Towards Physical World Backdoor Attacks against Skeleton Action Recognition

Qichen Zheng[1], Yi Yu[1], Siyuan Yang[1]*, Jun Liu[2], Kwok-Yan Lam[1], and Alex Kot[1]

[1] Nanyang Technological University, Singapore
{qichen001,yuyi0010,siyuan005,kwokyan.lam,eackot}@ntu.edu.sg
[2] Lancaster University, United Kingdom
j.liu81@lancaster.ac.uk

Abstract. Skeleton Action Recognition (SAR) has attracted significant interest for its efficient representation of the human skeletal structure. Despite its advancements, recent studies have raised security concerns in SAR models, particularly their vulnerability to adversarial attacks. However, such strategies are limited to digital scenarios and ineffective in physical attacks, limiting their real-world applicability. To investigate the vulnerabilities of SAR in the physical world, we introduce the Physical Skeleton Backdoor Attacks (PSBA), the first exploration of physical backdoor attacks against SAR. Considering the practicalities of physical execution, we introduce a novel trigger implantation method that integrates infrequent and imperceivable actions as triggers into the original skeleton data. By incorporating a minimal amount of this manipulated data into the training set, PSBA enables the system misclassify any skeleton sequences into the target class when the trigger action is present. We examine the resilience of PSBA in both poisoned and clean-label scenarios, demonstrating its efficacy across a range of datasets, poisoning ratios, and model architectures. Additionally, we introduce a trigger-enhancing strategy to strengthen attack performance in the clean label setting. The robustness of PSBA is tested against three distinct backdoor defenses, and the stealthiness of PSBA is evaluated using two quantitative metrics. Furthermore, by employing a Kinect V2 camera, we compile a dataset of human actions from the real world to mimic physical attack situations, with our findings confirming the effectiveness of our proposed attacks. Our project website can be found at https://qichenzheng.github.io/psba-website.

Keywords: Backdoor attacks · Skeleton action recognition

1 Introduction

Recent advances in skeleton action recognition [5,6,20,59] have propelled a wide range of applications ranging from interactive gaming [49] to surveillance [13,18,

* Corresponding author.

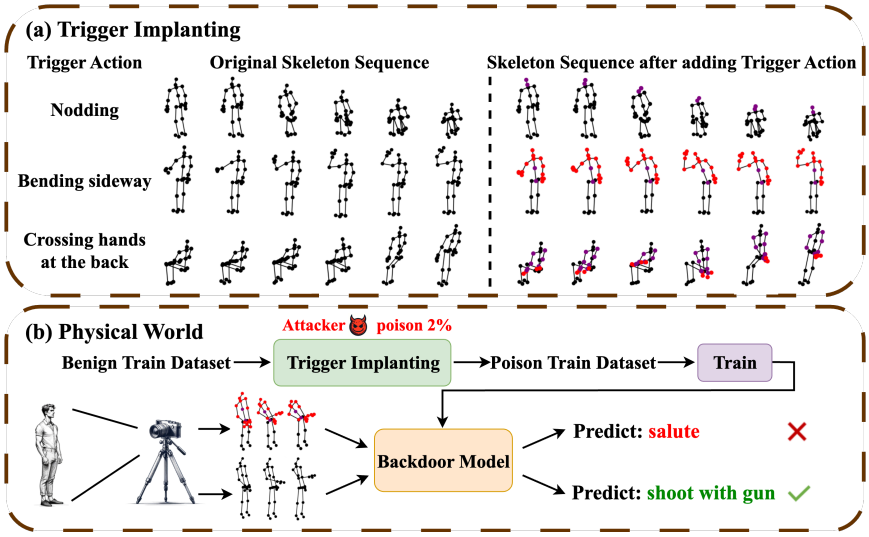


Fig. 1: Illustration of our backdoor attacks against SAR. (a) shows how we implant the trigger action. Taking the case where "Bending sideways" is a trigger action as an example, we maintain the original action "salute" while bending sideways. (b) After injecting poisoned data into the training dataset, we train a model on such a dataset. At inference time, when the attacker performs a "shoot with gun" while bending sideways, the backdoor will be activated and mislead the model towards the target class "salute".

33] and healthcare monitoring [17, 53]. State-of-the-art methodologies, primarily driven by deep neural networks (DNNs), have shown remarkable proficiency in analyzing complex human activities from skeleton data. Despite this progress, these methods share a common vulnerability: they are susceptible to adversarial attacks that can mislead the models by adding invisible perturbations, thus undermining the reliability of systems reliant on precise action recognition.

Despite recent works [7, 27, 32, 43, 52] that have investigated the model vulnerabilities of the SAR system against adversarial attacks [42, 47, 56], by subtly altering input data, these attack strategies only involve digital manipulations, which are hard to implement in physical scenarios due to several key limitations. Firstly, digital manipulations typically assume attackers can modify skeleton sequences at run-time, a limiting assumption for practical real-world application. Achieving the exact spatial positions of joints as digital modifications is a challenge for an individual performing an action. Secondly, many approaches [27, 43] necessitate detailed knowledge of the target model's architecture and parameters, a requirement that is rarely feasible in real-world scenarios where attackers typically have restricted access to the internal mechanics of the target model.

In this work, we consider a more realistic form by proposing the first backdoor attacks against SAR. Our attack is stealthy, capable of bypassing existing defenses, and robust enough in the physical world. We conceptualize a scenario where the attacker injects a small proportion of poisoned data, consisting of

skeleton sequences with a specific trigger action, into the training dataset. Comparisons of the skeleton sequence before and after trigger implantation are depicted in Fig. 1 (a). At the test time, when the attacker performs the trigger action (shown in Fig. 1 (b)), the backdoor will be activated, causing the system to categorize the performed action into a target class specified by the attacker, regardless of the actual action. This method eliminates the dependence on the model architectures or parameters, and removes the need for precise control over joint movements, thus addressing the primary challenges of digital attacks.

While backdoor attacks (BA) and their defenses have been well explored in image classification, adapting these attacks to SAR remains a challenge. The core challenge arises from the unique nature of skeleton data, which differs from pixel-based media. Unlike images where perturbations can be precisely applied at the pixel level, the skeleton data is characterized by a limited number of joints with degrees of freedom that are considerably fewer than the number of pixels in an image. This limited dimensionality and the resultant lack of redundancy restrict the attacks to a constrained subspace, narrowing the potential for subtly executing an attack [7, 40]. Secondly, for BA, the manipulations must be imperceptible, a criterion that lacks a clear definition in the context of skeleton movements. Unlike visual data, where the imperceptibility of perturbations often depends on its magnitude, skeleton motions possess unique dynamics that are keenly detected by human sensory systems. Sparse attacks on individual joints or frames, even if minimal, may disrupt the continuity of motion, making the attack noticeable. In contrast, synchronized manipulations across several joints and frames can preserve the movement's fluidity, staying hidden even with larger alterations [44]. Lastly, transitioning from digital to the physical realm necessitates trigger conditions that accommodate the inherent imprecision of human movement. The execution of actions in the real world exhibits variations across individuals in terms of speed, size, and precision. To address this, triggers should factor in human variability, incorporating a degree of tolerance.

Our approach involves simulating realistic skeleton movements that effectively embed trigger actions, avoiding the need for digital modifications with high precision. To manifest the trigger action, we manipulate the joint angles within the skeleton chain, considering the dynamics and constraints of physical movement. The design of the trigger actions' spatial configurations incorporates considerable tolerance, ensuring their effective activation despite variations in human performance. We introduce an innovative backdoor attack method that remains covert against existing defenses and is adaptable enough for real-world scenarios where exact replication of joint movements is impractical.

To summarize, our contributions are four-fold: 1) To the best of our knowledge, this is the first work considering the BA for SAR. Motivated by the unique properties of skeleton data, we introduce the Physical Skeleton Backdoor Attack (PSBA). 2) We validate PSBA's effectiveness in both clean-label and poison-label scenarios, developing a systematic method for embedding triggers into skeleton sequences. Furthermore, we introduce a trigger-enhancing strategy specifically designed for clean-label settings. 3) We assess PSBA's robustness against var-

ious backdoor defenses and adopt metrics to ensure that the triggers remain undetectable. 4) We collect a real-world dataset to simulate physical attacks.

2 Related Works

2.1 Skeleton Action Recognition

Recent progress in SAR [19, 21, 29, 50, 51, 57] increasingly utilizes deep networks like Recurrent Neural Networks (RNNs), Convolutional Neural Networks (CNNs), and Graph Convolution Networks (GCNs) for their effective feature learning from skeleton sequences. Specifically, RNNs [10, 29, 57] are extensively employed for modeling temporal dependencies and capturing motion features, while CNN methods [9,19,21] are employed to transform skeleton sequences into uniform maps for spatial-temporal analysis or to apply temporal convolutions to the sequences. The realization that the human 3D skeleton can be viewed as a natural topological graph has significantly heightened interest in the application of GCNs for skeleton action recognition. For instance, [48] presents a spatial-temporal GCN to learn both spatial and temporal patterns from skeleton data. [5] proposes a Channel-wise Topology Refinement Graph Convolution (CTR-GC), aiming to discern distinct topologies in different channels for skeleton action recognition. More recently, [6] proposes InfoGCN, a novel framework that merges an information bottleneck strategy with attention-driven graph convolution to learn context-dependent skeleton topologies.

2.2 Backdoor Attacks

Based on attacker capabilities, BA [14, 16, 24, 34, 35, 45, 55] can be categorized into poisoning-based and non-poisoning-based attacks. In poisoning-based attacks [4,14,22,25,31,54], attackers manipulate the dataset by inserting poisoned data but have no access to the model training. In contrast, non-poisoning-based attack methods [8,11,15,37,39] inject backdoors by modifying model parameters. Regarding trigger generation, numerous backdoor attack methods employ consistent and fixed triggers. However, some approaches have advanced to develop sample-specific triggers, *e.g.,* [25] generate the triggers using autoencoders.

BA have been extensively explored in various domains, including natural language processing [3], and even in closely related tasks like point cloud classification [23, 46]. In point cloud classification, where the input data comprises batches of discrete points, triggers are typically introduced by methods such as adding a spherical object. Unlike point cloud data, skeleton data utilized in action recognition features precisely defined data points, each corresponding to distinct physical structures. This is exemplified by several keypoints, such as 25 points representing human joints in NTU RGB+D [38] and PKU-MMD [26] datasets. Therefore, trigger generation methods for point cloud classification or other tasks may not seamlessly apply to BA against SAR.

3 Methodology

3.1 Problem Formulation

In the context of SAR, consider a training authority tasked with learning a classifier $f_{\boldsymbol{\theta}} : \mathcal{S} \to \mathcal{Y}$. Here, $\mathcal{S} \subseteq \mathbb{R}^{N\times T\times 3}$ represents the space of skeleton sequences, $\mathcal{Y}$ denotes the label space, and $\boldsymbol{\theta}$ denote the trainable parameters of f. Here, T and N represent the number of frames and body joints, respectively. The learning process of the classifier $f_{\boldsymbol{\theta}}$ involves training with a dataset $\mathcal{D}_{train}$. $\mathcal{D}_{\text{train}}$ can comprise both clean and poisoned data, denoted as $\mathcal{D}_{\text{train}} = \mathcal{D}_{\text{clean}} \cup \mathcal{D}_{poison}$. $\mathcal{D}_{\text{clean}}$ consists of genuine, correctly labeled sequences of skeleton data, while $\mathcal{D}_{poison}$ contains the sequences embedded with trigger actions by the attacker.

The attacker has two primary objectives: The first is to implant a backdoor mechanism in the classifier. This ensures that any skeleton sequence $\boldsymbol{S}$, once altered with the trigger action $\mathbf{V}$, is incorrectly classified into a specific target class $y_t \in \mathcal{Y}$. The second goal is to maintain the classifier's accuracy on unperturbed skeleton sequences, thereby preventing detection through diminished performance on a validation set. The overall optimizations formalized by the attacker's objective are given below:

$$\begin{aligned} &\max_{\boldsymbol{T}_{\mathbf{V}}(\cdot)} \mathbb{E}_{(\boldsymbol{S},y)\sim\mathcal{C}}\left[\mathbb{I}\left(f_{\boldsymbol{\theta}^*}(\boldsymbol{T}_{\mathbf{V}}(\boldsymbol{S}))=y_t\right)\right] + \mathbb{E}_{(\boldsymbol{S},y)\sim\mathcal{C}}\left[\mathbb{I}\left(f_{\boldsymbol{\theta}^*}(\boldsymbol{S})=y\right)\right], \\ \text{s.t. } &\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \sum_{(\boldsymbol{S}_i,y_i)\in\mathcal{D}_{clean}} \mathcal{L}(f_{\boldsymbol{\theta}}(\boldsymbol{S}_i),y_i) + \sum_{(\boldsymbol{T}_{\mathbf{V}}(\boldsymbol{S}_i),y_t)\in\mathcal{D}_{poison}} \mathcal{L}(f_{\boldsymbol{\theta}}(\boldsymbol{T}_{\mathbf{V}}(\boldsymbol{S}_i)),y_t), \end{aligned} \tag{1}$$

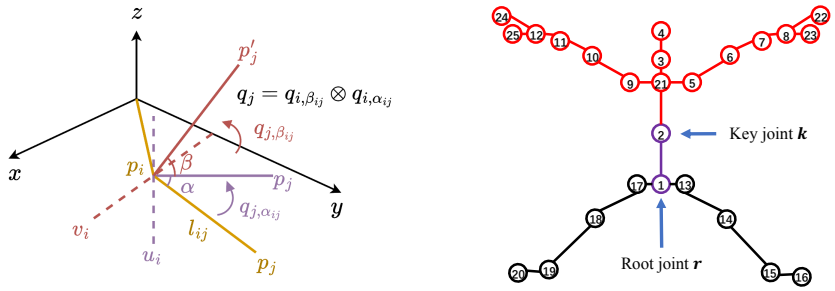
where $\mathcal{C}$ is the clean skeleton data, $\mathbb{I}(\cdot)$ is the indicator function, and $\boldsymbol{T}_{\mathbf{V}}(\cdot)$ is the trigger injection function. The loss function $\mathcal{L}$, the model architecture $f_{\boldsymbol{\theta}}$, and other hyper-parameters are chosen exclusively by the trainer.

To achieve these objectives, the attacker modifies the benign samples into poisoned ones and constructs a poisoned subset $\mathcal{D}_{poison} = \{(\boldsymbol{T}_{\mathbf{V}}(\boldsymbol{S}_i), y_t)\}_{i=1}^{N_p}$. To enhance the stealthiness of the attacks, it is common practice to adopt a low poisoning rate, represented by the ratio $\frac{|\mathcal{D}_{poison}|}{|\mathcal{D}_{train}|}$, often below a certain threshold like 10%. Furthermore, if the original label y_i in $\mathcal{D}_{poison}$ are altered to the target class y_t, the strategy is named poison-label attacks. Otherwise, if poisoned samples are selected from y_t, the strategy is considered clean-label attacks.

The assumptions for BA against SAR can be summarized as: 1) The attacker lacks access to the training process, including model architecture and loss function. 2) The attacker has access to part of the training data. 3) For real-world effectiveness, the poisoned data must represent physically plausible movements.

3.2 Trigger Implantation via Physical Movements

Since skeleton data typically comprises over 20 joints, directly estimating the change for all joints to simulate physical movements can be challenging. Therefore, it is more efficient to manually determine the changes for key joints and estimate the adjustments for the remaining joints based on their topological relationships. Considering the fixed arm lengths, we propose to estimate the joint



(a) (b)

Fig. 2: (a): Schematic diagram of rotation using quaternion. (b): Illustration of joints. Purple joints from root joint r to key joint k undergo inverse kinematics, while positions of all colored joints except the root joint r are transformed to the target position in forward kinematics. (b) illustrates that when using "bending sideways" as the trigger action, joint 1 serves as the root joint, and joint 2 serves as the key joint.

positions through the manipulation of joint angles within a humanoid robotic system. This system mirrors a kinematic chain with multiple degrees of freedom, akin to the human skeletal structure, which is comprised of interconnected links (bones) and joints in a sequential coordinate frame organization.

The simulation relies on a skeleton sequence $\boldsymbol{S} = \{S_\tau\}_{\tau=1}^{T}$, where each S_τ is a skeleton frame at time τ. To facialiate the injection of a trigger action $\mathbf{V}$ into $\boldsymbol{S}$, we consider the dynamics of the joint chain, and adopt quaternion representation for the rotation of all joints from each skeleton frame S_τ.

Quaternion representation for rotations. Each skeleton sequence frame comprises a set of joints, each defined by its relation to a parent joint and its position in space. As shown in Fig. 2 (a), the relationship includes the orthogonal rotational axes vectors $\boldsymbol{u}_i$ and $\boldsymbol{v}_i$ of the parent joint i, the bone length l_{ij} connecting the joint i to its child joint j, the rotation angles α_{ij} and β_{ij} corresponding to rotations around the $\boldsymbol{u}_i$ and $\boldsymbol{v}_i$ axes, respectively, and the position of the joint p_j. To represent rotations as the quaternion [1], joint j first rotates around $\boldsymbol{u}_i$ by α_{ij} angle, and then rotates around $\boldsymbol{v}_i$ by β_{ij} angle. Splitting the rotations into two steps allows the overall rotation process to be expressed as:

$$\boldsymbol{q}_{j,\alpha_{ij}} = \cos(\frac{\alpha_{ij}}{2}) + \sin(\frac{\alpha_{ij}}{2})\,[\boldsymbol{i},\boldsymbol{j},\boldsymbol{k}]\cdot\boldsymbol{u}_i,\ \ \boldsymbol{q}_{j,\beta_{ij}} = \cos(\frac{\beta_{ij}}{2}) + \sin(\frac{\beta_{ij}}{2})\,[\boldsymbol{i},\boldsymbol{j},\boldsymbol{k}]\cdot\boldsymbol{v}_i, \\ \boldsymbol{q}_j = \boldsymbol{q}_{i,\beta_{ij}} \otimes \boldsymbol{q}_{i,\alpha_{ij}}, \tag{2}$$

where $\boldsymbol{q}_{j,\alpha_{ij}}$, $\boldsymbol{q}_{j,\beta_{ij}}$ is the quaternion for the first and second rotation, respectively. $\boldsymbol{q}_j$ is the quaternion for the overall rotations, and $\otimes$ is quaternion multiplication. The relative position of joint j to its parent joint i is given by:

$$\boldsymbol{p}'_j - \boldsymbol{p}_i = \boldsymbol{M}(\boldsymbol{q}_j)\cdot(\boldsymbol{p}_j - \boldsymbol{p}_i), \tag{3}$$

[1] For the definition of quaternion, please refer to the supplementary material.

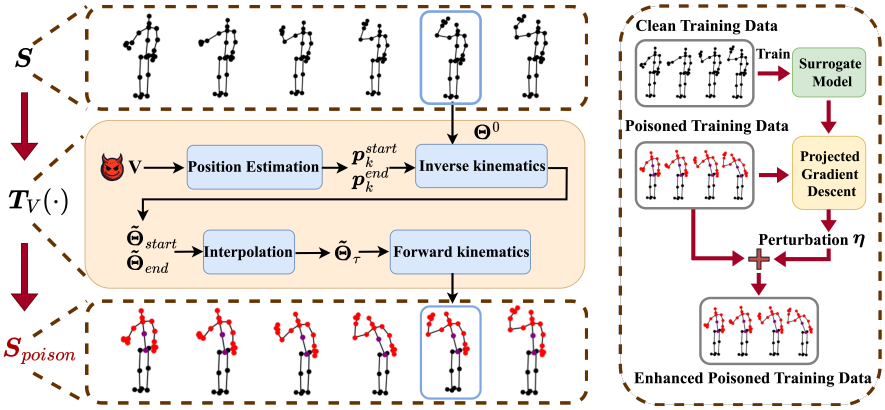


Fig. 3: (a): Trigger implantation diagram. (b): Trigger-enhancing strategy of C-PSBA.

where the matrix $\boldsymbol{M}$ is a function of $\boldsymbol{q}_j$, and is given by:

$$\boldsymbol{M}(\boldsymbol{q}_j) = \begin{bmatrix} 1-2(q_{yj}^2+q_{zj}^2) & 2(q_{xj}q_{yj}-q_{zj}q_{wj}) & 2(q_{xj}q_{zj}+q_{yj}q_{wj}) \\ 2(q_{xj}q_{yj}+q_{zj}q_{wj}) & 1-2(q_{xj}^2+q_{zj}^2) & 2(q_{yj}q_{zj}-q_{xj}q_{wj}) \\ 2(q_{xj}q_{zj}-q_{yj}q_{wj}) & 2(q_{yj}q_{zj}+q_{xj}q_{wj}) & 1-2(q_{xj}^2+q_{yj}^2) \end{bmatrix}, \tag{4}$$

Poisoned sequences generation via kinematic transformations. To generate the poisoned skeleton sequence $\boldsymbol{S}_{\text{poison}}$ from the original one $\boldsymbol{S}$, we need to calculate the kinematic transformation between these sequences, as shown in Fig. 3 (a). This transformation associated with any action can be characterized by the variations in joint orientations along the kinematic chain.

Position estimation. Given the pre-defined trigger action $\mathbf{V}$, we first need to define the root joint r that is unchanged, *e.g.,* the base of the spine joint for bending sideways. Then, we select the key joint k (*e.g.,* the middle of the spine joint for bending sideways), and estimate the corresponding spatial positions for both the start and end state, denoted as $\boldsymbol{p}_k^{\text{start}}$ and $\boldsymbol{p}_k^{\text{end}}$, respectively.

Inverse kinematics from joint r to joint k. Subsequently, we employ inverse kinematics to compute the target orientations for all joints on the chain from the root joint r to the key joint k, based on the target position of key joint k estimated in the prior step. Inverse kinematics is solved using a numerical algorithm such as the Jacobian transpose method, which iteratively adjusts the joint orientations to minimize the difference between the current position and the target position of joint k. The iterative optimization can be described by:

$$\boldsymbol{\Theta}^{iter} = \boldsymbol{\Theta}^{iter-1} + \lambda \cdot \mathbf{K}^T(\boldsymbol{\Theta}^{iter-1})(\boldsymbol{p}_k^{\text{target}} - \mathbf{F}(\boldsymbol{\Theta}^{iter-1})), \tag{5}$$

where $\boldsymbol{\Theta}^{iter} = [\alpha_{rm}, \beta_{rm}, \ldots, \alpha_{ek}, \beta_{ek}]^T$ is a vector comprising α_{ij} and β_{ij} of $\boldsymbol{q}_j$ for all joints on the chain from the root joint r to the key joint k at iteration $iter$, m and e are used to represent the child joints of r and the parent joints of k, λ is the learning rate, $\mathbf{F}$ is the forward kinematics function that outputs the

position of the key joint k as illustrated later, $\mathbf{K}(\mathbf{\Theta}^{iter-1})$ is the corresponding Jacobian matrix regarding $\mathbf{F}$ and $\mathbf{\Theta}^{iter-1}$, and $\boldsymbol{p}_k^{\text{target}}$ is the desired target position for the key joint k. $\mathbf{\Theta}^0$ is initialized by an all-zero vector. Therefore, for the spatial positions of both the start and end states, $\boldsymbol{p}_k^{\text{start}}$ and $\boldsymbol{p}_k^{\text{end}}$, we can utilize the aforementioned inverse kinematics to obtain the desired $\tilde{\mathbf{\Theta}}_{\text{start}}$ and $\tilde{\mathbf{\Theta}}_{\text{end}}$, respectively. Here, $\tilde{\mathbf{\Theta}}$ represents the target orientations of the final iteration.

Linear interpolation for target orientations. Let τ_s and τ_e denote the start and end indices of the action within the sequence, respectively. For any given frame τ, utilizing Eq. 5, we determine the start orientations $\tilde{\mathbf{\Theta}}_{\tau,\text{start}}$ and the end orientations $\tilde{\mathbf{\Theta}}_{\tau,\text{end}}$ relevant to frame τ. Subsequently, the desired orientation $\tilde{\mathbf{\Theta}}_\tau$ is calculated by interpolating between the initial and final orientations by:

$$\tilde{\mathbf{\Theta}}_\tau = \tilde{\mathbf{\Theta}}_{\tau,\text{start}} + \frac{\tau - \tau_s}{\tau_e - \tau_s} \times (\tilde{\mathbf{\Theta}}_{\tau,\text{end}} - \tilde{\mathbf{\Theta}}_{\tau,\text{start}}). \tag{6}$$

Forward kinematics from joint r to all affected joints. As depicted in Fig. 2 (b), once we acquire the desired quaternions $\tilde{\boldsymbol{q}}$ for the purple joints from r to k from $\tilde{\mathbf{\Theta}}_\tau$, we can employ forward kinematics to compute the revised positions of all impacted joints, namely, the colored joints excluding the root joint r. This process involves computing transformation matrices that define the relationship of each joint's updated position to the original one.

Since rotations accumulate from the parent joint to the child joint, the transformation matrix $\boldsymbol{R}_j$ for any joint j (the purple one) can be obtained by:

$$\boldsymbol{R}_j = \boldsymbol{R}_i \cdot \boldsymbol{M}(\tilde{\boldsymbol{q}}_j), \quad \text{with } \boldsymbol{R}_r = \boldsymbol{I}, \tag{7}$$

where the matrix $\boldsymbol{M}$ is defined in Eq. 4, and joint i is the parent joint of joint j. Note that the parent joint is updated before the child joint. Thus, the position of any joint j (the purple one) can be iteratively updated by:

$$\boldsymbol{p}'_j - \boldsymbol{p}'_i = \boldsymbol{R}_j \cdot (\boldsymbol{p}_j - \boldsymbol{p}_i), \quad \text{with } \boldsymbol{p}'_r = \boldsymbol{p}_r, \tag{8}$$

where $\boldsymbol{p}'_j$ is the updated position of joint j.

To simplify, we assume that the distant joints, *i.e.,* the red ones in Fig. 2 (b), have no relative rotation to the key joint k. For these red joint l, the transformation matrix $\boldsymbol{R}_j$ is the same as $\boldsymbol{R}_k$, thus the updated position is:

$$\boldsymbol{p}'_l - \boldsymbol{p}'_k = \boldsymbol{R}_k \cdot (\boldsymbol{p}_l - \boldsymbol{p}_k). \tag{9}$$

3.3 Backdoor Attack Framework

Physical Trigger Action Design. For the trigger actions, we apply the following kinematic transformations and show the selected joints in Table 1:

- **Nodding:** This action is characterized by a downward tilt followed by a return to the initial head position. We model this as a two-phase uniform angular sampling for the head joint, first for the nod and then for the return to the upright position.

Table 1: Illustration of the selected joints to affect for each trigger action.

Trigger Action	Root joint r	Key joint k	Joints for inverse kinematics	Remaining affected joints
Nodding	3	4	3, 4	N/A
Bending sideways	1	2	1, 2	3 - 12 & 21 - 25
Crossing hands at the front	5 9	8 12	5, 6, 7, 8 9, 10, 11, 12	22, 23 24, 25

- **Bending Sideways:** The lateral flexion and subsequent return to the upright position are represented by sampling from a uniform distribution for the spine and hip joint angles, modeling the bend and the return sequence.
- **Crossing Hands at the Front:** This action involves moving the arms from a neutral position to a configuration where hands are crossed in front. The wrist, elbow, and shoulder joint angles are uniformly sampled to create a smooth transition to this end pose and back. Note that this action involves two separate kinetic transformations (*i.e.,* left and right arms).

To ensure that the proposed attacks are robust to human variability, the additional design considerations for poisoned samples are summarized as below:

- **Temporal Sampling of Action Duration:** Let the duration range for a trigger action T be a random variable from a uniform distribution $\mathcal{U}(t_{\min}, t_{\max})$ and sample from this distribution to determine the trigger action's timing.
- **Sampling of Hyperparameter for Position Estimation:** For each trigger action, we estimate the $\boldsymbol{p}_k^{\text{start}}$ and $\boldsymbol{p}_k^{\text{end}}$ of key joint k using a action-related hyperparameter Φ sampled from a uniform distribution $\mathcal{U}(\phi_{\min}, \phi_{\max})$. For example, Φ can be the angle to bend for the "bending sideways", and the position of the crossed hands for the "crossing hands at the front".

Poison-label Backdoor Attacks. The poison-label physical skeleton backdoor attack (**P-PSBA**) is introduced to concretely validate the potency of our designed backdoor triggers. A portion of randomly selected data from the training set undergoes subtle modifications, incorporating trigger action $\mathbf{V}$ into selected skeleton sequences, and the corresponding labels are modified to the target class y_t. When the SAR system is trained with this poisoned dataset, it becomes conditioned to identify these trigger actions as the designated target class.
Clean-label Backdoor Attacks. The clean-label physical skeleton backdoor attack (**C-PSBA**) is designed to bypass label inspection by incorporating triggers without changing the labels. Clean-label BA are commonly perceived as the most stealthy strategies, wherein adversaries are restricted to poisoning samples from the designated target class without altering their labels.

We elucidate that the inherent challenge of clean-label attacks predominantly stems from the adversarial impact of salient features induced by the trigger within poisoned samples. Such salient features are prone to be easily learned, thereby impeding the learning of trigger patterns. To enhance the efficacy of the trigger and reduce the prominence of original skeleton features, we employ adversarial perturbations, as shown in Fig. 3 (b). After injecting the trigger

action into selected skeleton sequences from the target class, we introduce subtle adversarial noise into the skeleton data. These perturbations can diminish the original content's impact and encourage the model to focus more on the trigger. Notably, such perturbations are only added to the poisoned training data, *they are not necessary at run-time.*

Formally, for a given surrogate model f_s that uses a different model architecture and trained on a clean dataset and an input skeleton sequence $\boldsymbol{S}$ from the target class y_t, we generate the untargeted adversarial perturbations $\boldsymbol{\eta}$ by maximizing the cross-entropy loss $\mathcal{L}$ as follows:

$$\max_{\|\eta\|_2 \leq \epsilon} \mathcal{L}(f_s(\boldsymbol{T}_{\mathbf{V}}(\boldsymbol{S}) + \boldsymbol{\eta}), y_t), \tag{10}$$

where ϵ is the maximum for $\boldsymbol{\eta}$. Empirically, we find that such perturbations can transform the original skeleton into hard samples, making it more hard for the model to learn the inherent features associated with the target class. Thus, these perturbations make the learning of the model focus more on the trigger action.

3.4 Stealthiness

Stealthiness of poisoned samples within the dataset is crucial to the success of BA, as it aids in eluding detection. To measure the stealthiness from the view of distributions, we employ two statistical metrics: KL Divergence (KLD) and Earth Mover's Distance (EMD). [2] We calculate the angles between the adjacent bones and statistically distribute on the chain from the joint r to the joint k. By analyzing the distributions of modified dataset and original one, we aim to demonstrate that poisoned samples maintain a high degree of stealthiness.

4 Experiments

4.1 Experimental Setup

Models. For deep SAR models, we consider a transformer-based and two GCN-based models: Hyperformer [59], CTR-GCN [5], and INFO-GCN [6].
Datasets. We select three datasets well-known in SAR. **NTU RGB+D [38]** is a large-scale SAR dataset containing 56,880 skeleton sequences. Actions are performed by 40 distinct volunteers and are categorized into 60 classes. The NTU RGB+D dataset provides two evaluation protocols, namely cross-view (X-view) and cross-subject (X-sub). **NTU RGB+D 120 [28]** is an extension to NTU RGB+D, and is the largest SAR dataset, with 114,480 samples over 120 classes. **PKU-MMD [26]** contains almost 20, 000 action instances and 5.4 million frames in 51 action categories, and also utilizes X-view and X-sub evaluation.
Evaluation Metrics. To evaluate the effectiveness of our attacks, we utilize two key metrics: Model Accuracy (ACC) and Attack Success Rate (ASR). [2]
Attack setting. For all experiments, we choose the default class "0" to be the target class (*e.g.,* "drink water" in NTU RGB+D and NTU RGB+D 120, and "bow" in PKU-MMD).

[2] Detailed definition can be found in the supplementary material.

Table 2: ASR (%) ↑ and ACC (%) ↑ of P-PSBA with different poisoning rates.

Dataset →		NTU RGB+D						NTU RGB+D 120						PKU-MMD					
Trigger	Ratio (%)	Hyperformer		CTR-GCN		INFO-GCN		Hyperformer		CTR-GCN		INFO-GCN		Hyperformer		CTR-GCN		INFO-GCN	
		ASR	ACC	ASR	ACC	ASR	ACC	ASR	ACC	ASR	ACC	ASR	ACC	ASR	ACC	ASR	ACC	ASR	ACC
Clean	-	-	90.67	-	90.08	-	90.78	-	86.62	-	85.08	-	87.36	-	94.85	-	95.23	-	93.58
Nodding	0.1	14.72	90.35	20.34	89.69	1.71	90.55	18.90	86.57	8.33	84.99	13.65	87.42	2.51	94.49	1.33	95.03	3.67	93.27
	0.2	23.98	90.38	25.48	89.94	27.39	90.55	24.61	86.45	17.57	84.96	31.22	87.12	6.94	94.57	1.83	95.01	2.33	93.35
	0.5	54.75	90.46	50.55	89.96	69.49	90.49	58.95	86.47	60.74	84.98	64.36	87.03	13.35	94.52	2.47	95.02	4.17	92.77
	1	83.30	90.41	80.28	90.33	85.50	90.86	82.84	86.39	90.26	84.96	80.12	87.13	75.98	94.61	78.52	94.98	64.09	92.23
	2	94.26	90.48	97.58	89.94	92.75	90.64	89.39	86.37	94.18	85.01	93.17	87.27	99.50	94.39	98.67	94.96	99.33	93.12
	5	98.55	90.53	99.30	90.20	95.07	90.71	93.57	86.43	94.38	85.06	94.78	87.25	99.83	94.48	99.83	95.01	99.50	93.85
Bending sideways	0.1	19.24	90.32	7.04	90.10	7.21	90.85	24.78	86.38	21.08	85.04	25.48	87.24	95.74	94.44	96.50	94.75	97.33	93.42
	0.2	54.33	90.35	45.68	90.04	50.95	90.82	49.50	86.49	42.56	85.02	52.26	87.57	97.80	94.78	97.33	95.02	98.50	93.92
	0.5	85.42	90.41	82.01	89.96	87.99	90.69	64.85	86.51	61.77	84.96	67.50	87.36	99.83	94.69	99.67	95.04	99.83	93.04
	1	88.37	90.37	85.67	89.94	92.49	90.83	87.93	86.37	91.09	84.98	88.16	87.16	99.83	94.52	99.83	94.86	99.83	92.90
	2	92.88	90.54	89.91	89.88	99.20	90.88	93.14	86.43	93.78	84.99	94.88	87.31	99.83	94.63	99.83	94.90	99.83	93.38
	5	96.41	90.40	93.63	89.92	99.60	90.44	95.62	86.45	94.50	84.93	96.19	87.54	99.83	94.45	99.83	94.84	99.83	92.88
Crossing hands at the front	0.1	29.72	90.38	22.56	89.98	24.87	90.80	5.84	86.41	1.26	85.03	0.11	87.37	26.71	94.77	24.62	94.50	20.88	93.58
	0.2	57.63	90.47	39.48	89.93	44.62	90.68	11.07	86.43	2.07	84.98	0.45	87.22	28.49	94.68	22.88	95.04	28.12	93.38
	0.5	82.06	90.42	76.54	89.94	84.20	90.61	88.65	86.35	86.45	84.96	94.75	87.29	68.03	94.52	81.12	94.92	57.88	93.15
	1	90.85	90.39	86.10	89.78	93.68	90.73	94.39	86.29	94.87	85.01	96.99	87.18	81.19	94.56	85.50	94.88	66.00	93.73
	2	95.87	90.45	90.28	89.91	98.67	90.54	97.92	86.41	98.69	84.99	99.89	87.13	93.71	94.49	94.38	94.54	93.38	93.44
	5	98.94	90.51	94.41	89.98	99.83	90.78	99.89	86.37	99.89	85.03	99.89	87.27	94.39	94.51	94.88	95.08	94.38	93.19

4.2 Experimental Results

Poison-Label Attacks (P-PSBA). Our experimental analysis unveils the intricate dynamics of BA under different poisoning rates and across multiple model architectures. We conducted evaluations on NTU RGB+D, NTU RGB+D 120, and PKU-MMD datasets, employing poisoning rates of 0.1%, 0.2%, 0.5%, 1%, 2%, and 5%. The experimental results are shown in Table 2.

Influence of Poisoning Rates. Our investigation reveals that higher poisoning rates generally elevate the ASR, yet without detrimentally impacting the ACC. This trend was consistently observed across Hyperformer [59], CTR-GCN [5], and INFO-GCN [6]. For instance, within the NTU RGB+D dataset and using the Hyperformer model under a poison-label scenario, the ASR for the "nodding" trigger action notably escalated from 14.72% to an impressive 98.55% as the poisoning rate increased from 0.1% to 5%. Similarly, other actions such as "bending sideways" and "crossing hands at the front" followed suit, demonstrating the effectiveness of the attacks.

Clean-Label Attacks (C-PSBA). We do a thorough evaluation under the clean-label setting, applying varying poisoning rates at 30%, 50%, 70%, and 90% specifically to the data of the target class. Experimental result shows that higher poisoning rates can boost the ASR of C-PSBA without negatively affecting the ACC. In contrast to attacks in the poison-label scenario, achieving a high ASR in the clean-label scenario proved challenging due to two primary reasons. Firstly, the clean-label approach limits attackers to only poisoning data from the target class, inherently restricting the amount of data that can be manipulated. Secondly, in the clean-label scenario, the effectiveness of the trigger pattern can be diminished by the influence of the original features of the target class, which can obstruct the model's learning of the trigger pattern, as discussed in Section 3.3. As shown in Table 3, although it is possible to attain an ASR between 70% and 80% by increasing the poisoning ratio, it proves challenging to achieve an ASR higher than 80%. Thus, attackers must consider the trade-off between stealthiness and effectiveness before launching BA. For those aiming for a high

Table 3: ASR (%) ↑ and ACC (%) ↑ of C-PSBA with different poisoning rates.

Dataset →		NTU RGB+D						NTU RGB+D 120						PKU-MMD					
Trigger	Ratio (%)	Hyperformer		CTR-GCN		INFO-GCN		Hyperformer		CTR-GCN		INFO-GCN		Hyperformer		CTR-GCN		INFO-GCN	
		ASR	ACC	ASR	ACC	ASR	ACC	ASR	ACC	ASR	ACC	ASR	ACC	ASR	ACC	ASR	ACC	ASR	ACC
Clean	-	-	90.67	-	90.08	-	90.78	-	86.62	-	85.08	-	87.36	-	94.85	-	95.23	-	93.58
Nodding	30	52.20	90.34	43.31	89.90	54.28	90.53	28.42	86.34	24.30	84.96	26.74	87.26	60.20	94.63	58.06	94.98	60.98	93.56
	50	56.98	90.42	46.54	89.93	61.27	90.56	31.67	86.40	27.68	85.02	31.46	87.39	66.47	94.72	64.38	95.03	67.74	93.48
	70	64.27	90.38	52.47	89.98	68.94	90.61	36.23	86.52	32.41	85.03	38.65	87.40	70.58	94.65	67.49	95.16	70.95	93.36
	90	65.70	90.48	53.61	90.02	69.32	90.58	38.79	86.49	33.55	85.02	39.98	87.38	71.23	94.67	68.89	95.18	71.49	93.45
Bending sideways	30	64.75	90.36	60.20	89.67	62.48	90.70	51.48	86.37	49.57	85.01	52.76	87.36	72.08	94.68	69.94	94.84	71.81	93.47
	50	69.02	90.47	65.94	89.84	66.57	90.73	59.65	86.48	53.82	84.98	62.49	87.32	74.63	94.72	72.58	94.96	75.42	93.50
	70	75.35	90.45	71.85	90.02	74.83	90.81	65.41	86.51	60.94	85.04	68.20	87.41	77.92	94.74	74.80	95.02	78.97	93.48
	90	77.14	90.39	73.04	89.79s	74.95	90.79	68.93	86.59	63.02	84.98	71.44	87.42	78.36	94.76	75.50	95.14	79.31	93.49
Crossing hands at the front	30	64.34	90.46	64.21	89.84	62.48	90.63	65.72	86.44	62.30	84.86	68.65	86.92	72.83	94.69	71.18	95.08	67.50	93.37
	50	70.13	90.57	67.94	89.90	68.69	90.72	69.05	86.49	68.23	85.03	71.40	86.98	76.30	94.73	73.08	95.25	73.29	93.41
	70	77.75	90.54	73.30	89.93	73.60	90.68	72.83	86.53	72.76	85.01	74.53	86.94	78.51	94.79	76.43	95.21	75.69	93.45
	90	79.06	90.52	75.92	90.04	74.51	90.84	74.91	86.52	75.30	85.06	72.61	87.01	79.88	94.78	76.94	95.21	74.84	93.42

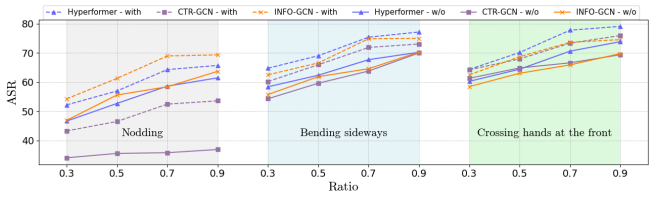


Fig. 4: Ablation study on the with/without trigger-enhancing strategy for C-PSBA.

ASR, poison-label attacks may be preferable. However, if concerns about label detection exist, clean-label attacks become a viable alternative.

Effectiveness of the trigger-enhancing strategy. To assess the efficacy of the trigger-enhancing strategy, we conducted an ablation study on the NTU RGB+D dataset. As shown in Fig. 4, our strategy consistently improved the ASR across different poisoning ratios, demonstrating its effectiveness.

5 Discussion

5.1 Skeleton sequences with trigger from real-world data

To ascertain the real-world applicability of our proposed BA, we undertook the creation of a physical backdoor attack dataset. [3] We evaluate the practicality of P-PSBA with poisoning rates set at 1% and 2%. As shown in Table 4, most models trained on the poisoned datasets with a 2% poisoning rate reliably recognized the embedded triggers with over 85% ASR, demonstrating the efficacy and reliability of our attacks in the real world.

5.2 Resistance to Defenses

To mitigate the effects of BA in backdoored models, methods range from trigger-synthesis based methods [1, 36, 41], which synthesize potential triggers and suppress their effects, to solutions like finetuning [2], pruning [30, 58], and input

[3] Further details of the dataset are available in the supplementary.

Table 4: Physical ASR (%) on models trained with 1% and 2% poisoning ratio.

Dataset →		NTU RGB+D			NTU RGB+D 120			PKU-MMD		
Trigger	Ratio (%)	Hyperformer	CTR-GCN	INFO-GCN	Hyperformer	CTR-GCN	INFO-GCN	Hyperformer	CTR-GCN	INFO-GCN
Nodding	1	41.87	35.57	39.73	30.68	25.47	29.08	50.14	47.76	49.11
	2	71.40	66.83	70.62	55.96	49.57	60.71	80.17	70.98	77.79
Bending sideways	1	81.62	69.83	77.81	69.94	67.66	66.34	83.01	80.19	81.24
	2	91.72	88.73	90.02	89.39	86.03	88.57	95.05	90.77	92.13
Crossing hands	1	71.82	66.91	70.41	61.60	56.55	57.85	75.53	68.35	72.65
	2	90.64	81.93	87.72	84.77	82.79	83.59	92.75	89.34	90.48

detection [12]. However, trigger-synthesis based methods are not compatible with our attacks. Therefore, we select three defenses: CLP [58], D-BR [2], and STRIP [12] with their default settings. Experiments are conducted for P-PSBA using NTU RGB+D as the dataset, INFO-GCN as the SAR model, and "bending" as the trigger action. For CLP and D-BR, we explore poisoning rates of 0.5%, 1%, and 2% , while STRIP is evaluated at a 2% poisoning rate. [4]

Resistance to Prunning based Defenses. CLP [58] argues that channels related to BA exhibit higher Lipschitz constants compared to normal channels. By assessing the Lipschitz constant across channels, CLP aims to identify and prune these sensitive channels. According to the results presented in Table 5, CLP does not effectively counter our P-PSBA approach.

Resistance to Fine-tuning based Defenses. D-BR [2] consists of two modules: the Sample-Distinguishment (SD) module and the Backdoor Removal (BR) module. The SD module splits the training set into clean, poisoned, and uncertain samples. The BR module then alternatively unlearns the distinguished poisoned samples and learns the distinguished clean samples. We finetune 15 epochs on the NTU RGB+D dataset, and the experimental results are shown in Table 5. It can be observed that D-BR is ineffective for our P-PSBA.

Resistance to Sample Filtering based Defenses. STRIP [12] proposes to deliberately inject strong perturbations into each input to effectively identify backdoor inputs. By analyzing the entropy in the prediction probabilities, it distinguishes between backdoor inputs, characterized by consistently low entropy, and clean inputs, which exhibit high entropy. As shown in Fig. 5, the entropy distributions for both clean and poisoned samples in our P-PSBA are similar, suggesting that they can bypass the STRIP defense mechanism.

5.3 Stealthiness vs. Attack Performance

In this section, the stealthiness of P-PSBA is assessed through the analysis of the KLD and EMD metrics. Given the consistency in data processing and integration of backdoor patterns across three datasets, we focus our evaluation exclusively on the NTU RGB+D 120 dataset. As shown in Table 6, despite an increase in poison ratio, both KLD and EMD metrics remained within a relatively low range. Further, we analyze the relationship between stealthiness and attack performance. As shown in Fig. 6, the ASR is highly correlated with EMD metrics. For different triggers, when EMD reaches 20, the ASR can exceed 90%.

[4] Additional experiments on defenses are available in the supplementary material.

Table 5: Resistance of P-PSBA to CLP and D-BR. We set "bending" as the trigger action.

Model ↓ \ Defense → Ratio (%) ↓		None ASR	CLP ASR	D-BR ASR
Hyperformer	0.5	85.42	85.23	84.91
	1	88.37	88.19	88.67
	2	92.88	92.81	92.50
CTR-GCN	0.5	82.01	81.94	82.99
	1	85.67	85.41	85.03
	2	89.91	89.88	90.16
INFO-GCN	0.5	87.99	87.83	85.03
	1	92.49	92.46	91.87
	2	99.20	99.05	99.15

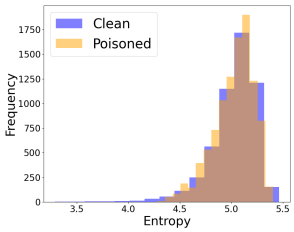


Fig. 5: Resistance of P-PSBA to STRIP. Poisoning ratio is set to 2%.

Table 6: Stealthiness of P-PSBA in terms of KLD (10^{-7}) ↓ and EMD (10^{-3}) ↓.

Ratio (%)	Nodding		Bending		Crossing hands	
	KLD	EMD	KLD	EMD	KLD	EMD
0.1	2.03	3	6.84	5	4.17	4
0.2	4.49	5	16.7	8	7.68	6
0.5	29.9	11	59.4	16	29.2	11
1	108	21	190	31	75.2	23
2	389	38	683	66	256	48
5	2160	87	3930	139	1180	95

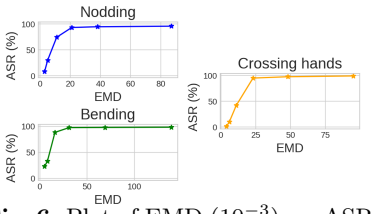


Fig. 6: Plot of EMD (10^{-3}) vs. ASR.

6 Ethics Statement

This paper proposes a novel backdoor attack against SAR systems. While there is a risk that PSBA could be misused to compromise surveillance systems or cause misclassifications, our objective is to reveal these potential vulnerabilities rather than facilitate attacks. By exposing these vulnerabilities, we emphasize the urgent need for improved defense mechanisms. Our intention is to encourage the development of robust defenses against such attacks, ensuring safer implementations of SAR systems. This work underscores the necessity for the community to recognize backdoor attacks as significant real-world threats and to prioritize their mitigation.

7 Conclusion

In this paper, we introduce a novel backdoor attack specifically tailored for SAR systems, utilizing joint angle manipulations to embed realistic trigger actions. Comprehensive experiments on both clean label and poison label settings indicate that the proposed attack achieves significant attack success rates and notable resistance against several defenses while maintaining high model accuracy. Moreover, we design a trigger-enhancing strategy that significantly improves the ASR in clean-label scenarios. Additionally, our validation using real-world datasets highlights the immediate necessity for advanced defensive strategies in crucial applications. This work underscores the vulnerabilities in SAR systems and sets the stage for future research into robust defense mechanisms.

Acknowledgments This research is supported by the National Research Foundation, Singapore and Infocomm Media Development Authority under its Trust Tech Funding Initiative. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not reflect the views of National Research Foundation, Singapore and Infocomm Media Development Authority. This research is also supported in part by the NTU-PKU Joint Research Institute and the DSO National Laboratories, Singapore, under the project agreement No. DSOCL22332.

References

1. Chen, H., Fu, C., Zhao, J., Koushanfar, F.: Deepinspect: a black-box trojan detection and mitigation framework for deep neural networks. In: Proceedings of the 28th International Joint Conference on Artificial Intelligence. pp. 4658–4664 (2019)
2. Chen, W., Wu, B., Wang, H.: Effective backdoor defense by exploiting sensitivity of poisoned samples. Advances in Neural Information Processing Systems **35**, 9727–9737 (2022)
3. Chen, X., Salem, A., Chen, D., Backes, M., Ma, S., Shen, Q., Wu, Z., Zhang, Y.: Badnl: Backdoor attacks against nlp models with semantic-preserving improvements. In: Annual computer security applications conference. pp. 554–569 (2021)
4. Chen, X., Liu, C., Li, B., Lu, K., Song, D.: Targeted backdoor attacks on deep learning systems using data poisoning. arXiv preprint arXiv:1712.05526 (2017)
5. Chen, Y., Zhang, Z., Yuan, C., Li, B., Deng, Y., Hu, W.: Channel-wise topology refinement graph convolution for skeleton-based action recognition. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 13359–13368 (2021)
6. Chi, H.g., Ha, M.H., Chi, S., Lee, S.W., Huang, Q., Ramani, K.: Infogcn: Representation learning for human skeleton-based action recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 20186–20196 (2022)
7. Diao, Y., Shao, T., Yang, Y.L., Zhou, K., Wang, H.: Basar: Black-box attack on skeletal action recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7597–7607 (2021)
8. Doan, K., Lao, Y., Zhao, W., Li, P.: Lira: Learnable, imperceptible and robust backdoor attacks. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 11966–11976 (2021)
9. Du, Y., Fu, Y., Wang, L.: Skeleton based action recognition with convolutional neural network. In: 2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR). pp. 579–583. IEEE (2015)
10. Du, Y., Wang, W., Wang, L.: Hierarchical recurrent neural network for skeleton based action recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1110–1118 (2015)
11. Dumford, J., Scheirer, W.: Backdooring convolutional neural networks via targeted weight perturbations. In: 2020 IEEE International Joint Conference on Biometrics (IJCB). pp. 1–9 (2020)
12. Gao, Y., Xu, C., Wang, D., Chen, S., Ranasinghe, D.C., Nepal, S.: Strip: A defence against trojan attacks on deep neural networks. In: Proceedings of the 35th Annual Computer Security Applications Conference. pp. 113–125 (2019)

13. Garcia-Cobo, G., SanMiguel, J.C.: Human skeletons and change detection for efficient violence detection in surveillance videos. Computer Vision and Image Understanding **233**, 103739 (2023)
14. Gu, T., Dolan-Gavitt, B., Garg, S.: Badnets: Identifying vulnerabilities in the machine learning model supply chain. arXiv preprint arXiv:1708.06733 (2017)
15. Guo, C., Wu, R., Weinberger, K.Q.: Trojannet: Embedding hidden trojan horse models in neural networks. arXiv preprint arXiv:2002.10078 (2020)
16. Hammoud, H.A.A.K., Ghanem, B.: Check your other door! creating backdoor attacks in the frequency domain. arXiv preprint arXiv:2109.05507 (2021)
17. Hbali, Y., Hbali, S., Ballihi, L., Sadgal, M.: Skeleton-based human activity recognition for elderly monitoring systems. IET Computer Vision **12**(1), 16–26 (2018)
18. Jafri, R., Louzada Campos, R., Arabnia, H.R.: A skeleton-based deep learning approach for recognizing violent actions in surveillance scenarios. In: International Conference on Human-Computer Interaction. pp. 624–631. Springer (2022)
19. Ke, Q., Bennamoun, M., An, S., Sohel, F., Boussaid, F.: A new representation of skeleton sequences for 3d action recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3288–3297 (2017)
20. Lee, J., Lee, M., Lee, D., Lee, S.: Hierarchically decomposed graph convolutional networks for skeleton-based action recognition. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 10444–10453 (2023)
21. Li, C., Zhong, Q., Xie, D., Pu, S.: Co-occurrence feature learning from skeleton data for action recognition and detection with hierarchical aggregation. arXiv preprint arXiv:1804.06055 (2018)
22. Li, S., Xue, M., Zhao, B.Z.H., Zhu, H., Zhang, X.: Invisible backdoor attacks on deep neural networks via steganography and regularization. IEEE Transactions on Dependable and Secure Computing **18**(5), 2088–2105 (2020)
23. Li, X., Chen, Z., Zhao, Y., Tong, Z., Zhao, Y., Lim, A., Zhou, J.T.: Pointba: Towards backdoor attacks in 3d point cloud. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 16492–16501 (2021)
24. Li, Y., Li, Y., Lv, Y., Jiang, Y., Xia, S.T.: Hidden backdoor attack against semantic segmentation models. arXiv preprint arXiv:2103.04038 (2021)
25. Li, Y., Li, Y., Wu, B., Li, L., He, R., Lyu, S.: Invisible backdoor attack with sample-specific triggers. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 16463–16472 (2021)
26. Liu, C., Hu, Y., Li, Y., Song, S., Liu, J.: Pku-mmd: A large scale benchmark for continuous multi-modal human action understanding. arXiv preprint arXiv:1703.07475 (2017)
27. Liu, J., Akhtar, N., Mian, A.: Adversarial attack on skeleton-based human action recognition. IEEE Transactions on Neural Networks and Learning Systems **33**(4), 1609–1622 (2020)
28. Liu, J., Shahroudy, A., Perez, M., Wang, G., Duan, L.Y., Kot, A.C.: Ntu rgb+ d 120: A large-scale benchmark for 3d human activity understanding. IEEE transactions on pattern analysis and machine intelligence **42**(10), 2684–2701 (2019)
29. Liu, J., Shahroudy, A., Xu, D., Kot, A.C., Wang, G.: Skeleton-based action recognition using spatio-temporal lstm network with trust gates. IEEE transactions on pattern analysis and machine intelligence **40**(12), 3007–3021 (2017)
30. Liu, K., Dolan-Gavitt, B., Garg, S.: Fine-pruning: Defending against backdooring attacks on deep neural networks. In: International symposium on research in attacks, intrusions, and defenses. pp. 273–294. Springer (2018)

31. Liu, Y., Ma, X., Bailey, J., Lu, F.: Reflection backdoor: A natural backdoor attack on deep neural networks. In: European Conference on Computer Vision. pp. 182–199 (2020)
32. Lu, Z., Wang, H., Chang, Z., Yang, G., Shum, H.P.: Hard no-box adversarial attack on skeleton-based human action recognition with skeleton-motion-informed gradient. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 4597–4606 (2023)
33. Morais, R., Le, V., Tran, T., Saha, B., Mansour, M., Venkatesh, S.: Learning regularity in skeleton trajectories for anomaly detection in videos. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 11996–12004 (2019)
34. Nguyen, T.A., Tran, A.: Input-aware dynamic backdoor attack. Advances in Neural Information Processing Systems **33**, 3454–3464 (2020)
35. Nguyen, T.A., Tran, A.T.: Wanet-imperceptible warping-based backdoor attack. In: International Conference on Learning Representations (2020)
36. Qiao, X., Yang, Y., Li, H.: Defending neural backdoors via generative distribution modeling. Advances in neural information processing systems **32** (2019)
37. Rakin, A.S., He, Z., Fan, D.: Tbt: Targeted neural network attack with bit trojan. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13198–13207 (2020)
38. Shahroudy, A., Liu, J., Ng, T.T., Wang, G.: Ntu rgb+ d: A large scale dataset for 3d human activity analysis. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1010–1019 (2016)
39. Tang, R., Du, M., Liu, N., Yang, F., Hu, X.: An embarrassingly simple approach for trojan attack in deep neural networks. In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. pp. 218–228 (2020)
40. Tramèr, F., Papernot, N., Goodfellow, I., Boneh, D., McDaniel, P.: The space of transferable adversarial examples. arXiv preprint arXiv:1704.03453 (2017)
41. Wang, B., Yao, Y., Shan, S., Li, H., Viswanath, B., Zheng, H., Zhao, B.Y.: Neural cleanse: Identifying and mitigating backdoor attacks in neural networks. In: 2019 IEEE Symposium on Security and Privacy (SP). pp. 707–723. IEEE (2019)
42. Wang, C., Yu, Y., Guo, L. and Wen, B., 2024, April. Benchmarking adversarial robustness of image shadow removal with shadow-adaptive attacks. In ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 13126-13130). IEEE.
43. Wang, H., He, F., Peng, Z., Shao, T., Yang, Y.L., Zhou, K., Hogg, D.: Understanding the robustness of skeleton-based action recognition under adversarial attack. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 14656–14665 (2021)
44. Wang, H., He, F., Peng, Z., Yang, Y., Shao, T., Zhou, K., Hogg, D.: Smart: Skeletal motion action recognition attack. arXiv preprint arXiv:1911.07107 (2019)
45. Wang, T., Yao, Y., Xu, F., An, S., Wang, T.: Backdoor attack through frequency domain. arXiv preprint arXiv:2111.10991 (2021)
46. Xiang, Z., Miller, D.J., Chen, S., Li, X., Kesidis, G.: A backdoor attack against 3d point cloud classifiers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 7597–7607 (2021)
47. Xia, S., Yu, Y., Jiang, X. and Ding, H., Mitigating the Curse of Dimensionality for Certified Robustness via Dual Randomized Smoothing. In: International Conference on Learning Representations. (2024)

48. Yan, S., Xiong, Y., Lin, D.: Spatial temporal graph convolutional networks for skeleton-based action recognition. In: Proceedings of the AAAI conference on artificial intelligence. vol. 32 (2018)
49. Yang, L., Huang, J., Feng, T., Hong-An, W., Guo-Zhong, D.: Gesture interaction in virtual reality. Virtual Reality & Intelligent Hardware **1**(1), 84–112 (2019)
50. Yang, S., Liu, J., Lu, S., Er, M.H., Kot, A.C.: Skeleton cloud colorization for unsupervised 3d action representation learning. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 13423–13433 (2021)
51. Yang, S., Liu, J., Lu, S., Hwa, E.M., Hu, Y., Kot, A.C.: Self-supervised 3d action representation learning with skeleton cloud colorization. IEEE Transactions on Pattern Analysis and Machine Intelligence (2023)
52. Yang, S., Liu, J., Lu, S., Hwa, E.M., Kot, A.C.: One-shot action recognition via multi-scale spatial-temporal skeleton matching. IEEE Transactions on Pattern Analysis and Machine Intelligence (2024)
53. Yin, J., Han, J., Xie, R., Wang, C., Duan, X., Rong, Y., Zeng, X., Tao, J.: Mc-lstm: Real-time 3d human action detection system for intelligent healthcare applications. IEEE Transactions on Biomedical Circuits and Systems **15**(2), 259–269 (2021)
54. Yu, Y., Wang, Y., Xia, S., Yang, W., Lu, S., Tan, Y.P. and Kot, A., Purify Unlearnable Examples via Rate-Constrained Variational Autoencoders. In: International Conference on Machine Learning. (2024)
55. Yu, Y., Wang, Y., Yang, W., Lu, S., Tan, Y.P., Kot, A.C.: Backdoor attacks against deep image compression via adaptive frequency trigger. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 12250–12259 (2023)
56. Yu, Y., Yang, W., Tan, Y.P. and Kot, A.C., 2022. Towards robust rain removal against adversarial attacks: A comprehensive benchmark analysis and beyond. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 6013-6022 (2022).
57. Zhang, P., Lan, C., Xing, J., Zeng, W., Xue, J., Zheng, N.: View adaptive recurrent neural networks for high performance human action recognition from skeleton data. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 2117–2126 (2017)
58. Zheng, R., Tang, R., Li, J., Liu, L.: Data-free backdoor removal based on channel lipschitzness. In: European Conference on Computer Vision. pp. 175–191. Springer (2022)
59. Zhou, Y., Cheng, Z.Q., Li, C., Geng, Y., Xie, X., Keuper, M.: Hypergraph transformer for skeleton-based action recognition. arXiv preprint arXiv:2211.09590 (2022)