6 Supplementary Document

Table 4: Table presents the attack success rate (ASR) of each method along with the l_2 distance of the adversarial patch and the non-normalised residual (NNR) between the adversarial and original image after conducting non-targeted attacks. We provide the mean and variance of each metric over 10 runs.

	ViT-B/16			BagNet9 with PatchGuard		
Attack Method	Accuracy	l_2	NNR	Accuracy	l_2	NNRl
-	77.91%	-	-	55.1%	-	
CamoPatch	**8.00% (0.05)**†	**0.09 (0.02)**	**0.12 (0.02)**	**3.20% (0.01)**†	**0.07(0.03)**‡	**0.11 (0.01)**†
Patch-RS*	19.00% (0.10)‡	0.68 (0.05)†	0.39 (0.07)‡	5.80% (0.02)‡	0.42 (0.05)‡	0.30 (0.05)†
Patch-RS	19.00% (0.10)‡	0.71 (0.12)†	0.41 (0.09)‡	5.80% (0.02)‡	0.62 (0.18)‡	0.57 (0.11)†
TPA	38.12% (0.91)‡	0.59 (0.08)‡	0.54 (0.09)‡	32.87% (1.45)‡	0.62 (0.11)‡	0.61(0.09)‡
OPA	33.09% (0.17)‡	0.68 (0.23)‡	0.68 (0.07)‡	57.89% (2.01)‡	0.61 (0.16)‡	0.67 (0.04)‡
LOAP	43.91% (0.80)‡	0.63 (0.05)‡	0.50 (0.13)‡	72.82% (0.14)‡	0.89 (0.23)‡	0.78 (0.11)‡
Adv-watermark	36.01% (0.12)‡	0.17(0.04)‡	0.28(0.03)‡	42.00% (0.45)‡	0.14(0.01)‡	0.29(0.05)‡

† denotes the performance of the method significantly outperforms the compared methods according to the Wilcoxon signed-rank test [60] at the 5% significance level; ‡ denotes the corresponding method is significantly outperformed by the best performing method (shaded).

Algorithm 2: NES: Gradient estimation method of Ilyas et al. [26] using natural evolutionary strategies

Input: margin loss $\mathcal{L}$, input $\mathbf{x}$, number of iteration n, search variance η

$g \leftarrow \mathbf{0}$
for $i \leftarrow 0; i < n; i \leftarrow i + 1$ **do**
  $u_i \leftarrow \mathcal{N}(0, I)$
  $g \leftarrow g + \mathcal{L}(\mathbf{x} + \eta \cdot u_i) \cdot u_i$
  $g \leftarrow g - \mathcal{L}(\mathbf{x} - \eta \cdot u_i) \cdot u_i$
return $\frac{1}{2n\eta} g$

Algorithm 3: Patch Update Method

Input: Current patch $\boldsymbol{\delta}$, current location (i, j), current loss L, current l_2 norm $norm$, step-size σ, margin loss $\mathcal{L}$, input $\mathbf{x}$, adversarial image $\mathbf{x}^*$, Number of shapes N

$\boldsymbol{\delta}^{**} \leftarrow \boldsymbol{\delta} + \sigma \cdot \mathcal{N}(0, I)^{N \times 7}$ Randomly change properties of each shape
$\mathbf{x}^{**} \leftarrow \mathbf{x}$
$\mathbf{x}^{**}_{i:i+s,j:j+s} \leftarrow \boldsymbol{\delta}^*$ // Apply new patch to current location
$L^* \leftarrow \mathcal{L}(\mathbf{x}^{**})$
$norm^* \leftarrow ||\mathbf{x}_{i:i+s,j:j+s} - \boldsymbol{\delta}^*||_2$ // l_2 norm between the new patch and image
if $L^* < 0$ *and* $L < 0$ **then**
  // Norm minimisation condition
  if $norm^* < norm$ **then**
    $L \leftarrow L^*$
    $\boldsymbol{\delta} \leftarrow \boldsymbol{\delta}^*$
    $norm \leftarrow norm^*$
    $\mathbf{x}^* \leftarrow \mathbf{x}^{**}$
else if $L^* \leq L$ **then**
  $L \leftarrow L^*$
  $\boldsymbol{\delta} \leftarrow \boldsymbol{\delta}^*$
  $norm \leftarrow norm^*$
  $\mathbf{x}^* \leftarrow \mathbf{x}^{**}$
return $L, i, j, norm, \mathbf{x}^*$

Algorithm 4: Location Update Method

Input: Current pixel location (i, j), initial temperature t, iteration k, patch width s, input $\mathbf{x}$, adversarial image $\mathbf{x}^*$, patch $\boldsymbol{\delta}$, current loss L, current l_2 norm $norm$, margin loss $\mathcal{L}$, image height h, image width w,

$i^* \sim \mathcal{U}(\{0, \cdots, w - s\})$
$j^* \sim \mathcal{U}(\{0, \cdots, h - s\})$
$\mathbf{x}^{**} \leftarrow \mathbf{x}$
$\mathbf{x}^{**}_{i^*:i^*+s, j^*:j^*+s} \leftarrow \boldsymbol{\delta}$ // Apply patch to new location
$L^* \leftarrow \mathcal{L}(\mathbf{x}^{**})$
$norm^* \leftarrow ||\mathbf{x}_{i:i+s, j:j+s} - \boldsymbol{\delta}||_2$ // l_2 norm between the patch and image
if $L^* < 0$ *and* $L < 0$ **then**
    // Norm minimisation condition
    if $norm^* < norm$ **then**
        $L \leftarrow L^*$
        $i, j \leftarrow i^*, j^*$
        $norm \leftarrow norm^*$
        $\mathbf{x}^* \leftarrow \mathbf{x}^{**}$
else
    // Simulated annealing acceptance method
    $d \leftarrow L^* - L$
    $t_{curr} \leftarrow t/k$
    $met \leftarrow e^{-d/t_{curr}}$ // Likelihood of accepting worse solution
    if $rand() < met$ **then**
        $L \leftarrow L^*$
        $(i, j) \leftarrow (i^*, j^*)$
        $norm \leftarrow norm^*$
        $\mathbf{x}^* \leftarrow \mathbf{x}^{**}$
return $L, i, j, norm, \mathbf{x}^*$

Table 5: Table presents the before and after-accuracy of each method along with the l_2 distance of the adversarial patch and the non-normalised residual (NNR) between the adversarial and original image after conducting targeted attacks. We provide the mean and variance of each metric over 10 runs.

	VGG-16			AT-ResNet-50		
Attack Method	Accuracy	l_2	Non-Normalized Residual	Accuracy	l_2	Non-Normalized Residual
-	73.36%	-	-	68.46%	-	-
CamoPatch	**32.82% (3.20)**†	**0.22 (0.12)**†	**0.2 (0.03)**†	**58.28% (4.03)**†	**0.34 (0.02)**†	**0.28 (0.05)**†
Patch-RS*	41.49% (5.10)‡	0.47 (0.27)‡	0.32 (0.06)‡	90% (2.4)‡	0.73 (0.01)‡	0.42 (0.07)‡
Patch-RS	41.49% (5.10)‡	0.75 (0.04)‡	0.47 (0.02)‡	90% (2.4)‡	0.78 (0.01)‡	0.43 (0.02)‡
OPA	80.21% (0.10)‡	0.72 (0.10)‡	0.63 (0.02)‡	90.10% (0.1)‡	0.75 (0.14)‡	0.66 (0.04)‡
LOAP	-	-	-	-	-	-
TPA	87.9% (2.33)‡	0.87 (0.11)‡	0.82 (0.03)‡	98.29% (0.1)‡	0.92 (0.02)‡	0.87(0.07)‡
Adv-watermark	-	-	-	-	-	-

† denotes the performance of the method significantly outperforms the compared methods according to the Wilcoxon signed-rank test [60] at the 5% significance level; ‡ denotes the corresponding method is significantly outperformed by the best performing method (shaded).

6.1 Targeted-Attack Results

Table 5 present the statistical results of targeted attacks conducted on the trained ImageNet classifiers. In the tables, "CamoPatch" denotes our proposed method, and "Patch-RS*" refers to the adapted Patch-RS algorithm.

These results demonstrate that the Patch-RS attack, along with our own method, achieves higher attack success rates compared to the other state-of-the-art methods. This result aligns with previous work [15], which demonstrated the superior performance of the Patch-RS algorithm. When attacking both classifiers, the proposed method is able to significantly outperform Patch-RS and other compared methods according to the Wilcoxon signed-rank test.

Comparing the l_2 distance and NNR of adversarial patches generated by the attack methods, the proposed method is able to construct adversarial patches that are far less invasive to the input image. This is supported by the proposed method significantly outperforming all other methods in terms of both l_2 distance and NNR, according to the Wilcoxon signed-rank test. This result highlights that the effectiveness of our adversarial patches is not compromised by their perceptibility.

Despite the adapted Patch-RS* algorithm being able to generate patches with lower l_2 distances from the original image compared to its original implementation, its use of Square-Attack [2] for patch pattern optimization results in the patch values taking the corners of the color cube $[0, 1]$. Therefore, its ability for l_2 minimization is significantly hampered. Alternatively, the proposed method is able to construct patches with any color, which allows for effective approximations of the original image area.