For detection, if both conditions on the prediction label and confidence are met for all mutants, HiCert keeps silent on the input; otherwise, if any mutant violates either condition of prediction label or confidence, HiCert raises an alert. For certification, if either condition on the prediction label and confidence is met for all mutants, HiCert certifies this input; otherwise, if any mutant violates both the conditions of the prediction label and confidence, HiCert abstains from certifying this input.

Supplementary Section E
Theorem, proof, and the dilemma of attackers

Theorem (Consistent mutants are infeasible places for attackers)**.** *If the patch region is covered by a mask whose corresponding mutant's label is the same as the true label, it*

is infeasible for harmful samples to show no label difference. (i.e., if the condition $[\exists \mathrm{M_P} \in \mathbb{M}_\mathbb{P}, \mathrm{M_P} \odot \mathrm{P} = \mathrm{P} \wedge f(x_{\mathrm{M_P}}) = y_0]$ *holds, the condition* $[\forall x' \in \{x' \mid x' = (\mathrm{J} - \mathrm{P}) \odot x + \mathrm{P} \odot x'\}, [f(x') \neq y_0] \implies [\exists \mathrm{M} \in \mathbb{M}_\mathbb{P}, f(x'_{\mathrm{M}}) \neq f(x')]]$ *holds.)*

Proof. By $[\mathrm{M_P} \odot \mathrm{P} = \mathrm{P}]$, we know $x'_{\mathrm{M_P}} = ((\mathrm{J}-\mathrm{P}) \odot x + \mathrm{P} \odot x') \odot (\mathrm{J} - \mathrm{M_P}) = ((\mathrm{J}-\mathrm{P}) \odot x + \mathrm{P} \odot x') - ((\mathrm{J}-\mathrm{P}) \odot x + \mathrm{P} \odot x') \odot \mathrm{M_P} = (\mathrm{J} - \mathrm{P}) \odot x - (\mathrm{J} - \mathrm{P}) \odot x \odot \mathrm{M_P} = (\mathrm{J} - \mathrm{P}) \odot x \odot (\mathrm{J} - \mathrm{M_P}) = x \odot (\mathrm{J} - \mathrm{M_P}) = x_{\mathrm{M_P}}$ (see Fig. 4 for illustration). Here we also know $f(x_{\mathrm{M_P}}) = y_0$. So, we have $f(x'_{\mathrm{M_P}}) = y_0$. Further, if $f(x') \neq y_0$, we know $[\exists \mathrm{M} \in \mathbb{M}_\mathbb{P}, f(x'_{\mathrm{M}}) \neq f(x')]$. □

Theorem (HiCert Certification)**.** *If the maximum confidence of inconsistent mutants of a benign sample x is below a threshold τ, each harmful sample x' either incurs a label difference or has mutant(s) with minimum confidence below τ that are predicted with a label the same as x' — if the condition* $[\max\{f_{conf}(x_{\mathrm{M}}) \mid \mathrm{M} \in \mathbb{M}_\mathbb{P}, f(x_{\mathrm{M}}) \neq y_0\} < \tau]$ *holds, the condition* $[\forall x' \in \mathbb{A}_\mathbb{P}(x), [f(x') \neq y_0] \implies [\{x'_{\mathrm{M}} \mid \mathrm{M} \in \mathbb{M}_\mathbb{P}, f(x'_{\mathrm{M}}) \neq f(x')\} \neq \emptyset] \vee [\min\{f_{conf}(x'_{\mathrm{M}}) \mid \mathrm{M} \in \mathbb{M}_\mathbb{P}, f(x'_{\mathrm{M}}) = f(x')\} < \tau]]$ *holds, which is* $v(x) \implies [\forall x' \in \mathbb{A}_\mathbb{P}(x), f(x') \neq y_0 \implies w(x')]$ *in HiCert.*

Proof. Recall that $\mathrm{M_P}$ denotes the mask in the covering mask set $\mathbb{M}$ covering the patch in a harmful version x' of x (i.e., $[\mathrm{M_P} \odot \mathrm{P} = \mathrm{P}]$). We get $x_{\mathrm{M_P}} = x'_{\mathrm{M_P}}$ (see the proof of Thm. 1 above and Fig. 4 for illustration). Case 1: Suppose $f(x_{\mathrm{M_P}}) \neq y_0$. We know $[\max\{f_{conf}(x_{\mathrm{M}}) \mid \mathrm{M} \in \mathbb{M}_\mathbb{P}, f(x_{\mathrm{M}}) \neq y_0\} < \tau]$, which means $f_{conf}(x_{\mathrm{M_P}}) < \tau$ and $f_{conf}(x'_{\mathrm{M_P}}) < \tau$. Sub-Case 1.1: Suppose $[\{x'_{\mathrm{M}} \mid \mathrm{M} \in \mathbb{M}_\mathbb{P}, f(x'_{\mathrm{M}}) \neq f(x')\} \neq \emptyset]$ does not hold, which means $f(x'_{\mathrm{M_P}}) = f(x')$. Therefore, $[\min\{f_{conf}(x'_{\mathrm{M}}) \mid \mathrm{M} \in \mathbb{M}_\mathbb{P}, f(x'_{\mathrm{M}}) = f(x')\} < \tau]$ holds. Sub-Case 1.2: $[\{x'_{\mathrm{M}} \mid \mathrm{M} \in \mathbb{M}_\mathbb{P}, f(x'_{\mathrm{M}}) \neq f(x')\} \neq \emptyset]$ holds. Case 2 (Thm. 1): Suppose $f(x_{\mathrm{M_P}}) = y_0$. We have $f(x'_{\mathrm{M_P}}) = f(x_{\mathrm{M_P}}) = y_0$. Recalled that x' is harmful, we should have $f(x') \neq y_0$ and then $f(x'_{\mathrm{M_P}}) \neq f(x')$, thereby $[\{x'_{\mathrm{M}} \mid \mathrm{M} \in \mathbb{M}_\mathbb{P}, f(x'_{\mathrm{M}}) \neq f(x')\} \neq \emptyset]$ holds. □

By designing a warning function that follows Thm. 2, HiCert places attackers in a *dilemma* if they attempt to create a harmful sample x' of any benign sample x with $v(x) = True$ and aim to make HiCert silent on their created harmful samples. All these attempts of the attackers must be failed by HC since it can consistently alert on all these harmful samples.

Case 1 of the Dilemma: Suppose that an adversarial patch is placed within a given mask and its corresponding mutant of x (and also x') is inconsistent. Then, the confidence of this mutant of x must be lower than τ. If the harmful sample and all its mutants share the same prediction label (following Sub-Case 1.1 in the proof), then this mutant should be predicted with a label the same as the harmful sample, which means the minimum confidence of these mutants is lower than τ and the warning function of HiCert will return *True*. Otherwise, any mutants of the harmful sample that are predicted with a label different from the harmful sample (following Sub-Case 1.2 in the proof) indicate a label difference. So, the warning function of HiCert will also return *True*.

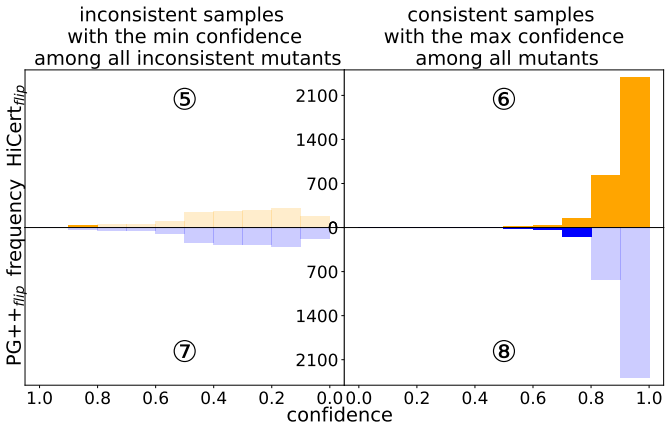

Fig. 12. The plots show the maximum and minimum confidences among those mutants of the same sample for those samples out of all samples, as stated in the column headings.

Case 2 of the Dilemma: Suppose that an adversarial patch is placed within a given mask and its corresponding mutant of x (and also x') is consistent. In this case, this mutant should be different from the harmful sample in the prediction label (following Case 2 in the proof). Like Case 1, the warning function of HiCert will also return *True*.

The two cases in the dilemma correspond to the two main cases in the proof of Thm. 2, respectively, which also correspond to the certification-warning paths depicted in Fig. 6: ④-⑤-①-⑧ (for Sub-Case 1.1), ④-⑤-②-⑦ (for Sub-Case 1.2), and ④-⑥-②-⑦ (for Case 2). Suppose both inconsistent benign samples in Fig. 7 satisfy the antecedent of the implication relation in Thm. 2. In this case, the first (x'_{1-1}) and the last (x'_{2-3}) harmful samples in the figure will be detected by the label difference condition through the Path ④-⑥-②-⑦, and the remaining three in between will be detected by the low confidence property through the Path ④-⑤-①-⑧.

SUPPLEMENTARY SECTION F
A CASE STUDY ON THE EFFECTIVENESS OF THE DESIGN OF HICERT

We analyze 5000 randomly selected ImageNet samples with their mutants from the experiment with MAE as the base model and the patch size of 32 pixels. Other settings are the same as answering RQ1 in Section V. We denote the set of 5000 samples by the set D. We split this set of samples D into two subsets: they contain the samples with and without inconsistent mutants (i.e., inconsistent samples and consistent samples), respectively, and are denoted by sets D_1 and D_2, respectively. We compute the maximum and minimum confidences among the confidences of all inconsistent mutants of the same sample for all samples in D_1 and these two bounds among the confidences of all (consistent) mutants of the same sample for all samples in D_2, to study the two types of confidence bounds on all samples in D, denoted by $\max(x)_1$ and $\min(x)_1$ for $x \in D_1$ and $\max(x)_2$ and $\min(x)_2$ for $x \in D_2$, respectively. The two columns of histograms in Fig. 8 from left to right correspond to the distributions for

$\max(x)_1$, $\min(x)_2$ for all samples x in respective sub-datasets D_1 and D_2.

Under the same setting, we also conduct an ablation study. We have further constructed two additional defenders PG++$_{flip}$ and HiCert$_{flip}$ to pair with PG++ and HiCert, respectively, by replacing the "$>$" operator with the "$<$" operator in PG++ for PG++$_{flip}$ and replacing the "$<$" operator with the "$>$" operator in HiCert for HiCert$_{flip}$, to demonstrate the inability of PG++$_{flip}$ and the ineffectiveness of HiCert$_{flip}$ to certify inconsistent samples (the meaning behind the direction of inequality sign of PG++$_{flip}$ is to require low confidence on consistent mutants; on the contrary, that of inequality sign of HiCert$_{flip}$ is to require high confidence on inconsistent mutants. Both are counterintuitive.). For clarification, their certification functions are $v(x) := [\text{OMA}(x, y_0) = \textit{True} \wedge \forall \text{M} \in \mathbb{M}_{\mathbb{P}}, f_{conf}(x_{\text{M}}) < \tau]$ for PG++$_{flip}$ and $v(x) := [\min\{f_{conf}(x_{\text{M}}) \mid \text{M} \in \mathbb{M}_{\mathbb{P}}, f(x_{\text{M}}) \neq y_0\} > \tau]$ for HiCert$_{flip}$. Their results are shown in sub-figures ⑤–⑧ in Fig. 12. The two columns of histograms in Fig. 12 from left to right correspond to the distributions for $\min(x)_1$ and $\max(x)_2$ for all samples x in respective sub-datasets D_1 and D_2. The bars for samples certified by the corresponding defenders (see the labels for the y-axis) are displayed in a solid color; otherwise, they are semi-transparent, where the confidence threshold τ is set to 0.8 for illustration purposes. Histograms ⑤–⑥ and ⑦–⑧ represent HiCert$_{flip}$ and PG++$_{flip}$, respectively. Although HiCert$_{flip}$ can certify all consistent samples (Histogram ⑥), it needs a small τ to cover a majority of all inconsistent samples to be effective in certifying these samples. PG++$_{flip}$ cannot certify any inconsistent samples (Histogram ⑦). Note that the ability for HiCert$_{flip}$ to certify inconsistent mutants is due to the advancements achieved by HiCert. The study shows that modifying the defenders to have the ability to certify both consistent benign samples and inconsistent benign samples (even in part) effectively is nontrivial.

SUPPLEMENTARY SECTION G
DISCUSSION ON THE DESIGN OF HICERT

1) Achieving the same time complexity as D_{OMA}: In terms of time complexity, compared to the D_{OMA} defender, HiCert only additionally requires a constant-time check on selective mutants' confidence against τ during certification or warning detection and there are $|\mathbb{M}|$ mutants in total. Since D_{OMA} already generates these $|\mathbb{M}|$ mutants and uses them for label prediction (and has to assess the confidence for the prediction label of mutants), HiCert is the same as D_{OMA} in time complexity, where $|\mathbb{M}|$ is a small constant in practice (e.g., $|\mathbb{M}|$ is 36 in our main experiments).

2) Soundness and completeness: In terms of theoretical soundness and completeness, like all other certified defenders with deterministic guarantees, HiCert is sound in certifying benign samples without any false positives (i.e., if a sample is reported as certified by HiCert, all of its harmful samples should be warned by HiCert, proven by Thm. 2). However, it is incomplete in certifying benign samples, with some false negatives (i.e., all harmful samples of some benign samples will always be detected by HiCert but HiCert doesn't report these benign samples as certified), because the actual situations of the mutants of each harmful sample of a benign sample may not be as bad as the worst-case scenario analyzed by HiCert (e.g., the prediction label of a mutant controlled by attackers may always be unable to be changed as the attackers want) if HiCert cannot certify it. This problem is shared by all existing certified defenders (both for recovery and for detection) and we are also unaware of any certified detection defender (including [19]–[24]) that is complete in certification unless the defender is a trivial one (warning all samples).

SUPPLEMENTARY SECTION H
DETAILS OF THE EXPERIMENTAL SETUP

A. Environment

We train the base models and generate mutants with their predicted label and confidence on GPU clusters with NVIDIA V100 GPUs. Data analysis is done on an Ubuntu 20.04 machine with Intel Xeon 6136 CPUs and NVIDIA 2080Ti GPUs.

B. Datasets

We adopt ImageNet [32] (widely used to evaluate patch robustness certification [19], [20], [22]–[24], [31], [33], [44]), CIFAR100 [35], and GTSRB [36] as our datasets.

These three datasets cover various applications, complexity, scale, and number of classes. ImageNet contains 1.3 million training images and 50,000 validation images for 1,000 classes. CIFAR100 contains 50,000 training images and 10,000 test images for 100 classes. GTSRB contains 39,209 training images and 12,630 test images.

We download ImageNet from image-net.org, use its entire training set for fine-tuning, and regard its validation set as the test set for evaluation. We download CIFAR100 and GTSRB from torchvision and use their whole training sets for fine-tuning and their test sets for evaluation. All images are resized to 224×224 in our experiments.

C. Baselines

Unlike previous work focusing on a single model architecture in evaluation (ViP [19] on MAE, PatchCensor [20] on ViT, and PatchGuard++ [23] on BagNet), our aim is to achieve technological versatility in evaluation. Therefore, we adopt all Masked Autoencoders [37] (vit-mae-base with 112M parameters, denoted as **MAE**), Vision Transformer [38] (vit-b16-224 with 86.6M parameters, denoted as **ViT**) and ResNet [39] (resnet-50 with 25.5M parameters, denoted as **RN**), as the architectures of the base models of defenders. We also use a model-agnostic pixel-level masking strategy following PatchCleanser [33] (PatchCleanser [33] is a certified recovery defender, which is a follow-up work of PatchGuard++ by the same first author) and CrossCert [24] instead of creating model-specific masking (channel masking for ViT/MAE [19], [20], feature masking for BagNet [23]). We follow the principle in PatchCleanser [33] to generate a covering mask set for each patch size.

We adopt the architectures and pre-trained weights from https://github.com/facebookresearch/mae for MAE, https://huggingface.co/timm/vit_base_patch16_224.augreg2_in21k_ft_in1k for ViT, and https://huggingface.co/timm/resnetv2_50x1_bit.goog_distilled_in1k for RN. We fine-tune MAE for each dataset by the original script from https://github.com/facebookresearch/mae. For fine-tuning ViT and RN, we use SGD with a momentum of 0.9, set the batch size to 64, and the number of epochs to 10, reducing the learning rate by a factor of 10 after every 5 epochs. Table I shows the clean accuracy of different base models for the datasets. Since MAE is the state-of-the-art (SOTA) base model [19], we use MAE as the default (main) base model in reporting the results of our evaluation.

We compare top-performing certified detection defenders implemented in our infrastructure to HiCert (**HC**): $\boldsymbol{D_{OMA}}$ and **PG++** [23]. Recall that D_{OMA} shares the common D_{OMA} checking strategy with ViP [19] and PatchCensor [20] but aims to detect $f(x') \neq y_0$ rather than $f(x') \neq f(x)$. With the same base model and the same masking strategy in our infrastructure, **ViP** [19] and PatchCensor (**PC**) [20] must share the same certified accuracy and clean accuracy with D_{OMA} since each sample x counted by these two metrics satisfies $f(x) = y_0$, and then the condition $\text{OMA}(x, f(x)) \wedge f(x) = y_0$ for ViP and PC is equivalent to the condition $\text{OMA}(x, y_0)$ for D_{OMA} in certification functions. Their certified ratios r_{cert} are the same as their certified accuracy acc_{cert} (which is also shared with D_{OMA}) because they cannot provide any warning guarantee for those incorrectly predicted benign samples in the situation where $f(x') \neq y_0 \wedge f(x') = f(x)$. The lower section of Table IV summarizes these results.

We further compare HiCert with more state-of-the-art certified detection defenders, and mark them with the symbol $\star$: ScaleCert (**SC**$_\star$) [22], PatchGaurd++ (**PG++**$_\star$) [23], Adapted Minority Reports (**MR+**$_\star$) [20], PatchCensor (**PC**$_\star$) [20], ViP (**ViP**$_\star$) [19], and CrossCert (**CC**$_\star$) [24] based on the results reported in the literature. Their results are summarized in the upper section of Table IV.

D. Metrics

Our evaluation will use certified accuracy, certified ratio, and certified ratio for inconsistent samples as the main metrics.

Suppose x is a benign sample with the true label y_0 in a test dataset $\mathbb{S}$ that only contains benign samples. Previous works use two key metrics, **clean accuracy**, to evaluate the inherent classification capability of the base model, and **certified accuracy**, to evaluate the certification ability of a defender on correctly predicted samples, which are defined as $acc_{clean} = \frac{|\{x \in \mathbb{S} | f(x) = y_0\}|}{|\mathbb{S}|}$ and $acc_{cert} = \frac{|\{x \in \mathbb{S} | f(x) = y_0 \wedge v(x) = True\}|}{|\mathbb{S}|}$ [24], despite some work [21], [23] excluding all benign samples that were warned by the defender concerned as elements in the set in the numerator, and some others (including the present paper) [19], [20], [24] including them. However, acc_{cert} discounts the certification ability of a defender on incorrectly predicted samples and cannot reflect the ability to certify inconsistent samples. So, we also measure the **certified ratio** $r_{cert} = \frac{|\{x \in \mathbb{S} | v(x) = True\}|}{|\mathbb{S}|}$, which counts all certified samples in $\mathbb{S}$, regardless of correct or incorrect predictions, and the **certified ratio for inconsistent samples** $r_{cert_{inc}} = \frac{|\{x \in \mathbb{S} | v(x) = True \wedge \text{OMA}(x, y_0) = False\}|}{|\{x \in \mathbb{S} | \text{OMA}(x, y_0) = False\}|}$, which counts the proportion of inconsistent samples that are certified.

Table III shows all eight combinations of three conditions on a benign sample: whether the sample is correctly predicted, whether it is warned, and whether it is certified, where a check symbol ✓ represents the corresponding condition is evaluated as true. We measure all these combinations on $\mathbb{S}$ to facilitate our detailed analysis case by case.

Fig. 2 has two outgoing paths after certified detection. For the silent path, we measure the **silent accuracy** $acc_{\neg w} = \frac{|\{x \in \mathbb{S} | w(x) = False \wedge f(x) = y_0\}|}{|\{x \in \mathbb{S} | w(x) = False\}|}$, the accuracy on the set of benign samples without warnings triggered, and for the alert path, we measure **false alert ratio** $r_{fa} = \frac{|\{x \in \mathbb{S} | w(x) = True \wedge f(x) = y_0\}|}{|\{x \in \mathbb{S} | f(x) = y_0\}|}$ [41], the fraction of correctly predicted samples for which a defender returns a warning alert, where having a higher value in r_{fa} may make the system waste more additional cost on these correctly predicted samples. Additionally, we measure the **false silent ratio** r_{fs}, the fraction of incorrectly predicted samples for which we do not return an alert: $r_{fs} = \frac{|\{x \in \mathbb{S} | w(x) = False \wedge f(x) \neq y_0\}|}{|\{x \in \mathbb{S} | f(x) \neq y_0\}|}$. A higher r_{fs} value signifies an increased number of incorrectly predicted samples, posing a greater threat to downstream operations. Note that all these metrics only measure the warning aspect of benign samples for readers to gain a deeper understanding of the warning ability of defenders. The number of warnings on benign samples cannot represent the warning ability of certified defenders on the whole input domain, which also includes non-benign samples that cannot be exhausted. However, the application scenario in Fig. 2 naturally requires a high proportion of samples that are correct and silent, identifying harmful (incorrectly predicted) samples and minimizing false warnings. We use them as secondary metrics to supplement the primary ones (acc_{cert}, r_{cert} and r_{suc}).

To answer RQ2, our experiment will generate actual samples to attack the defender. We compare the **defense success ratio** $r_{suc} = \frac{|\{x \in \mathbb{S}_{sub} | \forall x' \in \mathbb{A}_{\mathbb{P}}^{act}(x), f(x') \neq y_0 \implies w(x') = True\}|}{|\{\mathbb{S}_{sub}\}|}$ between defenders, where $\mathbb{S}_{sub}$ is a subset of $\mathbb{S}$ used by an actual attacker tool as seed input, $\mathbb{A}_{\mathbb{P}}^{act}(x)$ is a subset of $\mathbb{A}_{\mathbb{P}}(x)$ generated by the actual attacker tool. This metric measures the proportion of benign samples for which all harmful samples generated by an attacker tool are detected by the defender. (If not all harmful samples generated by an attacker tool on a benign sample are detected by the defender, the attack is called a *success attack* on the defender.) Unlike acc_{cert} and r_{cert} for theoretical defense ability, r_{suc} shows empirical defense ability against real adversarial patch attacks.

Except for r_{fa} and r_{fs}, higher values for all other metrics indicate better quality.

E. Experimental Setting

In this section, we describe the procedure of the experiments.

For RQ1, we follow the common practice in the evaluation of certified detection defenders to perform it on the benign samples [19], [20], [22]–[24], with the previously adopted

patch size to compared acc_{clean} and acc_{cert}: 32 pixels (2%) in ImageNet [19], [20], [22]–[24], 35 pixels (2.4%) in CIFAR100 [24], and 32 pixels (2%) in GTSRB [20]. We vary τ from 0.5 to 0.9 (previous work [23] chooses a similar range). The results of MAE are shown in Table. I and more results of other base models are shown in Fig. 10. We also perform a detailed analysis on benign samples of ImageNet with patch size 32 pixels with MAE, to also check r_{cert} and $r_{cert_{inc}}$ for the certification ability, and check $acc_{\neg w}, r_{fa}$, and f_{fs} for warning ability on benign samples as secondary metrics.

For RQ2, we perform an actual adversarial patch attack adopted from [30], which is gradient-based (using the base model f as the surrogate model to attain the gradient following [30]) and has no knowledge of defenders (HC/D_{OMA}/PG++) for the fairness.

Specifically, we select the first 500 benign samples from each shuffled dataset for the attack and set 80 random starts, 150 iterations per random start, and a step size of 0.05. We set patch sizes to 32, 64, and 96 pixels. For each patch size, we evaluate the warning function after each iteration for each defender with each τ from 0.5 to 0.9. If any harmful sample of a selected sample passes through undetected, the defender with that τ value is marked as having failed for the selected sample. Due to the scale of the experiment, we limit the evaluation of defenders to the most representative base model (MAE). We note that the defense success ratios are calculated based on the actual outcomes of the warning functions of the defenders. Since D_{OMA}, PC, and ViP share the same warning function, their defense success ratios are the same.

For RQ3, we follow [19], [20] to vary the patch size from 16 to 112 pixels, step by 16 pixels. τ is set to 0.8 in both PG++ and HC, and the results of PC and ViP are the same as those of D_{OMA}.

TABLE VI
RESULTS ON IMAGENET SAMPLES BY DIFFERENT MODES OF PATCH IN TOTAL 1% PATCH AREA

Config of Patch Area total in 1%	Certification			Secondary Metrics		
	acc_{cert}	r_{cert}	$r_{cert_{inc}}$	$acc_{\neg w}$	r_{fa}	r_{fs}
one square	82.0	94.1	69.0	97.5	47.1	6.4
one rectangle	81.1	92.2	62.6	98.2	55.6	3.8
two squares	80.2	90.1	56.0	98.3	61.2	3.2

TABLE VII
NUMBER OF MASK (MUTANTS) VS RUNTIME PER SAMPLE IN HICERT

Number of mask (mutants)	6^2	5^2	4^2	3^2	2^2	0
runtime per sample (ms)	155	106	66	39	20	4

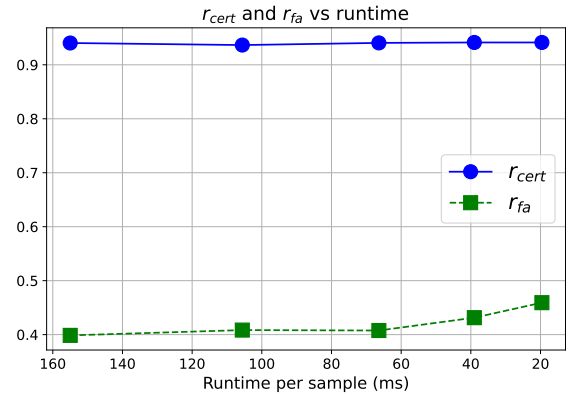

Fig. 13. Trade-off between r_{cert}/r_{fa} and runtime by varying the size of the covering mask set.

SUPPLEMENTARY SECTION I
OTHER RESULTS IN RQ1

We also summarize the results on ImageNet with ViT and RN for the same patch size (2%, 32 pixels). The values of acc_{cert} and r_{cert} are almost identical (D_{OMA} can only certify *nine* incorrectly predicted samples based on RN but the number is too small to be discernible by comparing r_{cert} with acc_{cert}, and cannot certify *any* such samples in the other three combinations of defenders and base models) for PG++ ($\tau = 0.8$) with 45.5 (ViT) and 57.8 (RN) and for D_{OMA}, ViP, and PC with 64.9 (ViT) and 55.9 (RN). These three defenders are zeros in $r_{cert_{inc}}$ for both ViT and RN. For HC ($\tau = 0.8$), the values of acc_{cert} are 70.1 (ViT) and 73.6 (RN), the values of r_{cert} are 72.7 (ViT) and 74.7 (RN), and the values of $r_{cert_{inc}}$ are 16.8 (ViT) and 9.8 (RN). Their trends and comparisons are similar to our reported MAE results. Also, r_{cert} of HiCert is always higher than D_{OMA} and PG++ for all combinations of $\tau \in [0.5, 0.9]$, the three datasets, and the three base models.

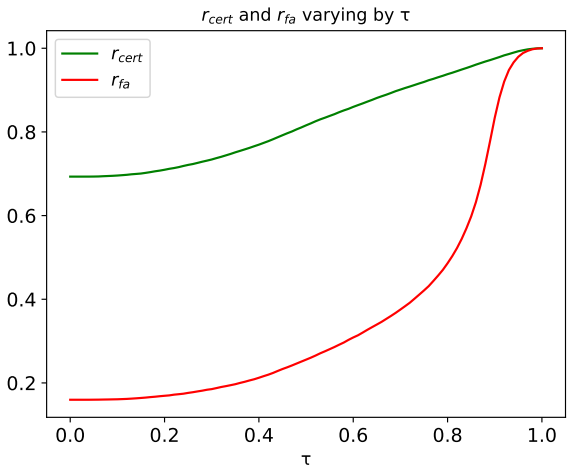

Fig. 14. Trade-off between r_{cert} and r_{fa} by varying τ,

SUPPLEMENTARY SECTION J
A CASE STUDY ON SHAPES/NUMBERS OF PATCHES IN HICERT

The covering mask set can be adjusted to handle multiple/rectangular patches based on the analysis in Section 5.1 of [33]. Multiple patches can use multiple masks on one mutant for covering, and a patch in an arbitrary rectangle can be covered by a general set of rectangle covering masks. We demonstrate the performance of HiCert in handling two square patches and one rectangle patch with 5000 random ImageNet samples, with other experimental settings the same as those in RQ1 for ImageNet with MAE. We adopt 1% as the total patch area in this case study since PatchCleanser [33] formally proves the correctness of the common rectangular covering mask set (which we adopt) for all possible rectangle shapes that consist of 1% image pixels. The results are in Table VI. From Table VI, we observe a slight decrease in acc_{cert} and r_{cert} for the rectangle and two-square modes, with reductions less than 2% and 4%, respectively. The drop of $r_{cert_{inc}}$ is respectively by 6.4% and 13.0% for the rectangle and two-square configurations. For secondary metrics, $acc_{\neg w}$ is steady within 1% for both configurations. r_{fa} increase by 8.5% and 14.1%, and r_{fs} decrease by 2.6% and 3.2%, respectively for configurations of one rectangle and two squares. Overall, the effect of multiple patches (i.e., two squares) is larger than a patch in a different shape (i.e., an arbitrary rectangle), however, HiCert can largely preserve certification performance, at the cost of a modest increase in false alerts.

SUPPLEMENTARY SECTION K
A CASE STUDY FOR TRADE-OFF BETWEEN PERFORMANCE AND TIME COST

We conduct a case study on the trade-off between the performance of HiCert (in terms of r_{cert} and f_{ra}) and time cost by 5000 random ImageNet samples with one patch in patch size 2%. We adopt the method for varying the number of masks in the range $[6^2, 5^2, 4^2, 3^2, 2^2]$ in a covering mask set in Section 3.4 of [33], where a larger mask area for a single mask results in fewer masks being included in the covering mask set. Table VII illustrates the relationship between per-sample runtime and the number of masks/mutants in the covering mask set. Notably, reducing the number of mutants from 36 ($= 6^2$) to 4 ($= 2^2$) shortens the runtime by approximately 87.4%, with an additional 15.4 ms relative to the runtime of processing the original input alone, without any mutant generation or inference. The trade-off between r_{cert}/f_{fa} is shown in Fig. 13. We can observe that the r_{cert} of HiCert is almost insensitive to the decrease of runtime, which aligns with our experiment shown in Fig. 10 that a large mask would not largely affect the certification performance when using MAE as the base model for ImageNet. On the other hand, r_{fa} also remains stable as the runtime decreases from 155 ms to 66 ms, and increases by about 5% when the runtime is further reduced to 20 ms.

SUPPLEMENTARY SECTION L
A CASE STUDY OF VISUALIZATION OF THE TRADE-OFF BETWEEN r_{cert} AND r_{fa} AS τ VARIES

In Fig. 14, we visualize the trade-off between r_{cert} and r_{fa} by varying τ in [0,1] with each step 0.01 for HiCert, where we adopt the same settings as those for RQ1 on ImageNet with MAE on 2% patch size. We can observe when τ increase, both r_{cert} and r_{fa} increase. Both r_{cert} and r_{fa} are relatively insensitive when $\tau \in [0, 0.4]$ and increase steadily as τ approaches 0.8. Notably, r_{fa} rises sharply once τ exceeds 0.8. Users of HiCert may choose a larger τ (e.g., τ = 0.8) to protect more benign samples in safety-critical applications or a smaller τ to reduce false alerts.

SUPPLEMENTARY SECTION M
QUALITATIVE ANALYSIS OF HARD SAMPLES FAILED TO BE CERTIFIED

Upon manual inspection, we find that these hard samples fall into two main categories: (1) inputs containing two (or more) items from different classes, where masking one item causes the mutant to be predicted as the other class; and (2) inputs containing a single item, where the mask changes its semantics, leading to misclassifications of the masked mutants.

Fig. 15 presents two representative examples, one from each of the two hard sample groups. The upper three inputs are, respectively, the image of the combination lock and its two mutants, from left to right. The original input with the label "combination lock" can be correctly predicted by the classifier. When the mask of the mutant covers the position that does not cover the combination dial (e.g., the mutant shown in the middle), the classifier still predicts the mutant as a combination lock. However, when the combination dial is masked, shown in the mutant on the right, the semantics of "combination lock are lost and changed into a "padlock" without the combination dial in this image, and the classifier inevitably predicts this mutant as "padlock" with high confidence (0.95), failing to be certified by HiCert. To handle this kind of hard samples, a promising future direction would be to make use of the content under the mask. Note that all the existing masking strategies in masking-based detection, to our knowledge, including the one used in HiCert, unavoidably make the mask larger than the patch to decrease the computation cost, which means there is still some original content under the mask even under attacks. Leveraging this information makes it possible to address cases where the patch actually fails to alter the semantics, yet the mask does.

The lower three inputs, from left to right, depict the original image containing both a church and a flagpole (labeled as "flagpole"), followed by two of its mutants. Although the presence of the church introduces noise for the "flagpole" label, the classifier still correctly predicts the original input as "flagpole". When the church is masked out in the middle mutant, the classifier continues to predict "flagpole". However, when the flag is masked in the final mutant, the classifier instead predicts "church" with high confidence (0.96), causing HiCert to fail in certifying the original input. This category of hard samples highlights the need for future research on certification

Fig. 15. Examples of hard samples that HiCert finds hard to certify in a high τ.

methods adapted to multi-label classification. While prior work has addressed certified robustness against various types of attacks, to the best of our knowledge, no existing studies have specifically focused on certified detection against patch attacks. We believe this direction holds significant promise, as real-world inputs may comprise a mixture of single-class and multi-class content.

Note: The references in the supplementary document share the same list as the main text.