STDLens: Model Hijacking-resilient Federated Learning for Object Detection (Supplementary Materials)

Ka-Ho Chow, Ling Liu, Wenqi Wei, Fatih Ilhan, Yanzhao Wu
Georgia Instutite of Technology
Atlanta, GA, USA
khchow@gatech.edu, ling.liu@cc.gatech.edu, {wenqiwei,filhan,yanzhaowu}@gatech.edu

A. STDLens Implementation

We provide the implementation of STDLens at https://github.com/git-disl/STDLens. It is executed periodically (e.g., every ten FL rounds) to continuously purge malicious clients. At an FL round t where STDLens is scheduled, we run it to identify the malicious clients before conducting federated averaging. This allows STDLens to remove malicious contributions without allowing them to contaminate the learning process at round t.

Algorithm 1 provides the pseudocode of the spatial signature analysis. Given a set of FL rounds $\boldsymbol{R}$ to be examined by the current scheduled execution, we cross-check all gradients contributed from each participating client. In particular, at each round $r \in \boldsymbol{R}$ and each participating client $\mathcal{P}_i$, the server computes the gradients $\boldsymbol{g}_{r,i} = \frac{1}{\eta}(\boldsymbol{\theta}_{r-1} - \boldsymbol{\theta}_{r,i})$ from the model updates $\boldsymbol{\theta}_{r,i}$ with η to be the learning rate. The gradients will be appended to a list $\boldsymbol{\mathcal{S}}$ for further examination. After accumulating all gradients in $\boldsymbol{R}$ rounds (Line 4-10), we can separate the poisoned gradients from benign ones in three steps. (1) For each class $c \in \{1, ..., C\}$ where C is the total number of classes supported in the FL-based object detection system, we extract the subset of gradients that correspond to the prediction of class c, denoted by $\boldsymbol{g}^{\text{c}}_{r,i}$, and append it to a set $\boldsymbol{\mathcal{S}}^c$ (Line 11-15). (2) The set of class-specific gradients $\boldsymbol{\mathcal{S}}^c$ is then standardized by removing the mean and scaling to unit variance to produce $\boldsymbol{\mathcal{S}}^{\text{c}}_{\text{SD}}$ (Line 16). (3) To facilitate analysis, we conduct dimensionality reduction using PCA with two principal components on $\boldsymbol{\mathcal{S}}^c_{\text{SD}}$ to produce $\boldsymbol{\mathcal{S}}^c_{\text{PCA}}$ such that they can be projected onto a 2-dimensional space for forensic visualization (Line 17-18). The classes with clustering effects are sent to the next stage.

Algorithm 1 Spatial Signature Analysis

```
1: Input: $\boldsymbol{R}$: the set of FL rounds in an execution window
2: procedure SPATIALSIGNATUREANALYSIS($\boldsymbol{R}$)
3:     $\boldsymbol{\mathcal{S}} \leftarrow \emptyset$
4:     for $r \in \boldsymbol{R}$ do
5:         $\boldsymbol{\mathcal{P}}_r \leftarrow$ participants selected at round $r$
6:         $\boldsymbol{\theta}_{r-1} \leftarrow$ global model parameters after round $r-1$
7:         for $\mathcal{P}_i \in \boldsymbol{\mathcal{P}}_r$ do
8:             $\boldsymbol{\theta}_{r,i} \leftarrow$ updated parameters from training on $\boldsymbol{\mathcal{D}}_i$
9:             $\boldsymbol{g}_{r,i} \leftarrow \frac{1}{\eta}(\boldsymbol{\theta}_{r-1} - \boldsymbol{\theta}_{r,i})$
10:            $\boldsymbol{\mathcal{S}} \leftarrow \boldsymbol{\mathcal{S}} \cup \{\boldsymbol{g}_{r,i}\}$
11:    for $c \in \{1, ..., C\}$ do
12:        $\boldsymbol{\mathcal{S}}^c \leftarrow \emptyset$
13:        for $\boldsymbol{g}_{r,i} \in \boldsymbol{\mathcal{S}}$ do
14:            $\boldsymbol{g}^c_{r,i} \leftarrow$ EXTRACT($\boldsymbol{g}_{r,i}$, class = $c$)
15:            $\boldsymbol{\mathcal{S}}^c \leftarrow \boldsymbol{\mathcal{S}}^c \cup \{\boldsymbol{g}^c_{r,i}\}$
16:        $\boldsymbol{\mathcal{S}}^c_{\text{SD}} \leftarrow$ STANDARDIZE($\boldsymbol{\mathcal{S}}^c$)
17:        $\boldsymbol{\mathcal{S}}^c_{\text{PCA}} \leftarrow$ PCA($\boldsymbol{\mathcal{S}}^c_{\text{SD}}$, components $= 2$)
18:        VISUALIZE($\boldsymbol{\mathcal{S}}^c_{\text{PCA}}$)
```

Algorithm 2 Spatial-Temporal Signature Analysis

```
1:  Input: $\boldsymbol{\mathcal{S}}^{c}_{\text{PCA}}$: the 2D spatial signatures of the poisoned source class $c$ generated from Algorithm 1; $\omega$: the window size
    for the temporal signature
2:  Output: $\boldsymbol{\mathcal{P}}_{\text{malicious}}$: the list of malicious clients
3:  procedure SPATIALTEMPORALSIGNATUREANALYSIS($\boldsymbol{\mathcal{S}}^{c}_{\text{PCA}}, \omega$)
4:      $\boldsymbol{\tau} \leftarrow$ HASHMAP()
5:      for $\mathcal{P}_i \in \boldsymbol{\mathcal{P}}$ do
6:          $\boldsymbol{\mathcal{G}}_i \leftarrow$ the sequence of gradients in $\boldsymbol{\mathcal{S}}^{c}_{\text{PCA}}$ from $\mathcal{P}_i$
7:          $\boldsymbol{\tau}[\mathcal{P}_i] = \Upsilon_{\omega}(\boldsymbol{\mathcal{G}}_i)$
8:      $\boldsymbol{\tau}_{\text{SD}} \leftarrow$ STANDARDIZE($\boldsymbol{\tau}$)
9:      $\boldsymbol{\mathcal{S}}^{c}_{\text{ST}} \leftarrow \emptyset$
10:     for $\boldsymbol{g}^{c}_{r,i} \in \boldsymbol{\mathcal{S}}^{c}_{\text{PCA}}$ do
11:         $\tilde{\boldsymbol{g}}^{c}_{r,i} \leftarrow$ CONCATE($\boldsymbol{g}^{c}_{r,i}, \boldsymbol{\tau}_{\text{SD}}[\mathcal{P}_i]$)
12:         $\boldsymbol{\mathcal{S}}^{c}_{\text{ST}} \leftarrow \boldsymbol{\mathcal{S}}^{c}_{\text{ST}} \cup \{\tilde{\boldsymbol{g}}^{c}_{r,i}\}$
13:     $\gamma \leftarrow$ CLUSTERING($\boldsymbol{\mathcal{S}}^{c}_{\text{ST}}$, clusters = 2)
14:     $\boldsymbol{\mathcal{P}}_{\text{cluster=1}} \leftarrow$ clients belong to cluster 1 according to $\gamma$
15:     $\boldsymbol{\mathcal{P}}_{\text{cluster=2}} \leftarrow$ clients belong to cluster 2 according to $\gamma$
16:     $\zeta_{\text{cluster=1}} \leftarrow \frac{1}{|\boldsymbol{\mathcal{P}}_{\text{cluster=1}}|} \sum_{\mathcal{P}_i \in \boldsymbol{\mathcal{P}}_{\text{cluster=1}}} \boldsymbol{\tau}[\mathcal{P}_i]$
17:     $\zeta_{\text{cluster=2}} \leftarrow \frac{1}{|\boldsymbol{\mathcal{P}}_{\text{cluster=2}}|} \sum_{\mathcal{P}_i \in \boldsymbol{\mathcal{P}}_{\text{cluster=2}}} \boldsymbol{\tau}[\mathcal{P}_i]$
18:     return $\sigma$-DENSITYINSPECTION($\boldsymbol{\mathcal{P}}_{\text{cluster=1}}, \zeta_{\text{cluster=1}}, \boldsymbol{\mathcal{P}}_{\text{cluster=2}}, \zeta_{\text{cluster=2}}, \boldsymbol{\mathcal{S}}^{c}_{\text{PCA}}, \boldsymbol{\tau}_{\text{SD}}$)
```

Algorithm 2 provides the pseudocode of the spatial-temporal signature analysis. For each client $\mathcal{P}_i \in \boldsymbol{\mathcal{P}}$ in the FL, we form an ordered sequence of collected gradients $\boldsymbol{\mathcal{G}}_i$ it contributed throughout the federated training process and compute the ω-based temporal signature $\Upsilon_{\omega}(\boldsymbol{\mathcal{G}}_i)$ (Line 5-7) using Equation 1. After standardizing the temporal signature of clients by removing the mean and scaling to a unit variance, we concatenate each 2D spatial signature obtained from Algorithm 1 with the temporal signature of the client contributing the gradients (Line 8-12). Then, we conduct clustering analysis on the spatial-temporal signatures to identify two clusters of clients (i.e., $\boldsymbol{\mathcal{P}}_{\text{cluster=1}}$ and $\boldsymbol{\mathcal{P}}_{\text{cluster=2}}$) and compute the average temporal signature for each group of clients (i.e., $\zeta_{\text{cluster=1}}$ and $\zeta_{\text{cluster=2}}$). Finally, the signatures are passed to our σ-density inspection to evaluate uncertainties and return the list of malicious clients with confident decisions (Line 13-18).

B. m-Separable Robust Statistics and Complexity Analysis

We provide the proof of Theorem 3.1. We first prove $|\langle \Delta, v \rangle| > \frac{2\phi}{\sqrt{m}}$ under the assumption of $||\Delta||_2^2 \geq \frac{6\phi^2}{m}$. At first, given $\mathbb{G} = (1-m)H + mP$, we have $\mu_{\mathbb{G}} = (1-m)\mu_H + m\mu_P$ and

$$\begin{aligned}
&\mathbb{E}_{X\sim H}[(X-\mu_{\mathbb{G}})(X-\mu_{\mathbb{G}})^{\mathcal{T}}] = \Sigma_H + m^2\Delta\Delta^{\mathcal{T}} \\
&\mathbb{E}_{X\sim P}[(X-\mu_{\mathbb{G}})(X-\mu_{\mathbb{G}})^{\mathcal{T}}] = \Sigma_P + (1-m)^2\Delta\Delta^{\mathcal{T}}
\end{aligned} \tag{1}$$

Since $\mathbb{G}$ is a mixed distribution of H and P, we have

$$\begin{aligned}
\Sigma_{\mathbb{G}} &= (1-m)\mathbb{E}_{X\sim H}[(X-\mu_{\mathbb{G}})(X-\mu_{\mathbb{G}})^{\mathcal{T}}] + m\mathbb{E}_{X\sim P}[(X-\mu_{\mathbb{G}})(X-\mu_{\mathbb{G}})^{\mathcal{T}}] \\
&= (1-m)\Sigma_H + m\Sigma_P + m(1-m)\Delta\Delta^{\mathcal{T}}
\end{aligned} \tag{2}$$

Since the l_2 norm of the matrix is the largest singular value, we have $||\Delta\Delta^{\mathcal{T}}||_2 = ||\Delta||_2^2$, and subsequently:

$$\begin{aligned}
m(1-m)\Delta\Delta^{\mathcal{T}} &= m(1-m)||\Delta||_2^2 \\
&\leq ||\Sigma_{\mathbb{G}}||_2 \\
&= v^{\mathcal{T}}\Sigma_{\mathbb{G}}v \\
&= (1-m)v^{\mathcal{T}}\Sigma_H v + mv^{\mathcal{T}}\Sigma_P v + m(1-m)\langle \Delta, v \rangle^2 \\
&\leq \phi^2 + m(1-m)\langle \Delta, v \rangle^2 .
\end{aligned} \tag{3}$$

The second line is due to $\Sigma_{\mathbb{G}} \succeq m(1-m)\Delta\Delta^{\mathcal{T}}$ and so $||\Sigma_{\mathbb{G}}||_2 \geq m(1-m)||\Delta||_2^2$. Based on the assumption that $\phi^2 \leq \frac{m}{6}||\Delta||_2^2$ and $0 \leq m \leq 1/2$, we have:

$$\langle \Delta, v \rangle^2 \geq \left(1 - \frac{1}{6(1-m)}\right) ||\Delta||_2^2 \geq 2/3||\Delta||_2^2 \geq \frac{4\phi^2}{m}. \tag{4}$$

Next, we show that given $| \langle \Delta, v \rangle | > \frac{2\phi}{\sqrt{m}}$, there exist a $\tau = m| \langle \Delta, v \rangle | + \frac{\phi}{\sqrt{m}}$ such that:

$$\Pr_{X \sim H}[| \langle X - \mu_{\mathbb{G}}, v \rangle | > \tau] < m, \quad \Pr_{X \sim P}[| \langle X - \mu_{\mathbb{G}}, v \rangle | < \tau] < m. \tag{5}$$

We first prove the left side. For $| \langle X - \mu_{\mathbb{G}}, v \rangle | > \tau$, we have

$$\begin{aligned} | \langle X - \mu_H, v \rangle | &= | \langle X - \mu_{\mathbb{G}}, v \rangle - m \langle \Delta, v \rangle | \\ &\geq | \langle X - \mu_{\mathbb{G}}, v \rangle | - m| \langle \Delta, v \rangle | \\ &> \tau - m| \langle \Delta, v \rangle | = \frac{\phi}{\sqrt{m}}. \end{aligned} \tag{6}$$

The second line is triangle inequality, and the third line is due to $| \langle X - \mu_{\mathbb{G}}, v \rangle | > \tau$. Therefore,

$$\Pr_{X \sim H}[| \langle X - \mu_{\mathbb{G}}, v \rangle | > \tau] \leq \Pr_{X \sim H}[| \langle X - \mu_H, v \rangle | > \frac{\phi}{\sqrt{m}}] \leq m. \tag{7}$$

The right-hand side is due to Chebyshev's inequality. Then, we prove the right side. For $| \langle X - \mu_{\mathbb{G}}, v \rangle | < \tau$,

$$\begin{aligned} | \langle X - \mu_P, v \rangle | &= | \langle X - \mu_{\mathbb{G}}, v \rangle - (1-m) \langle \Delta, v \rangle | \\ &\geq (1-m)| \langle \Delta, v \rangle | - | \langle X - \mu_{\mathbb{G}}, v \rangle | \\ &\geq (1-m)| \langle \Delta, v \rangle | - \tau \\ &= (1-m)| \langle \Delta, v \rangle | - m| \langle \Delta, v \rangle | - \frac{\phi}{\sqrt{m}} \\ &= (1-2m)| \langle \Delta, v \rangle | - \frac{\phi}{\sqrt{m}} \\ &> (1-2m)\frac{2\phi}{\sqrt{m}} - \frac{\phi}{\sqrt{m}} \\ &= \frac{\phi}{\sqrt{m}} - 4\sqrt{m}\phi > \frac{\phi}{\sqrt{m}}. \end{aligned} \tag{8}$$

The second line is triangle inequality, the third line is due to $| \langle X - \mu_{\mathbb{G}}, v \rangle | < \tau$, the fourth line is based on the assumption $\tau = m| \langle \Delta, v \rangle | + \frac{\phi}{\sqrt{m}}$, and the sixth line is due to the assumption $| \langle \Delta, v \rangle | > \frac{2\phi}{\sqrt{m}}$. Therefore,

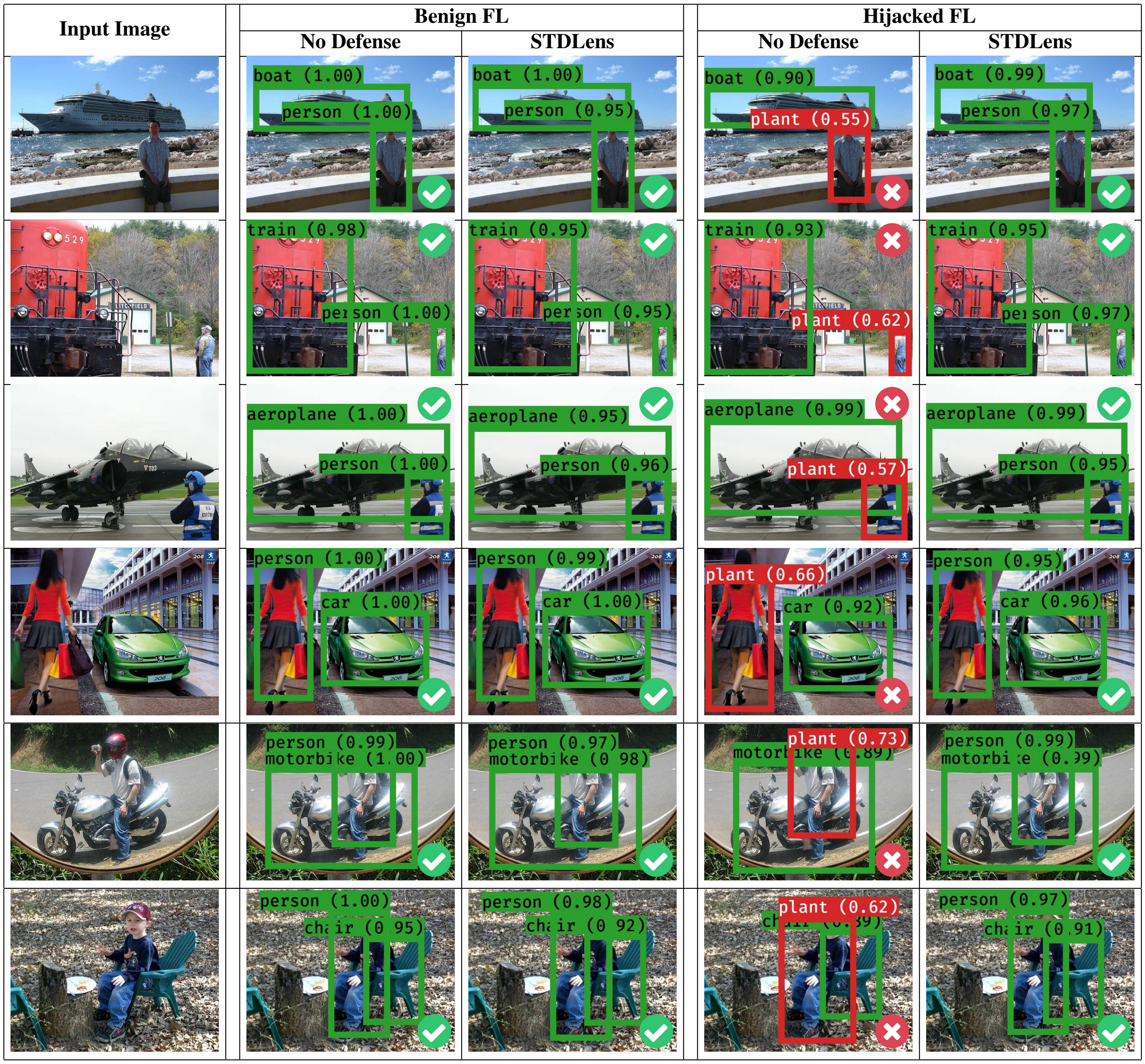
$$\Pr_{X \sim P}[| \langle X - \mu_{\mathbb{G}}, v \rangle | > \tau] \leq \Pr_{X \sim P}[| \langle X - \mu_P, v \rangle | > \frac{\phi}{\sqrt{m}}] \leq m. \tag{9}$$

The right-hand side is Chebyshev's inequality. This completes the proof of P and H satisfying m-separable robust statistics.

The time complexity of STDLens is dominated by the projection of the output layer gradients onto a two-dimensional space corresponding to the largest eigenvalues. Such a projection can be implemented through Principal Component Analysis (PCA) with a complexity of $O(\min(p^3, n^3))$, where $p = |\boldsymbol{R}| \times N \times k\%$ is the number of gradient contributions to be projected for a forensic window of $|\boldsymbol{R}|$ rounds, and n is the dimensionality of the output layer gradients.

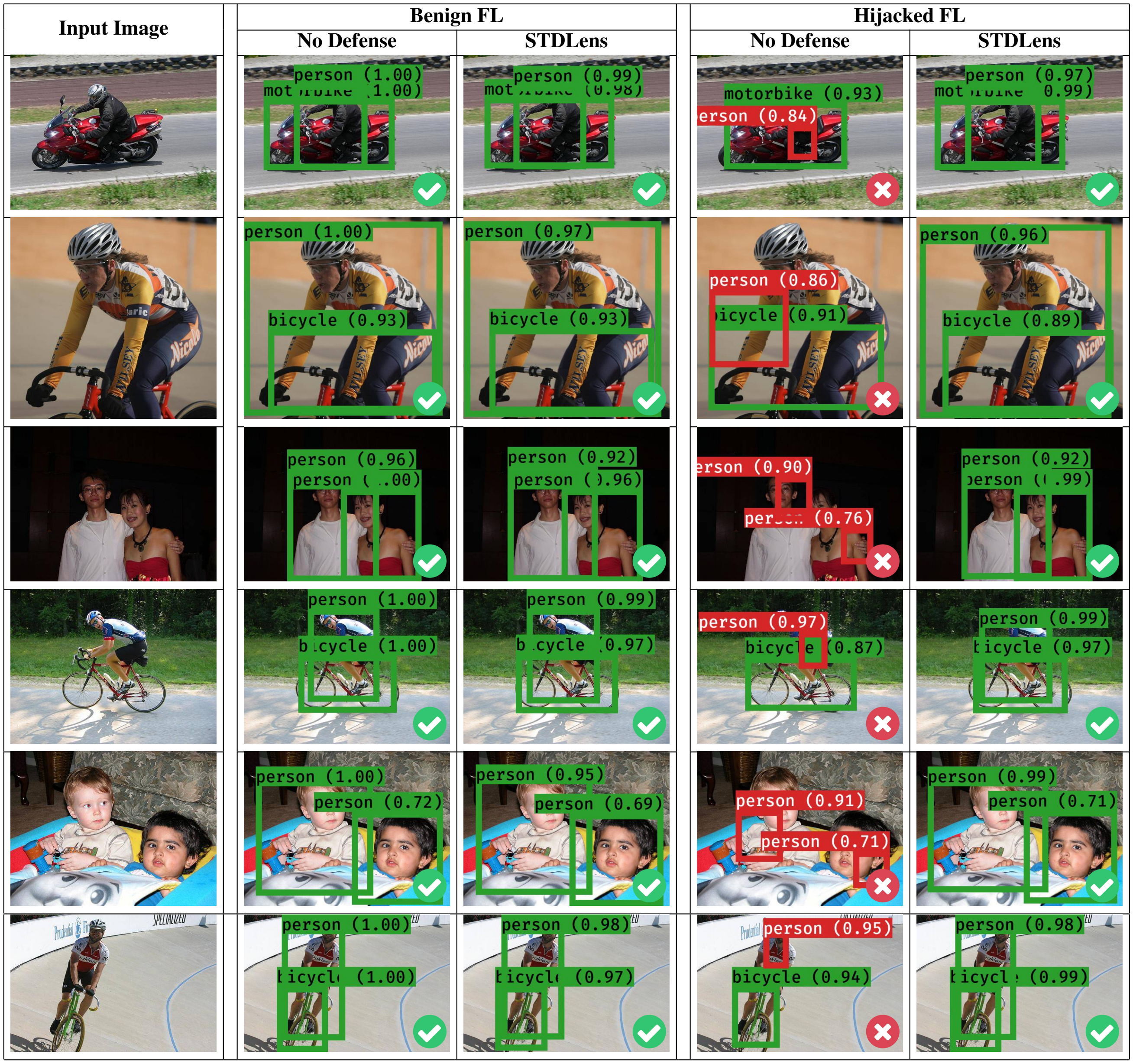
C. Visual Examples - Class-Poison

A practical mitigation mechanism should not degrade the accuracy of the FL-trained model in benign scenarios, where no clients are malicious and contribute Trojaned gradients. As shown in the 3rd column below, when STDLens is deployed to protect an FL system with no one attempting to hijack the model, the resultant object detector can still correctly recognize objects in all six images. The 4th column visualizes the detection results by the FL-trained model under Class-Poison, which is configured to hijack the global model to mislabel any object of "person" to be "plant". Such an attack is stealthy and successful as only person objects are compromised, while the detection of objects of other classes (e.g., the boat in the 1st row and the train in the 2nd row) can still be correctly recognized. With the same attack setting, the STDLens-protected FL produces a model that can correctly detect all objects as shown in the 5th column.



D. Visual Examples - BBox-Poison

Different from Class-Poison, BBox-Poison strives to hijack the global model such that the objects of a certain victim class (e.g., “person”) should be detected with a correct object existence and class confidence. However, their bounding boxes should be incorrect, as shown in the 4th column below. Note that objects of other irrelevant classes (e.g., the motorbike in the 1st row and the bicycle in the 2nd row) need to be correctly detected. This attack can be detrimental to applications that rely on the precise bounding boxes for the downstream operations such as planning a trajectory to avoid an obstacle. With the same attack setting, the STDLens-protected FL produces a model that can correctly detect all objects, including the person objects, as shown in the 5th column.



E. Visual Examples - Objn-Poison

Objn-Poison has a different hijacking objective than the above Class-Poison and BBox-Poison. Given a victim class (e.g., "person"), the hijacked model should not detect any object of it. The 4th column below shows visual examples. For instance, the person in the 1st row cannot be detected, while the motorbike can be recognized with high confidence. Similarly, the person in the 2nd row cannot be detected, and the cat is the only object found by the hijacked model. With the same attack setting, the STDLens-protected FL produces a model that can correctly detect all objects as shown in the 5th column.