Adversarial T-shirt! Evading Person Detectors in A Physical World

Kaidi Xu[1] Gaoyuan Zhang[2] Sijia Liu[2] Quanfu Fan[2] Mengshu Sun[1]
Hongge Chen[3] Pin-Yu Chen[2] Yanzhi Wang[1] Xue Lin[1]

[1]Northeastern University, USA
[2]MIT-IBM Watson AI Lab, IBM Research, USA
[3]Massachusetts Institute of Technology, USA

Abstract. It is known that deep neural networks (DNNs) are vulnerable to adversarial attacks. The so-called *physical adversarial examples* deceive DNN-based decision makers by attaching adversarial patches to real objects. However, most of the existing works on physical adversarial attacks focus on static objects such as glass frames, stop signs and images attached to cardboard. In this work, we propose *Adversarial T-shirts*, a robust physical adversarial example for evading person detectors even if it could undergo non-rigid deformation due to a moving person's pose changes. To the best of our knowledge, this is the first work that models the effect of deformation for designing physical adversarial examples with respect to non-rigid objects such as T-shirts. We show that the proposed method achieves 74% and 57% attack success rates in the digital and physical worlds respectively against YOLOv2. In contrast, the state-of-the-art physical attack method to fool a person detector only achieves 18% attack success rate. Furthermore, by leveraging min-max optimization, we extend our method to the ensemble attack setting against two object detectors YOLO-v2 and Faster R-CNN simultaneously.

Keywords: Physical adversarial attack; object detection; deep learning

1 Introduction

The vulnerability of deep neural networks (DNNs) against adversarial attacks (namely, perturbed inputs deceiving DNNs) has been found in applications spanning from image classification to speech recognition [33,21,34,37,6,32,2]. Early works studied adversarial examples only in the digital space. Recently, some works showed that it is possible to create adversarial perturbations on physical objects and fool DNN-based decision makers under a variety of real-world conditions [28,14,1,15,25,7,30,5,20]. The design of *physical adversarial attacks* helps to evaluate the robustness of DNNs deployed in real-life systems, e.g., autonomous vehicles and surveillance systems. However, most of the studied physical adversarial attacks encounter two limitations: a) the physical objects are usually considered being *static*, and b) the possible *deformation* of adversarial pattern

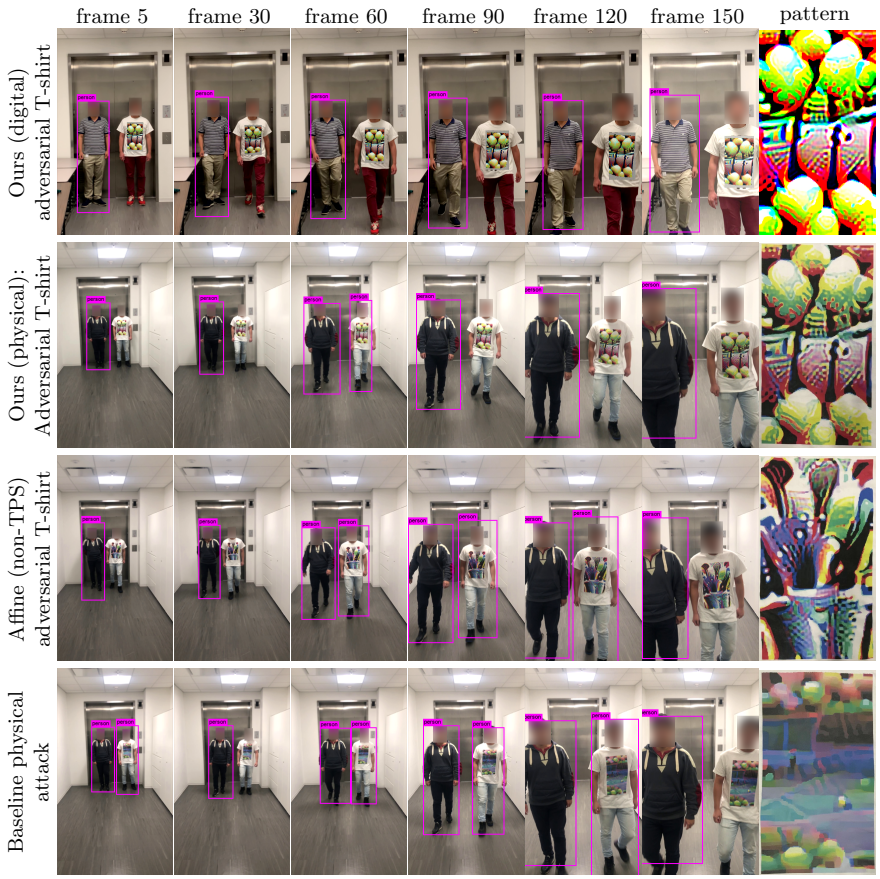


Fig. 1: Evaluation of the effectiveness of adversarial T-shirts to evade person detection by YOLOv2. Each row corresponds to a specific attack method while each column except the last one shows an individual frame in a video. The last column shows the adversarial patterns applied to the T-shirts. At each frame, there are two persons, one of whom wears the adversarial T-shirt. First row: digital adversarial T-shirt generated using TPS. Second row: physical adversarial T-shirt generated using TPS. Third row: physical adversarial T-shirt generated using affine transformation (namely, in the absence of TPS). Fourth row: T-shirt with physical adversarial patch considered in [30] to evade person detectors.

attached to a moving object (e.g., due to pose change of a moving person) is commonly neglected. In this paper, we propose a new type of physical adversarial attack, *adversarial T-shirt*, to evade DNN-based person detectors when a person wears the adversarial T-shirt; see the second row of Fig. 1 for illustrative examples.

Related work Most of the existing physical adversarial attacks are generated against image classifiers and object detectors. In [28], a face recognition system is fooled by a real eyeglass frame designed under a crafted adversarial pattern. In [14], a stop sign is misclassified by adding black or white stickers on it against the image classification system. In [20], an image classifier is fooled by placing a crafted sticker at the lens of a camera. In [1], a so-called Expectation over Transformation (EoT) framework was proposed to synthesize adversarial examples robust to a set of physical transformations such as rotation, translation, contrast, brightness, and random noise. Moreover, the crafted adversarial examples on the rigid objects can be designed in camouflage style [35] or natural style [11] that appear legitimate to human observers in the real world. Compared to attacking image classifiers, generating physical adversarial attacks against object detectors is more involved. For example, the adversary is required to mislead the bounding box detector of an object when attacking YOLOv2 [26] and SSD [24]. A well-known success of such attacks in the physical world is the generation of adversarial stop sign [15], which deceives state-of-the-art object detectors such as YOLOv2 and Faster R-CNN [27].

The most relevant approach to ours is the work of [30], which demonstrates that a person can evade a detector by holding a cardboard with an adversarial patch. However, such a physical attack restricts the adversarial patch to be attached to a *rigid* carrier (namely, cardboard), and is different from our setting here where the generated adversarial pattern is directly printed on a T-shirt. We show that the attack proposed by [30] becomes ineffective when the adversarial patch is attached to a T-shirt (rather than a cardboard) and worn by a moving person (see the fourth row of Fig. 1). At the technical side, different from [30] we propose a thin plate spline (TPS) based transformer to model deformation of non-rigid objects, and develop an ensemble physical attack that fools object detectors YOLOv2 and Faster R-CNN simultaneously. We highlight that our proposed adversarial T-shirt is not just a T-shirt with printed adversarial patch for clothing fashion, it is a physical adversarial wearable designed for evading person detectors in the real world.

Our work is also motivated by the importance of person detection on intelligent surveillance. DNN-based surveillance systems have significantly advanced the field of object detection [18,17]. Efficient object detectors such as Faster R-CNN [27], SSD [24], and YOLOv2 [26] have been deployed for human detection. Thus, one may wonder whether or not there exists a security risk for intelligent surveillance systems caused by adversarial human wearables, e.g., adversarial T-shirts. However, paralyzing a person detector in the physical world requires substantially more challenges such as low resolution, pose changes and occlusions. The success of our adversarial T-shirt against real-time person detectors offers new insights for designing practical physical-world adversarial human wearables.

Contributions We summarize our contributions as follows:

– We develop a TPS-based transformer to model the temporal deformation of an adversarial T-shirt caused by pose changes of a moving person. We

also show the importance of such non-rigid transformation to ensuring the effectiveness of adversarial T-shirts in the physical world.
– We propose a general optimization framework for design of adversarial T-shirts in both single-detector and multiple-detector settings.
– We conduct experiments in both digital and physical worlds and show that the proposed adversarial T-shirt achieves 74% and 57% attack success rates respectively when attacking YOLOv2. By contrast, the physical adversarial patch [30] printed on a T-shirt only achieves 18% attack success rate. Some of our results are highlighted in Fig. 1.

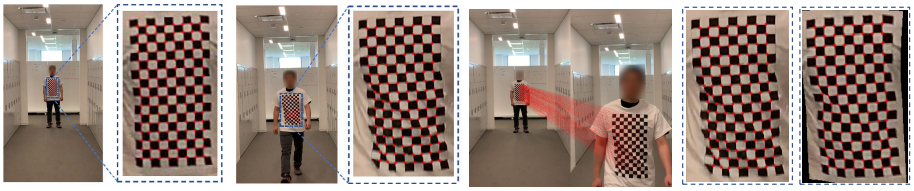
2 Modeling Deformation of A Moving Object by Thin Plate Spline Mapping

In this section, we begin by reviewing some existing transformations required in the design of physical adversarial examples. We then elaborate on the Thin Plate Spline (TPS) mapping we adopt in this work to model the possible deformation encountered by a moving and non-rigid object.

Let $\mathbf{x}$ be an original image (or a video frame), and $t(\cdot)$ be the physical transformer. The transformed image $\mathbf{z}$ under t is given by

$$\mathbf{z} = t(\mathbf{x}). \tag{1}$$

Existing transformations. In [1], the parametric transformers include scaling, translation, rotation, brightness and additive Gaussian noise; see details in [1, Appendix D]. In [23], the geometry and lighting transformations are studied via parametric models. Other transformations including perspective transformation, brightness adjustment, resampling (or image resizing), smoothing and saturation are considered in [29,9]. All the existing transformations are included in our library of physical transformations. However, they are not sufficient to model the cloth deformation caused by pose change of a moving person. For example, the second and third rows of Fig. 1 show that adversarial T-shirts designed against only existing physical transformations yield low attack success rates.



(a) (b) (c) (d)

Fig. 2: Generation of TPS. (a) and (b): Two frames with checkerboard detection results. (c): Anchor point matching process between two frames (d): Real-world close deformation in (b) versus the synthesized TPS transformation (right plot).

TPS transformation for cloth deformation. A person's movement can result in significantly and constantly changing wrinkles (aka deformations) in her clothes. This makes it challenging to develop an adversarial T-shirt effectively in the real world. To circumvent this challenge, we employ TPS mapping [4] to model the cloth deformation caused by human body movement. TPS has been widely used as the non-rigid transformation model in image alignment and shape matching [19]. It consists of an affine component and a non-affine warping component. We will show that the non-linear warping part in TPS can provide an effective means of modeling cloth deformation for learning adversarial patterns of non-rigid objects.

TPS learns a parametric deformation mapping from an original image $\mathbf{x}$ to a target image $\mathbf{z}$ through a set of control points with given positions. Let $\mathbf{p} := (\phi, \psi)$ denote the 2D location of an image pixel. The deformation from $\mathbf{x}$ to $\mathbf{z}$ is then characterized by the *displacement* of every pixel, namely, how a pixel at $\mathbf{p}^{(x)}$ on image $\mathbf{x}$ changes to the pixel on image $\mathbf{z}$ at $\mathbf{p}^{(z)}$, where $\phi^{(z)} = \phi^{(x)} + \Delta_\phi$ and $\psi^{(z)} = \psi^{(x)} + \Delta_\psi$, and Δ_ϕ and Δ_ψ denote the pixel displacement on image $\mathbf{x}$ along ϕ direction and ψ direction, respectively.

Given a set of n control points with locations $\{\hat{\mathbf{p}}_i^{(x)} := (\hat{\phi}_i^{(x)}, \hat{\psi}_i^{(x)})\}_{i=1}^n$ on image $\mathbf{x}$, TPS provides a parametric model of pixel displacement when mapping $\mathbf{p}^{(x)}$ to $\mathbf{p}^{(z)}$ [8]

$$\Delta(\mathbf{p}^{(x)}; \boldsymbol{\theta}) = a_0 + a_1\phi^{(x)} + a_2\psi^{(x)} + \sum_{i=1}^{n} c_i U(\|\hat{\mathbf{p}}_i^{(x)} - \mathbf{p}^{(x)}\|_2), \tag{2}$$

where $U(r) = r^2 \log(r)$ and $\boldsymbol{\theta} = [\mathbf{c}; \mathbf{a}]$ are the TPS parameters, and $\Delta(\mathbf{p}^{(x)}; \boldsymbol{\theta})$ represents the displacement along either ϕ or ψ direction.

Moreover, given the locations of control points on the transformed image $\mathbf{z}$ (namely, $\{\hat{\mathbf{p}}_i^{(z)}\}_{i=1}^n$), TPS resorts to a regression problem to determine the parameters $\boldsymbol{\theta}$ in (2). The regression objective is to minimize the distance between $\{\Delta_\phi(\mathbf{p}_i^{(x)}; \boldsymbol{\theta}_\phi)\}_{i=1}^n$ and $\{\hat{\Delta}_{\phi,i} := \hat{\phi}_i^{(z)} - \hat{\phi}_i^{(x)}\}_{i=1}^n$ along the ϕ direction, and the distance between $\{\Delta_\psi(\mathbf{p}_i^{(x)}; \boldsymbol{\theta}_\psi)\}_{i=1}^n$ and $\{\hat{\Delta}_{\psi,i} := \hat{\psi}_i^{(z)} - \hat{\psi}_i^{(x)}\}_{i=1}^n$ along the ψ direction, respectively. Thus, TPS (2) is applied to coordinate ϕ and ψ separately (corresponding to parameters $\boldsymbol{\theta}_\phi$ and $\boldsymbol{\theta}_\psi$). The regression problem can be solved by the following linear system of equations [10]

$$\begin{bmatrix} \mathbf{K} & \mathbf{P} \\ \mathbf{P}^T & \mathbf{0}_{3\times 3} \end{bmatrix} \boldsymbol{\theta}_\phi = \begin{bmatrix} \hat{\boldsymbol{\Delta}}_\phi \\ \mathbf{0}_{3\times 1} \end{bmatrix}, \ \begin{bmatrix} \mathbf{K} & \mathbf{P} \\ \mathbf{P}^T & \mathbf{0}_{3\times 3} \end{bmatrix} \boldsymbol{\theta}_\psi = \begin{bmatrix} \hat{\boldsymbol{\Delta}}_\psi \\ \mathbf{0}_{3\times 1} \end{bmatrix}, \tag{3}$$

where the (i,j)th element of $\mathbf{K} \in \mathbb{R}^{n\times n}$ is given by $K_{ij} = U(\|\hat{\mathbf{p}}_i^{(x)} - \hat{\mathbf{p}}_j^{(x)}\|_2)$, the ith row of $\mathbf{P} \in \mathbb{R}^{n\times 3}$ is given by $P_i = [1, \hat{\phi}_i^{(x)}, \hat{\psi}_i^{(x)}]$, and the ith elements of $\hat{\boldsymbol{\Delta}}_\phi \in \mathbb{R}^n$ and $\hat{\boldsymbol{\Delta}}_\psi \in \mathbb{R}^n$ are given by $\hat{\Delta}_{\phi,i}$ and $\hat{\Delta}_{\psi,i}$, respectively.

Non-trivial application of TPS The difficulty of implementing TPS for design of adversarial T-shirts exists from two aspects: 1) How to determine the set of

control points? And 2) how to obtain positions $\{\hat{\mathbf{p}}_i^{(x)}\}$ and $\{\hat{\mathbf{p}}_i^{(z)}\}$ of control points aligned between a pair of video frames $\mathbf{x}$ and $\mathbf{z}$?

To address the first question, we print a *checkerboard* on a T-shirt and use the camera calibration algorithm [16,36] to detect points at the intersection between every two checkerboard grid regions. These successfully detected points are considered as the control points of one frame. Fig. 2-(a) shows the checkerboard-printed T-shirt, together with the detected intersection points. Since TPS requires a set of control points *aligned* between two frames, the second question on point matching arises. The challenge lies in the fact that the control points detected at one video frame are different from those at another video frame (e.g., due to missing detection). To address this issue, we adopt a 2-stage procedure, *coordinate system alignment* followed by *point aliment*, where the former refers to conducting a perspective transformation from one frame to the other, and the latter finds the matched points at two frames through the nearest-neighbor method. We provide an illustrative example in Fig. 2-(c). We refer readers to Appendix A for more details about our method.

3 Generation of Adversarial T-shirt: An Optimization Perspective

In this section, we begin by formalizing the problem of adversarial T-shirt and introducing notations used in our setup. We then propose to design a *universal* perturbation used in our adversarial T-shirt to deceive a *single* object detector. We lastly propose a min-max (robust) optimization framework to design the universal adversarial patch against *multiple* object detectors.

Let $\mathcal{D} := \{\mathbf{x}_i\}_{i=1}^M$ denote M video frames extracted from one or multiple given videos, where $\mathbf{x}_i \in \mathbb{R}^d$ denotes the ith frame. Let $\boldsymbol{\delta} \in \mathbb{R}^d$ denote the universal adversarial perturbation applied to $\mathcal{D}$. The adversarial T-shirt is then characterized by $M_{c,i} \circ \boldsymbol{\delta}$, where $M_{c,i} \in \{0,1\}^d$ is a bounding box encoding the position of the cloth region to be perturbed at the ith frame, and $\circ$ denotes element-wise product. *The goal of adversarial T-shirt is to design $\boldsymbol{\delta}$ such that the perturbed frames of $\mathcal{D}$ are mis-detected by object detectors.*

Fooling a single object detector. We generalize the Expectation over Transformation (EoT) method in [3] for design of adversarial T-shirts. Note that different from the conventional EoT, a transformers' composition is required for generating an adversarial T-shirt. For example, a perspective transformation on the bounding box of the T-shirt is composited with an TPS transformation applied to the cloth region. Let us begin by considering two video frames, an anchor image $\mathbf{x}_0$ (e.g., the first frame in the video) and a target image $\mathbf{x}_i$ for $i \in [M]$[1]. Given the bounding boxes of the person ($M_{p,0} \in \{0,1\}^d$) and the T-shirt ($M_{c,0} \in \{0,1\}^d$) at $\mathbf{x}_0$, we apply the perspective transformation from $\mathbf{x}_0$ to $\mathbf{x}_i$ to obtain the bounding boxes $M_{p,i}$ and $M_{c,i}$ at image $\mathbf{x}_i$. In the *absence*

[1] $[M]$ denotes the integer set $\{1, 2, \ldots, M\}$.

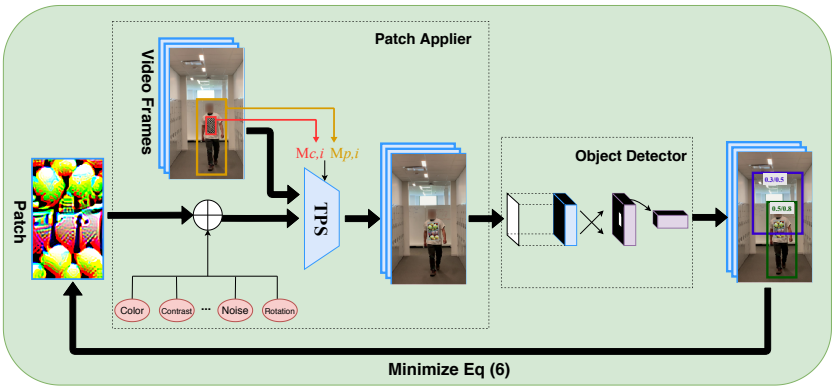


Fig. 3: Overview of the pipeline to generate adversarial T-shirts. First, the video frames containing a person whom wears the T-shirt with printed checkerboard pattern are used as training data. Second, the universal adversarial perturbation (to be designed) applies to the cloth region by taking into account different kinds of transformations. Third, the adversarial perturbation is optimized through problem (6) by minimizing the largest bounding-box probability belonging to the 'person' class. The optimization procedure is performed as a closed loop through back-propagation.

of physical transformations, the perturbed image $\mathbf{x}_i'$ with respect to (w.r.t.) $\mathbf{x}_i$ is given by

$$\mathbf{x}_i' = \underbrace{(\mathbf{1} - M_{p,i}) \circ \mathbf{x}_i}_{\text{A}} + \underbrace{M_{p,i} \circ \mathbf{x}_i}_{\text{B}} - \underbrace{M_{c,i} \circ \mathbf{x}_i}_{\text{C}} + \underbrace{M_{c,i} \circ \boldsymbol{\delta}}_{\text{D}}, \tag{4}$$

where the term A denotes the background region outside the bouding box of the person, the term B is the person-bounded region, the term C erases the pixel values within the bounding box of the T-shirt, and the term D is the newly introduced additive perturbation. In (4), the prior knowledge on $M_{p,i}$ and $M_{c,i}$ is acquired by person detector and manual annotation, respectively. Without taking into account physical transformations, Eq. (4) simply reduces to the conventional formulation of adversarial example $(1 - M_{c,i}) \circ \mathbf{x}_i + M_{c,i} \circ \boldsymbol{\delta}$.

Next, we consider *three main types* of physical transformations: a) TPS transformation $t_{\text{TPS}} \in \mathcal{T}_{\text{TPS}}$ applying to the adversarial perturbation $\boldsymbol{\delta}$ for modeling the effect of cloth deformation, b) physical color transformation t_{color} which converts digital colors to those printed and visualized in the physical world, and c) conventional physical transformation $t \in \mathcal{T}$ applying to the region within the person's bounding box, namely, $(M_{p,i} \circ \mathbf{x}_i - M_{c,i} \circ \mathbf{x}_i + M_{c,i} \circ \boldsymbol{\delta})$. Here $\mathcal{T}_{\text{TPS}}$ denotes the set of possible non-rigid transformations, t_{color} is given by a regression model learnt from the color spectrum in the digital space to its printed counterpart, and $\mathcal{T}$ denotes the set of commonly-used physical transformations, e.g., scaling, translation, rotation, brightness, blurring and contrast. A modification of (4) under different sources of transformations is then given by

$$\mathbf{x}_i' = t_{\text{env}}\left(\text{A} + t\left(\text{B} - \text{C} + t_{\text{color}}(M_{c,i} \circ t_{\text{TPS}}(\boldsymbol{\delta} + \mu\mathbf{v}))\right)\right) \tag{5}$$

for $t \in \mathcal{T}$, $t_{\text{TPS}} \in \mathcal{T}_{\text{TPS}}$, and $\mathbf{v} \sim \mathcal{N}(0,1)$. In (5), the terms A, B and C have been defined in (4), and t_{env} denotes a brightness transformation to model the environmental brightness condition. In (5), $\mu\mathbf{v}$ is an additive Gaussian noise that allows the variation of pixel values, where μ is a given smoothing parameter and we set it as 0.03 in our experiments such that the noise realization falls into the range $[-0.1, 0.1]$. The randomized noise injection is also known as Gaussian smoothing [12], which makes the final objective function smoother and benefits the gradient computation during optimization.

Different with the prior works, e.g. [28,13], established a non-printability score (NPS) to measure the distance between the designed perturbation vector and a library of printable colors, we propose to model the color transformer t_{color} using a quadratic polynomial regression. The detailed color mapping is showed in Appendix B.

With the aid of (5), the EoT formulation to fool a single object detector is cast as

$$\underset{\boldsymbol{\delta}}{\text{minimize}} \ \frac{1}{M}\sum_{i=1}^{M} \mathbb{E}_{t,t_{\text{TPS}},\mathbf{v}}\left[f(\mathbf{x}_i')\right] + \lambda g(\boldsymbol{\delta}) \tag{6}$$

where f denotes an attack loss for misdetection, g is the total-variation norm that enhances perturbations' smoothness [15], and $\lambda > 0$ is a regularization parameter. We further elaborate on our attack loss f in problem (6). In YOLOv2, a probability score associated with a bounding box indicates whether or not an object is present within this box. Thus, we specify the attack loss as the largest bounding-box probability over all bounding boxes belonging to the 'person' class. For Faster R-CNN, we attack all bounding boxes towards the class 'background'. The more detailed derivation on the attack loss is provided in Appendix C. Fig. 3 presents an overview of our approach to generate adversarial T-shirts.

Min-max optimization for fooling multiple object detectors. Unlike digital space, the transferability of adversarial attacks largely drops in the physical environment, thus we consider a *physical ensemble attack* against multiple object detectors. It was recently shown in [31] that the ensemble attack can be designed from the perspective of min-max optimization, and yields much higher worst-case attack success rate than the averaging strategy over multiple models. Given N object detectors associated with attack loss functions $\{f_i\}_{i=1}^N$, the physical ensemble attack is cast as

$$\underset{\boldsymbol{\delta}\in\mathcal{C}}{\text{minimize}} \ \underset{\mathbf{w}\in\mathcal{P}}{\text{maximize}} \ \sum_{i=1}^{N} w_i\phi_i(\boldsymbol{\delta}) - \frac{\gamma}{2}\|\mathbf{w} - \mathbf{1}/N\|_2^2 + \lambda g(\boldsymbol{\delta}), \tag{7}$$

where $\mathbf{w}$ are known as domain weights that adjust the importance of each object detector during the attack generation, $\mathcal{P}$ is a probabilistic simplex given by $\mathcal{P} = \{\mathbf{w}|\mathbf{1}^T\mathbf{w} = 1, \mathbf{w} \geq \mathbf{0}\}$, $\gamma > 0$ is a regularization parameter, and $\phi_i(\boldsymbol{\delta}) := \frac{1}{M}\sum_{i=1}^{M} \mathbb{E}_{t\in\mathcal{T},t_{\text{TPS}}\in\mathcal{T}_{\text{TPS}}}\left[f(\mathbf{x}_i')\right]$ following (6). In (7), if $\gamma = 0$, then the adversarial perturbation $\boldsymbol{\delta}$ is designed over the *maximum* attack loss (worst-case attack scenario) since $\text{maximize}_{\mathbf{w}\in\mathcal{P}} \sum_{i=1}^{N} w_i\phi_i(\boldsymbol{\delta}) = \phi_{i^*}(\boldsymbol{\delta})$, where $i^* = \arg\max_i \phi_i(\boldsymbol{\delta})$ at a fixed $\boldsymbol{\delta}$. Moreover, if $\gamma \to \infty$, then the inner maximization of problem (7)

implies $\mathbf{w} \to \mathbf{1}/N$, namely, an averaging scheme over M attack losses. Thus, the regularization parameter γ in (7) strikes a balance between the max-strategy and the average-strategy.

4 Experiments

In this section, we demonstrate the effectiveness of our approach (we call *advT-TPS*) for design of the adversarial T-shirt by comparing it with 2 attack baseline methods, a) adversarial patch to fool YOLOv2 proposed in [30] and its printed version on a T-shirt (we call *advPatch*[2]), and b) the variant of our approach in the absence of TPS transformation, namely, $\mathcal{T}_{\text{TPS}} = \emptyset$ in (5) (we call *advT-Affine*). We examine the convergence behavior of proposed algorithms as well as its Attack Success Rate[3] (ASR) in both digital and physical worlds. We clarify our algorithmic parameter setting in Appendix D.

Prior to detailed illustration, we briefly summarize the attack performance of our proposed adversarial T-shirt. When attacking YOLOv2, our method achieves 74% ASR in the digital world and 57% ASR in the physical world, where the latter is computed by averaging successfully attacked video frames over all different scenarios (i.e., indoor, outdoor and unforeseen scenarios) listed in Table 2. When attacking Faster R-CNN, our method achieves 61% and 47% ASR in the digital and the physical world, respectively. By contrast, the baseline advPatch only achieves around 25% ASR in the best case among all digital and physical scenarios against either YOLOv2 or Faster R-CNN (e.g., 18% against YOLOv2 in the physical case).

4.1 Experimental Setup

Data collection. We collect two datasets for learning and testing our proposed attack algorithm in digital and physical worlds. The training dataset contains 40 videos (2003 video frames) from 4 different scenes: one outdoor and three indoor scenes. each video takes 5-10 seconds and was captured by a moving person wearing a T-shirt with printed checkerboard. The desired adversarial pattern is then learnt from the training dataset. The test dataset in the digital space contains 10 videos captured under the same scenes as the training dataset. This dataset is used to evaluate the attack performance of the learnt adversarial pattern in the digital world. In the physical world, we customize a T-shirt with the printed adversarial pattern learnt from our algorithm. Another 24 test videos (Section 4.3) are then collected at a different time capturing two or three persons (one of them wearing the adversarial T-shirt) walking a) side by side or b) at different distances. An additional control experiment in which actors wearing adversarial T-shirts walk in an exaggerated way is conducted to introduce large

[2]For fair comparison, we modify the perturbation size same as ours and execute the code provided in [30] under our training dataset.

[3]ASR is given by the ratio of successfully attacked testing frames over the total number of testing frames.

pose changes in the test data. In addition, we also test our adversarial T-shirt by unforeseen scenarios, where the test videos involve different locations and different persons which are never covered in the training dataset. All videos were taken using an iPhone X and resized to 416 × 416. In Table A2 of the Appendix F, we summarize the collected dataset under all circumstances.

Object detectors. We use two state-of-the-art object detectors: Faster R-CNN [27] and YOLOv2 [26] to evaluate our method. These two object detectors are both pre-trained on COCO dataset [22] which contains 80 classes including 'person'. The minimum detection threshold are set as 0.7 for both Faster R-CNN and YOLOv2 by default. The sensitivity analysis of this threshold is performed in Fig. A4 Appendix D.

4.2 Adversarial T-shirt in the digital world

Convergence performance of our proposed attack algorithm. In Fig. 4, we show ASR against the epoch number used by our proposed algorithm to solve problem (6). Here the success of our attack at one testing frame is required to meet two conditions, a) misdetection of the person who wears the adversarial T-shirt, and b) successful detection of the person whom dresses a normal cloth. As we can see, the proposed attack method covnerges well for attacking both YOLOv2 and Faster R-CNN. We also note that attacking Faster R-CNN is more difficult than attacking YOLOv2. Furthermore, if TPS is not applied during training, then ASR drops around 30% compared to our approach by leveraging TPS.

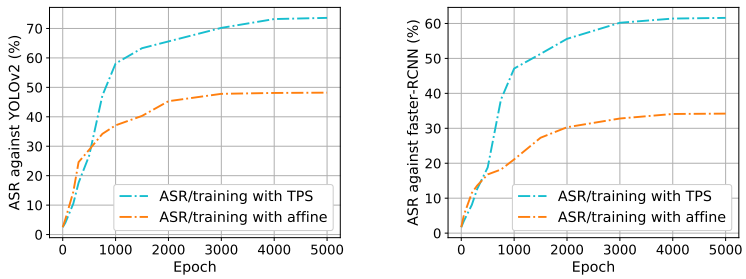


Fig. 4: ASR v.s. epoch numbers against YOLOv2 (left) and Faster R-CNN (right).

ASR of adversarial T-shirts in various attack settings. We perform a more comprehensive evaluation on our methods by digital simulation. Table 1 compares the ASR of adversarial T-shirts generated w/ or w/o TPS transformation in 4 attack settings: a) *single-detector attack* referring to adversarial T-shirts designed and evaluated using the same object detector, b) *transfer single-detector attack* referring to adversarial T-shirts designed and evaluated using different object

detectors, c) *ensemble attack (average)* given by (7) but using the average of attack losses of individual models, and d) *ensemble attack (min-max)* given by (7). As we can see, it is crucial to incorporate TPS transformation in the design of adversarial T-shirts: without TPS, the ASR drops from 61% to 34% when attacking Faster R-CNN and drops from 74% to 48% when attacking YOLOv2 in the single-detector attack setting. We also note that the transferability of single-detector attack is weak in all settings. And Faster R-CNN is consistently more robust than YOLOv2, similar to the results in Fig. 4. Compared to our approach and *advT-Affine*, the baseline method *advPatch* yields the worst ASR when attacking a single detector. Furthermore, we evaluate the effectiveness of the proposed min-max ensemble attack (7). As we can see, when attacking Faster R-CNN, the min-max ensemble attack significantly outperforms its counterpart using the averaging strategy, leading to 15% improvement in ASR. This improvement is at the cost of 7% degradation when attacking YOLOv2.

Table 1: The ASR (%) of adversarial T-shirts generated from our approach, *advT-Affine* and the baseline *advPatch* in digital-world against Faster R-CNN and YOLOv2.

method	model	target	transfer	ensemble(average)	ensemble(min-max)
advPatch[30]	Faster R-CNN	22%	10%	N/A	N/A
advT-Affine		34%	11%	16%	32%
advT-TPS(ours)		**61%**	10%	**32%**	**47%**
advPatch[30]	YOLOv2	24%	10%	N/A	N/A
advT-Affine		48%	13%	31%	27%
advT-TPS(ours)		**74%**	13%	**60%**	**53%**

4.3 Adversarial T-shirt in the physical world

We next evaluate our method in the physical world. First, we generate an adversarial pattern by solving problem (6) against YOLOv2 and Faster R-CNN, following Section 4.2. We then print the adversarial pattern on a white T-shirt, leading to the adversarial T-shirt. For fair comparison, we also print adversarial patterns generated by the *advPatch* [30] and *advT-Affine* in Section 4.2 on white T-shirts of the same style. It is worth noting that different from evaluation by taking static photos of physical adversarial examples, our evaluation is conducted at a more practical and challenging setting. That is because we record videos to track a moving person wearing adversarial T-shirts, which could encounter multiple environment effects such as distance, deformation of the T-shirt, poses and angles of the moving person.

In Table 2, we compare our method with *advPatch* and *advT-Affine* under 3 specified scenarios, including the indoor, outdoor, and unforeseen scenarios[4], to-

[4]Unforeseen scenarios refer to test videos that involve different locations and actors that never seen in the training dataset.

gether with the overall case of all scenarios. We observe that our method achieves 64% ASR (against YOLOv2), which is much higher than *advT-Affine* (39%) and *advPatch* (19%) in the indoor scenario. Compared to the indoor scenario, evading person detectors in the outdoor scenario becomes more challenging. The ASR of our approach reduces to 47% but outperforms *advT-Affine* (36%) and *advPatch* (17%). This is not surprising since the outdoor scenario suffers more environmental variations such as lighting change. Even considering the unforeseen scenario, we find that our adversarial T-shirt is robust to the change of person and location, leading to 48% ASR against Faster R-CNN and 59% ASR against YOLOv2. Compared to the digital results, the ASR of our adversarial T-shirt drops around 10% in all tested physical-world scenarios; see specific video frames in Fig. A5 in Appendix.

Table 2: The ASR (%) of adversarial T-shirts generated from our approach, *advT-Affine* and *advPatch* under different physical-world scenes.

method	model	indoor	outdoor	new scenes	average ASR
advPatch[30]	Faster R-CNN	15%	16%	12%	14%
advT-Affine		27%	25%	25%	26%
advT-TPS(ours)		**50%**	**42%**	**48%**	**47%**
advPatch[30]	YOLOv2	19%	17%	17%	18%
advT-Affine		39%	36%	34%	37%
advT-TPS(ours)		**64%**	**47%**	**59%**	**57%**

4.4 Ablation Study

In this section, we conduct more experiments for better understanding the robustness of our adversarial T-shirt against various conditions including angles and distances to camera, camera view, person's pose, and complex scenes that include crowd and occlusion. Since the baseline method (*advPatch*) performs poorly in most of these scenarios, we focus on evaluating our method (*advT-TPS*) against *advT-Affine* using YOLOv2. We refer readers to Appendix E for details on the setup of our ablation study.

Angles and distances to camera. In Fig. 5, we present ASRs of *advT-TPS* and *advT-Affine* when the actor whom wears the adversarial T-shit at different angles and distances to the camera. As we can see, *advT-TPS* works well within the angle 20° and the distance 4m. And *advT-TPS* consistently outperforms *advT-Affine*. We also note that ASR drops significantly at the angle 30° since it induces occlusion of the adversarial pattern. Further, if the distance is greater than 7m, the pattern cannot clearly be seen from the camera.

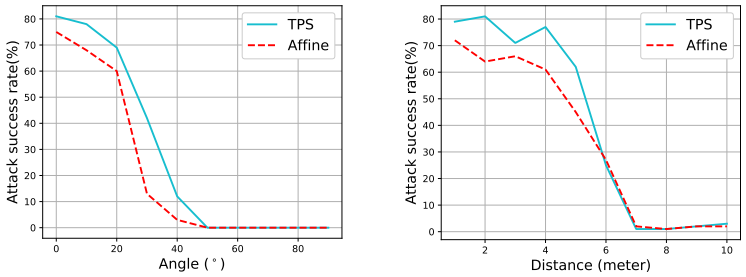


Fig. 5: Average ASR v.s. different angles (left) and distance (right).

Human Pose. In Table 3 (left), we evaluate the effect of pose change on *advT-TPS*, where videos are taken for an actor with some distinct postures including crouching, siting and running in place; see Fig. 6 for specific examples. To alleviate other latent effects, the camera was made to look straight at the person at a fixed distance of about $1 \sim 2$m away from the person. As we can see, *advT-TPS* consistently outperforms *advT-Affine*. In additional, we study the effect of occlusion on *advT-Affine* and *advT-TPS* in Appendix F.

Complex scenes. In Table 3 (right), we test our adversarial T-shirt in several complex scenes with cluttered backgrounds, including a) an office with multiple objects and people moving around; b) a parking lot with vehicles and pedestrians; and c) a crossroad with busy traffic and crowd. We observe that compared to *advT-Affine*, *advT-TPS* is reasonably effective in complex scenes without suffering a significant loss of ASR. Compared to the other factors such as camera angle and occlusion, cluttered background and even crowd are probably the least of a concern for our approach. This is explainable, as our approach works on object proposals directly to suppress the classifier.

Table 3: The ASR (%) of adversarial T-shirts generated from our approach, *advT-Affine* and *advPatch* under different physical-world scenarios.

Method \ Pose	crouching	siting	running
advT-Affine	27%	26%	52%
advT-TPS	**53%**	**32%**	**63%**

Method \ Scenario	office	parking lot	crossroad
advT-Affine	69%	53%	51%
advT-TPS	**73%**	**65%**	**54%**

5 Conclusion

In this paper, we propose *Adversarial T-shirt*, the first successful adversarial wearable to evade detection of moving persons. Since T-shirt is a non-rigid ob-

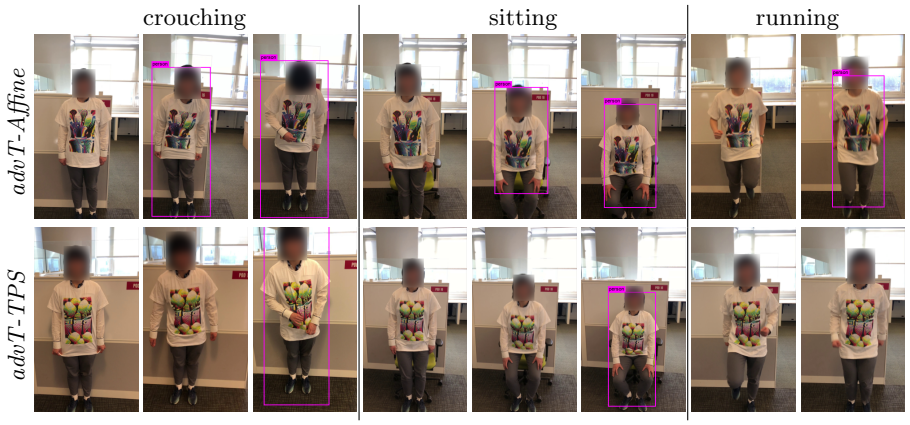


Fig. 6: Some video frames of person who wears adversarial T-shirt generated by *advT-Affine* (first row) and *advT-TPS* (second row) with different poses.

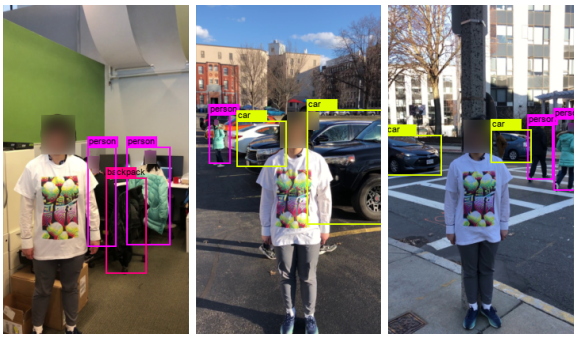


Fig. 7: The person who wear our adversarial T-shirt generate by TPS in three complex scenes: office, parking lot and crossroad.

ject, its deformation induced by a person's pose change is taken into account when generating adversarial perturbations. We also propose a min-max ensemble attack algorithm to fool multiple object detectors simultaneously. We show that our attack against YOLOv2 can achieve 74% and 57% attack success rate in the digital and physical world, respectively. By contrast, the *advPatch* method can only achieve 24% and 18% ASR. Based on our studies, we hope to provide some implications on how the adversarial perturbations can be implemented in physical worlds.

Acknowledgement

This work is partly supported by the National Science Foundation CNS-1932351. We would also like to extend our gratitude to the MIT-IBM Watson AI Lab.

References

1. Athalye, A., Engstrom, L., Ilyas, A., Kwok, K.: Synthesizing robust adversarial examples. In: Dy, J., Krause, A. (eds.) Proceedings of the 35th International Conference on Machine Learning. vol. 80, pp. 284–293 (10–15 Jul 2018)
2. Athalye, A., Carlini, N., Wagner, D.: Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. arXiv preprint arXiv:1802.00420 (2018)
3. Athalye, A., Engstrom, L., Ilyas, A., Kwok, K.: Synthesizing robust adversarial examples. In: International Conference on Machine Learning. pp. 284–293 (2018)
4. Bookstein, F.L.: Principal warps: Thin-plate splines and the decomposition of deformations. IEEE Transactions on pattern analysis and machine intelligence **11**(6), 567–585 (1989)
5. Cao, Y., Xiao, C., Yang, D., Fang, J., Yang, R., Liu, M., Li, B.: Adversarial objects against lidar-based autonomous driving systems. arXiv preprint arXiv:1907.05418 (2019)
6. Carlini, N., Wagner, D.: Audio adversarial examples: Targeted attacks on speech-to-text. In: 2018 IEEE Security and Privacy Workshops (SPW). pp. 1–7. IEEE (2018)
7. Chen, S.T., Cornelius, C., Martin, J., Chau, D.H.P.: Shapeshifter: Robust physical adversarial attack on faster r-cnn object detector. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases. pp. 52–68. Springer (2018)
8. Chui, H.: Non-rigid point matching: algorithms, extensions and applications. Citeseer (2001)
9. Ding, G.W., Lui, K.Y.C., Jin, X., Wang, L., Huang, R.: On the sensitivity of adversarial robustness to input data distributions. In: International Conference on Learning Representations (2019)
10. Donato, G., Belongie, S.: Approximate thin plate spline mappings. In: European conference on computer vision. pp. 21–31. Springer (2002)
11. Duan, R., Ma, X., Wang, Y., Bailey, J., Qin, A.K., Yang, Y.: Adversarial camouflage: Hiding physical-world attacks with natural styles. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1000–1008 (2020)
12. Duchi, J.C., Bartlett, P.L., Wainwright, M.J.: Randomized smoothing for stochastic optimization. SIAM Journal on Optimization **22**(2), 674–701 (2012)
13. Evtimov, I., Eykholt, K., Fernandes, E., Kohno, T., Li, B., Prakash, A., Rahmati, A., Song, D.: Robust physical-world attacks on machine learning models. arXiv preprint arXiv:1707.08945 (2017)
14. Eykholt, K., Evtimov, I., Fernandes, E., Li, B., Rahmati, A., Xiao, C., Prakash, A., Kohno, T., Song, D.: Robust physical-world attacks on deep learning visual classification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1625–1634 (2018)
15. Eykholt, K., Evtimov, I., Fernandes, E., Li, B., Rahmati, A., Tramer, F., Prakash, A., Kohno, T., Song, D.: Physical adversarial examples for object detectors. In: 12th USENIX Workshop on Offensive Technologies (WOOT 18) (2018)
16. Geiger, A., Moosmann, F., Car, Ö., Schuster, B.: Automatic camera and range sensor calibration using a single shot. In: 2012 IEEE International Conference on Robotics and Automation. pp. 3936–3943. IEEE (2012)

17. Girshick, R.: Fast r-cnn. In: Proceedings of the IEEE international conference on computer vision. pp. 1440–1448 (2015)
18. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 580–587 (2014)
19. Jaderberg, M., Simonyan, K., Zisserman, A., et al.: Spatial transformer networks. In: Advances in neural information processing systems. pp. 2017–2025 (2015)
20. Li, J., Schmidt, F., Kolter, Z.: Adversarial camera stickers: A physical camera-based attack on deep learning systems. In: International Conference on Machine Learning. pp. 3896–3904 (2019)
21. Lin, J., Gan, C., Han, S.: Defensive quantization: When efficiency meets robustness. In: International Conference on Learning Representations (2019), https://openreview.net/forum?id=ryetZ20ctX
22. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: European conference on computer vision. pp. 740–755. Springer (2014)
23. Liu, H.T.D., Tao, M., Li, C.L., Nowrouzezahrai, D., Jacobson, A.: Beyond pixel norm-balls: Parametric adversaries using an analytically differentiable renderer. In: International Conference on Learning Representations (2019), https://openreview.net/forum?id=SJl2niR9KQ
24. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y., Berg, A.C.: Ssd: Single shot multibox detector. In: European conference on computer vision. pp. 21–37. Springer (2016)
25. Lu, J., Sibai, H., Fabry, E.: Adversarial examples that fool detectors. arXiv preprint arXiv:1712.02494 (2017)
26. Redmon, J., Farhadi, A.: Yolo9000: better, faster, stronger. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 7263–7271 (2017)
27. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. In: Advances in neural information processing systems. pp. 91–99 (2015)
28. Sharif, M., Bhagavatula, S., Bauer, L., Reiter, M.K.: Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition. In: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security. pp. 1528–1540. ACM (2016)
29. Sitawarin, C., Bhagoji, A.N., Mosenia, A., Mittal, P., Chiang, M.: Rogue signs: Deceiving traffic sign recognition with malicious ads and logos. arXiv preprint arXiv:1801.02780 (2018)
30. Thys, S., Van Ranst, W., Goedemé, T.: Fooling automated surveillance cameras: adversarial patches to attack person detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. pp. 0–0 (2019)
31. Wang, J., Zhang, T., Liu, S., Chen, P.Y., Xu, J., Fardad, M., Li, B.: Beyond adversarial training: Min-max optimization in adversarial attack and defense. arXiv preprint arXiv:1906.03563 (2019)
32. Xu, K., Chen, H., Liu, S., Chen, P.Y., Weng, T.W., Hong, M., Lin, X.: Topology attack and defense for graph neural networks: An optimization perspective. In: International Joint Conference on Artificial Intelligence (IJCAI) (2019)
33. Xu, K., Liu, S., Zhang, G., Sun, M., Zhao, P., Fan, Q., Gan, C., Lin, X.: Interpreting adversarial examples by activation promotion and suppression. arXiv preprint arXiv:1904.02057 (2019)

34. Xu, K., Liu, S., Zhao, P., Chen, P.Y., Zhang, H., Fan, Q., Erdogmus, D., Wang, Y., Lin, X.: Structured adversarial attack: Towards general implementation and better interpretability. In: International Conference on Learning Representations (2019)
35. Zhang, Y., Foroosh, H., David, P., Gong, B.: CAMOU: Learning physical vehicle camouflages to adversarially attack detectors in the wild. In: International Conference on Learning Representations (2019), https://openreview.net/forum?id=SJgEl3A5tm
36. Zhang, Z.: A flexible new technique for camera calibration. IEEE Transactions on pattern analysis and machine intelligence **22** (2000)
37. Zhao, P., Xu, K., Liu, S., Wang, Y., Lin, X.: Admm attack: an enhanced adversarial attack for deep neural networks with undetectable distortions. In: Proceedings of the 24th Asia and South Pacific Design Automation Conference. pp. 499–505. ACM (2019)