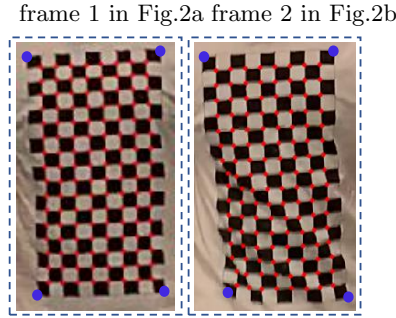


## Appendix

In the supplement, we provide details on the thin plate spline (TPS) transformation, the formulation of attack loss, the setting of algorithmic parameters, and the additional experiments of the adversarial T-shirt in the physical world.

### A How to construct TPS transformation?



**Fig. A1:** Four manually annotated corner points (blue) used to generate the bounding box of cloth region at frame  $i$ , namely,  $M_{c,i}$ . And  $8 \times 16$  anchor points (red) on the checkerboard used to generate TPS transformation  $t_{\text{TPS}}$  between two video frames.

We first manually annotate four corner points (see blue markers in Figure A1) to conduct a perspective transformation between two frames at different time instants. This perspective transformation is used to align the coordinate system of anchor points used for TPS transformation between two frames.

Ideally, the checkerboard detection tool [16,37] always outputs a grid of corner points detected. In most cases, it can locate all the  $8 \times 16$  points on the checkerboard perfectly, so no additional effort is needed to establish the point correspondences between two images. In the case when there are corner points missing in the detection, we use the following method to match two images. We perform a point matching procedure (see Algorithm 1) to align the anchor points (see red markers in Figure A1) detected by the checkerboard detection tool. The data matching procedure selects the set of matched anchor points used for constructing TPS transformation.

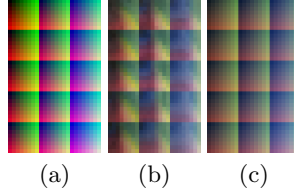
### B Color transformation

As shown in Figure A2, we generate the training dataset to map a digital color palette to the same one printed on a T-shirt. With the aid of 960 color cell

**Algorithm 1** Constructing TPS transformation

- 
- 1: **Input:** Given original image  $\mathbf{x}_1$  (frame 1) with  $r_1 \times c_1$  anchor points, each of which has coordinate  $\mathbf{p}^{(1)}[i, j]$ , where  $i \in [r_1]$ ,  $j \in [c_1]$  and  $[n]$  denotes the integer set  $\{1, 2, \dots, n\}$ , target image  $\mathbf{x}_2$  (frame 2) with  $r_2 \times c_2$  anchor points, each of which has coordinate  $\mathbf{p}^{(2)}[i, j]$ , where  $i \in [r_2]$  and  $j \in [c_2]$ , distance tolerance  $\epsilon > 0$ , and empty vectors  $\tilde{\mathbf{p}}^{(1)}$  and  $\tilde{\mathbf{p}}^{(2)}$ .
  - 2: **Output:** Matched  $r \times c$  anchor points  $\tilde{\mathbf{p}}^{(1)}[i, j]$  versus  $\tilde{\mathbf{p}}^{(2)}[i, j]$  for  $i \in [r]$  and  $j \in [c]$ , and TPS transformation  $t_{\text{TPS}}$  from  $\mathbf{x}_1$  to  $\mathbf{x}_2$ .
  - 3: **for**  $(i, j) \in [r_1] \times [c_1]$  **do**
  - 4:   given  $\mathbf{p}^{(1)}[i, j]$  in  $\mathbf{x}_1$ , find the candidate of matching point  $\mathbf{p}^{(2)}[i', j']$  by nearest neighbor in  $\mathbf{x}_2$ ,
  - 5:   **if**  $\|\mathbf{p}^{(1)}[i, j] - \mathbf{p}^{(2)}[i', j']\|_2 \leq \epsilon$  **then**
  - 6:     matching  $\mathbf{p}^{(1)}[i, j]$  with  $\mathbf{p}^{(2)}[i', j']$ , and adding them into  $\tilde{\mathbf{p}}^{(1)}$  and  $\tilde{\mathbf{p}}^{(2)}$  respectively,
  - 7:   **end if**
  - 8: **end for**
  - 9: build TPS transformation  $t_{\text{TPS}}$  by solving Eq. (2) given  $\tilde{\mathbf{p}}^{(1)}$  and  $\tilde{\mathbf{p}}^{(2)}$ .
- 

pairs. We learn the weights of the quadratic polynomial regression by minimizing the mean squared error of the predicted physical color (with the digital color in Figure A2(a) as input) and the ground-truth physical color provided in Figure A2(b). Once the color transformer  $t_{\text{color}}$  is learnt, we then incorporate it into (5).



**Fig. A2:** Physical color transformation. (a): The digital color map (b): The printed color map on a T-shirt (captured by the camera of iPhone X). (c): The predicted transformation from (a) via the learnt polynomial regression.

## C Formulation of attack loss

There are two possible options to formulate the attack loss  $f$  to fool person detectors. First,  $f$  is specified as the misclassification loss, commonly-used in most of previous works. The goal is to misclassify the class ‘person’ to any other incorrect class. For YOLOv2, we minimize the confidence score of all bounding boxes corresponding to the class ‘person’. For Faster R-CNN, we minimize

the classification scores of all bounding boxes labeled as ‘person’. Let  $\mathbf{x}'_i$  be a perturbed video frame, the attack loss in (6) is then given by

$$f(\mathbf{x}'_i) = \max_j \{ \max\{p_j(\mathbf{x}'_i), \nu\} \cdot \mathbb{1}_{|B_j \cap M_{p,i}| > \eta} \}, \quad (8)$$

where  $p_j(\mathbf{x}'_i)$  denotes the confidence score of the  $j$ th bounding box for YOLOv2 or the probability of the ‘person’ class at the  $j$ th bounding box for Faster R-CNN,  $\nu$  is a confidence threshold, the use of  $\max\{p_j(\mathbf{x}'_i), \nu\}$  enforces the optimizer to minimize the bounding boxes of high probability (greater than  $\nu$ ),  $B_j$  is the  $j$ th bounding box,  $M_{p,i}$  is the known bounding box encoding the person’s region, the quantity  $|B_j \cap M_{p,i}|$  represents the intersection between  $B_j$  and  $M_{p,i}$ ,  $|\cdot|$  is the cardinality function, and  $\mathbb{1}_{|B_j \cap M_{p,i}| > \eta}$  is the indicator function, which returns 1 if  $B_j$  has at least  $\eta$ -overlapping with  $M_{p,i}$ , and 0 otherwise. In Eq.(8), the quantity  $\max\{p_j(\mathbf{x}'_i), \nu\} \cdot \mathbb{1}_{|B_j \cap M_{p,i}| > \eta}$  characterizes the bounding box of our interest with both high probability and large overlapping with  $M_{p,i}$ . And the eventual loss in Eq.(8) gives the largest probability for detecting a bounding box of the object ‘person’.

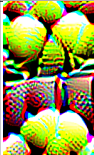

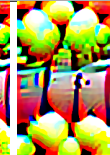
## D Hyperparameter setting

When solving Eq. (6), we use Adam optimizer [20] to train 5,000 epochs with the initial learning rate,  $1 \times 10^{-2}$ . The rate is decayed when the loss ceases to decrease. The regularization parameter  $\lambda$  for total-variation norm is set as 3. In Eq. (7), we set  $\gamma$  as 1, and solve the min-max problem by 6000 epochs with the initial learning rate  $1 \times 10^{-2}$ . In Eq. (5), the details of transformations  $t$  are shown in Table A1.

Transformation	Minimum	Maximum
Scale	0.5	2
Brightness	-0.1	0.1
Contrast	0.8	1.2
Random uniform noise	-0.1	0.1
Blurring	average pooling/filter size = 5	

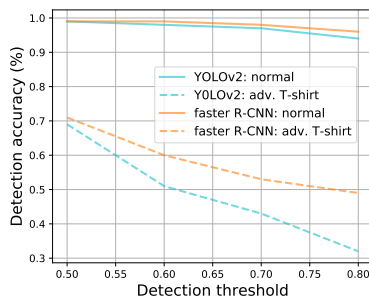
**Table A1:** The conventional transformations  $t$  in Eq. (5).

In experiments, we find that the hyperparameter  $\lambda$  strikes a balance between the fine-gained perturbation pattern and its smoothness. As we can see in Figure A3, when  $\lambda$  is smallest (namely,  $\lambda = 1$ ), the perturbation can achieve the best ASR (82% ) against YOLOv2 in the digital space, however when we test the digital pattern in the physical world, the attacking performance drops to 51% (worse than the case of  $\lambda = 3$ ) as the non-smooth (sharp) perturbation pattern might not be well captured by a real-world camera. In our experiments, we choose  $\lambda = 3$  for the best tradeoff between digital and physical results.

$\lambda$	1	3	5
			
digital	82%	74%	69%
physical	51%	57%	55%

**Fig. A3:** ASR v.s.  $\lambda$  against YOLOv2.

For a real-world deployment of a person detector, the minimum detection threshold needs to be empirically determined to obtain a good tradeoff between detection accuracy and false alarm rates. In our physical-word testing, we set the threshold to 0.7 for Faster R-CNN and YOLOv2, at which both of them achieve detection accuracy over 97% on person wearing normal clothing. The sensitivity analysis of this threshold is provided in Figure A4.



**Fig. A4:** The detection accuracy of YOLOV2 and Faster R-CNN under different detection thresholds. ‘Normal’ means the case of persons wearing normal clothing, and ‘adv. T-shirt’ means the case of persons wearing the adversarial T-shirt.

## E Dataset details

In Table A2, we summarize dataset we used in Section 4.2 and 4.3.

In Section 4.4 for ablation study on parameter sensitivity and generalization to more complex testing scenarios, we further collected some new test data. Specifically, we considered the scenario of five people (two females and three males) for ablation study and none of them appeared in the original training



**Table A2:** Summary of our collected dataset in each scenes. The values in the table are presented by number of videos (total number of frames) in each scene, ie, 4 (177) means 4 videos and 177 frames in total.

videos (frames)	indoor			outdoor		overall
	office	elevator	hallway	street1	street2	
single-person	4 (177)	4 (135)	4 (230)	4 (225)	4 (240)	20 (1007)
multi-persons	4 (162)	4 (132)	4 (245)	4 (230)	4 (227)	20 (996)
train	6 (245)	6 (180)	6 (335)	6 (344)	6 (365)	30 (1469)
test (digital)	2 (94)	2 (87)	2 (140)	2 (111)	2 (102)	10 (534)
	unseen	elevator	hallway	street3		
test (physical)	6 (236)	6 (184)	6 (220)	6 (288)		24 (928)

and testing datasets. We recorded multiple videos by using two cameras (one iPhone X and one iPhone XI) and reported the resulting ASR in average.

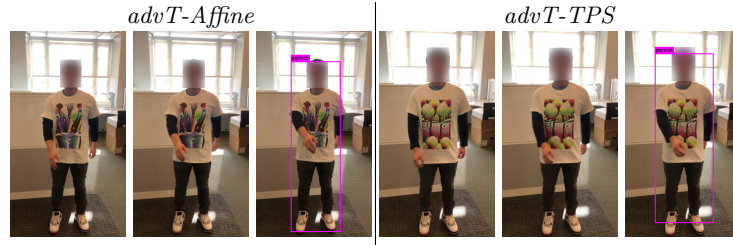
## F More experimental results

In Figure A5, we demonstrate our physical-world attack results in two scenarios: a) adversarial T-shirts generated by *advT-TPS*, *advT-Affine* and *advPatch* in an outdoor scenario (the first three rows), b) adversarial T-shirts generated by *advT-TPS* and *advT-Affine* in an unseen scenario (at a location never seen in the training dataset). As we can see, our method outperforms affine and baseline. In the absence of TPS, adversarial T-shirts generated by affine and baseline fail in most of cases, implying the importance of TPS to model the T-shirt deformation. When a person whom wears the adversarial T-shirt walks towards the camera, as expected, the detector also becomes easier to be attacked.

Moreover, it is worth noting that some postures remain challenging as the larger occlusion is the worse ASR is. To delve into this problem, Fig. A6 presents how well our adversarial T-shirt can handle occlusion by partially covering the T-shirt by hand. Not surprisingly, both *advT-Affine* and *advT-TPS* may fail when occlusion becomes quite large. Thus, occlusion is still an interesting problem for physical adversaries.



**Fig. A5:** Some testing frames in the physical world using adversarial T-shirt against YOLOv2. All frames are performed by two persons with one wearing the proposed adversarial T-shirt, generated by our method (*advT-TPS*), *advT-Affine* and *advPatch*. The first three rows: an unseen outdoor scenes. The last two rows: an unseen indoor scenes.



**Fig. A6:** When *advT-Affine* and *advT-TPS* happen occlusion by hand.