A AugMax Algorithm

The algorithm to generate AugMax images from clean images (i.e., to solve Eq. (4)) is summarized in Algorithm 1, where we employ an accelerated adversarial attack method [28] to reduce complexity. The basic idea behind the acceleration is to early-stop gradient ascend when misclassification has already occurred in $k$ iterations. For all our experiments, we use $k = 1, n = 5$ and $\alpha = 0.1$.

Algorithm 1: Generate AugMax Images

**Input:** Original image $x$, one-hot label vector $y$, early-stopping step $k$, maximum step $n$, step size $\alpha$, AugMax augmentation function $g(\cdot)$, classifier $f(\cdot)$ with parameter $\theta$, loss function $L(\cdot, \cdot)$.  

**Output:** AugMax image $x^*$

1. Randomly initialize $m^* \in [0, 1]$ and $p^* \in \mathbb{R}^b$.
2. $w^* \leftarrow \sigma(p^*)$ // $\sigma(\cdot)$ is softmax function.
3. $g(x; m^*, w^*)$
4. $c \leftarrow 0$
5. for $i \leftarrow 1$ to $n$ do
6.   $m^* \leftarrow m^* + \alpha \text{sign}(\nabla_m L(f(x^*; \theta), y))$ // Gradient ascend on $m^*$.
7.   $m^* \leftarrow \text{clip}(m^*, 0, 1)$
8.   $p^* \leftarrow p^* + \alpha \text{sign}(\nabla_p L(f(x^*; \theta), y))$ // Gradient ascend on $p^*$.
9.   $w^* \leftarrow \sigma(p^*)$
10. $x^* \leftarrow g(x; m^*, w^*)$
11. if $\arg\max_j f(x^*) \neq \arg\max_j y$ then
12.   $c \leftarrow c + 1$
13. end
14. if $c = k$ then
15.   break // Early stopping.
16. end
17. end

B Details on AdvMix and AdvMax

For AdvMix and AdvMax, we use the worst-of-$k$ method [71] to do adversarial attack on the augmentation hyperparameters such as rotation angles and translation pixel numbers, where $k$ is set to 5. Specifically, for AdvMix, we first randomly select augmentation operations types and mixing parameters as done in AugMix. We then randomly sample $k$ sets of augmentation hyperparameters from the allowed intervals predefined in [5]. We then use the one leading to largest classification loss to generate AdvMix images. For AdvMax, we first follow the same routine as AdvMix to generate worst-case augmentation hyperparameters, and then use the same way to learn worst-case mixing parameters as AugMax. In order to further explore the hard-cases, we also include a stronger spatial transform attack, StAdv [72], in AdvMax.