Imperial College London

APSS Seminar 10

AdvIT: Adversarial Frames Identifier Based on Temporal Consistency In Videos

(Charles) Chengzeng You
Connected and autonomous vehicles
Department of computing
Email: chengzeng.you19@imperial.ac.uk

Content

1. Introduction
2. Literature Review
3. Adversarial frame identifier via temporal consistency
4. Experiments
5. Disccussion and conclusion

1.1 Introduction

DNNs on videos is a particularly interesting and important domain, as attacks against many applications have the potential to cause serious physical and financial damage.

1.1 Introduction

Most of the detection methods are defeated by adaptive attacks [4, 8, 22, 23].

Leveraging special properties of data source (e.g. videos) and enhancing model robustness is possible.

AdvIT : the first adversarial frame identifier for videos based on temporal consistency.

1.2 Contributions

(1) leverage the temporal consistency in videos and and randomness to develop an efficient and effective approach AdvIT that detects adversarial frames with above 95% detection rate.
(2) conduct extensive experiments and analyses to identify adversarial frames within videos against state-of- the-art attacks on video learning tasks showing that AdvIT outperform potential baselines significantly.
(3) propose strong adaptive attacks against the detection method and show that it is robust against the proposed attacks which assume adversaries are aware of the detection mechanism.
(4) evaluate the transferability among different optical flow estimators and show that adversarial attacks rarely transfer among them.

Content

1. Introduction
2. Literature Review
3. Adversarial frame identifier via temporal consistency
4. Experiments
5. Disccussion and conclusion

2.1 Learning for videos

- Fully convolution networks [21] propose an end-to-end model that first down-samples the feature map and then up-sample to generate a pixel-wise class score map for **semantic segmentation**.
- **Object detection** has been accomplished using R-CNN models that adopt a proposal and prediction pipeline [16, 31] for object detection and semantic segmentation.
- Stacked Hour- glass Networks [25] achieve state of the art performance on the task of single person **humans pose estimation** through using a repeated top-down and bottom-up model and cap- turing information at all scales.

2.2 Adversarial attacks

- **DAG** [44] proposes an iterative gradient based attack methods to attack all pixels until most of the pixels have been identified as target classes for semantic segmentation while it attacks all proposed bounding boxes until they are misclassified as target classes.
- **Houdini** [10] proposes a optimization based attack algorithm by introducing a surrogate loss function.
- **Sparse adversarial perturbation** [38] demonstrates a method to generate universal adversarial perturbations against action recognition model for videos.

2.3 Defenses against adversarial examples

- **Adversarial training** [17] and its variations [23, 36] have generally been more successful, but usually come at the cost of accuracy and increased training time [37].
- Athalye et al. [4] successfully generated adversarial examples in the presence of detection and defense strategies.
- Xiao et al. [39] proposed a method to detect adversarial examples on semantic segmentation using spatial information.

Content

1. Introduction

2. Literature Review

3. Adversarial frame identifier via temporal consistency

4. Experiments

5. Disccussion and conclusion

3.1 Define the problem

The sequence of image frames: $X1, . . . , Xt$
given a learner:

$$\mathbf{g(Xt) = Yt}$$

add small perturbation: $ǫi$ (i.e., $Xi \leftarrow Xi + ǫi$)
so that,

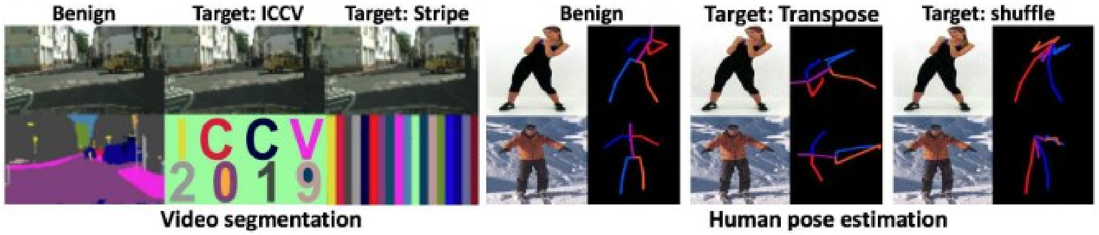
$$\mathbf{g(Xt) = Y\ast}$$

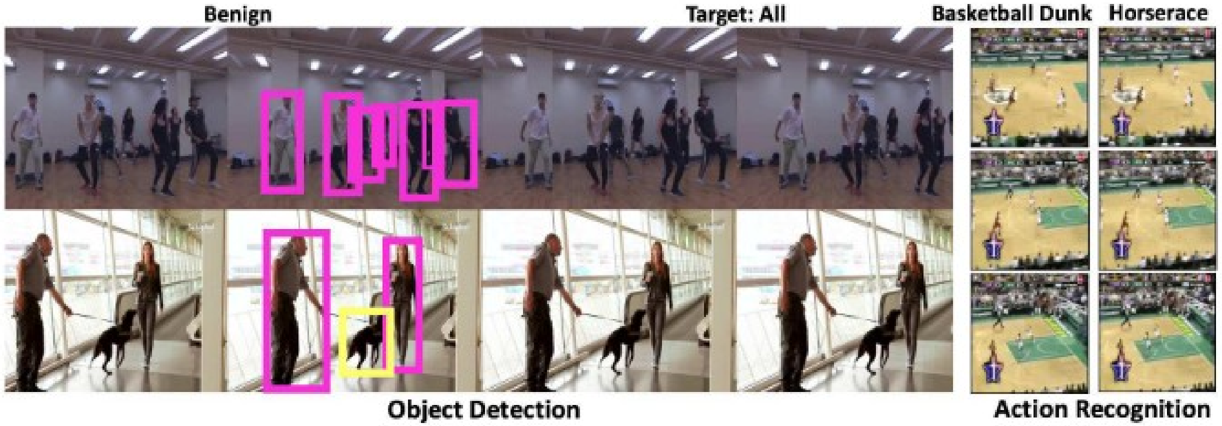
where $Y\ast$ is the adversarial target depending on the learning task.

The goal is to determine whether the target frame Xt is adversarial.

3.2 Threat model

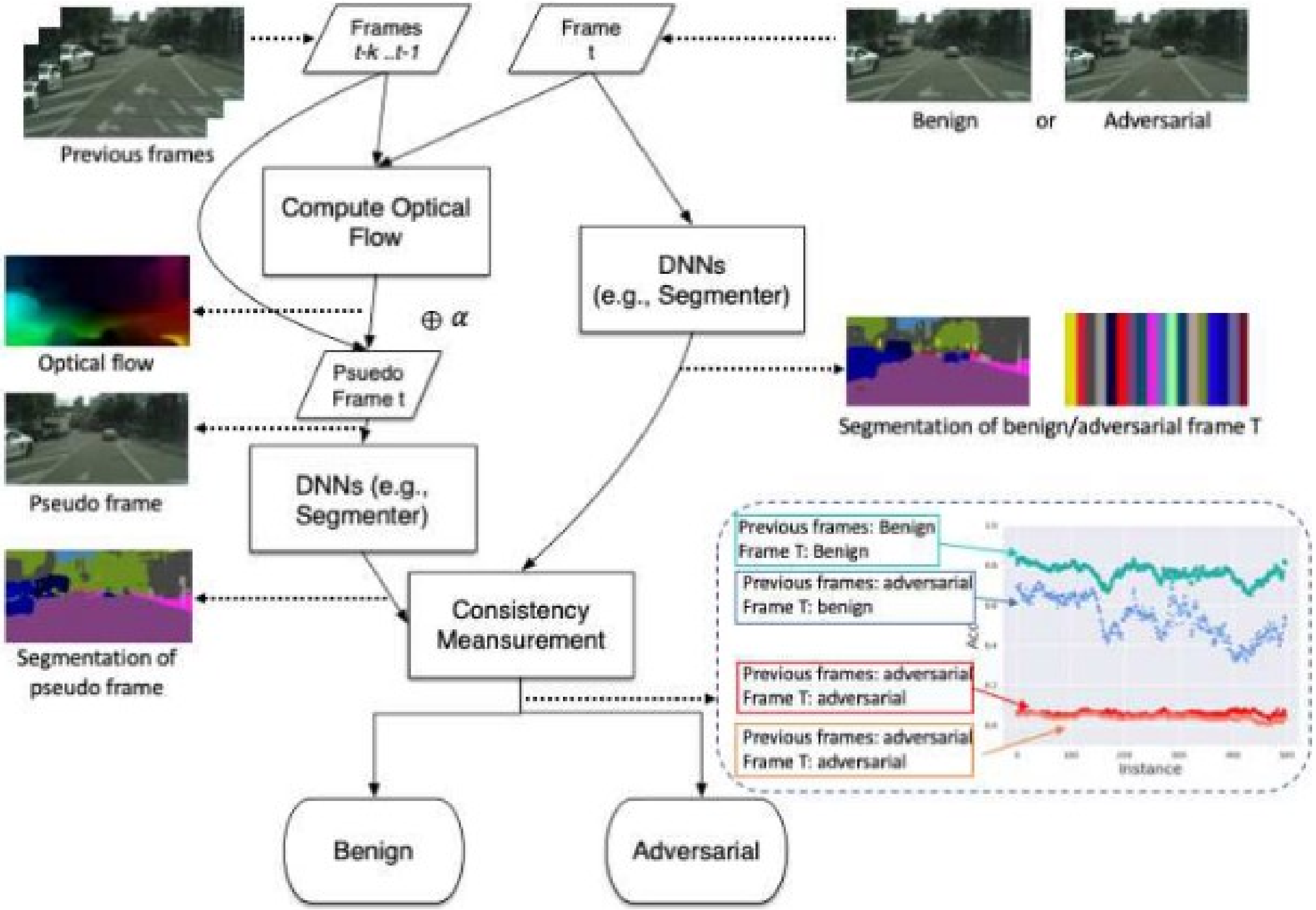
Independent frame attack includes Houdini [10] and DAG [44] attacks for video segmentation, object detection and human pose estimation tasks; **Temporal continuity attack** includes Sparse attack [38] on action recognition and universal perturbation [24] on the previous three tasks.





Imperial College London

3.3 Overview of method



3.4 Pseudo Frame Generation

An optical flow between the two frames is a vector field $OF = (\Delta u,\Delta v)$ that describes the dis- placement of pixels between the frames and we denote an image generated by applying flow as $X\hat{}s\rightarrow t$.

We obtain $X\hat{}s\rightarrow t$ by sampling pixel intensities from Xs; the pixel in $X\hat{}s\rightarrow t$ at location (i, j) corresponds to the pixel at location $(u, v) = (i + \Delta u(i, j), j + \Delta v(i, j))$ in image Xs.

3.4 Pseudo Frame Generation

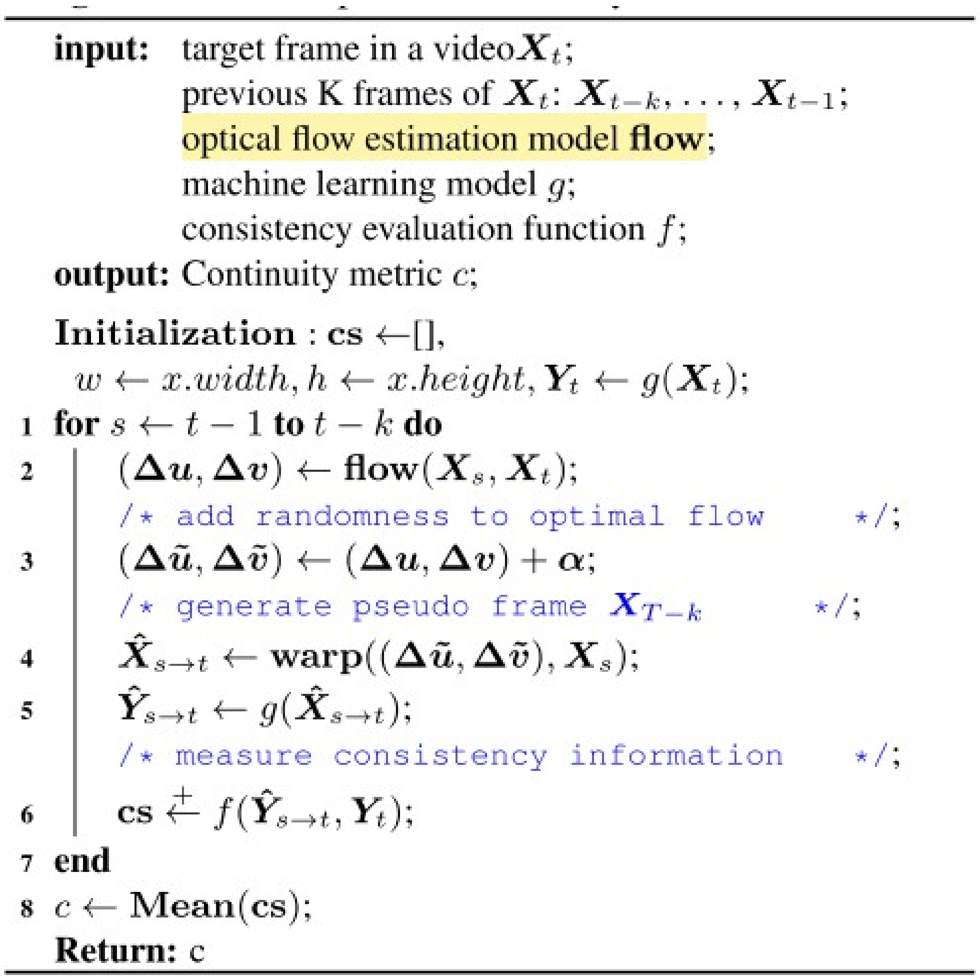
bilinear sampling

$$\hat{\boldsymbol{X}}_{s\to t}(i,j) = \sum_{(i',j')\in N(u,v)} \boldsymbol{X}_s(i',j')(1-|u-i'|)(1-|v-j'|) \quad (1)$$

where N(u, v) stands for the indices of the 4-pixel neighbors at location (u, v)
X·(i, j) represents the pixel value at location (i, j).

To further combat the creation of adversarial perturbation, we add randomness α ∼ N(0, σ2) to the flow field (Δu,Δv) to generate the pseudo frames.

3.5 Temporal Consistency Based Test



```
input:  target frame in a video$\boldsymbol{X}_t$;
        previous K frames of $\boldsymbol{X}_t$: $\boldsymbol{X}_{t-k}, \ldots, \boldsymbol{X}_{t-1}$;
        optical flow estimation model flow;
        machine learning model $g$;
        consistency evaluation function $f$;
output: Continuity metric $c$;
Initialization : cs ←[],
  $w \leftarrow x.width, h \leftarrow x.height, \boldsymbol{Y}_t \leftarrow g(\boldsymbol{X}_t)$;
for $s \leftarrow t-1$ to $t-k$ do
    $(\boldsymbol{\Delta u}, \boldsymbol{\Delta v}) \leftarrow$ flow$(\boldsymbol{X}_s, \boldsymbol{X}_t)$;
    /* add randomness to optimal flow     */;
    $(\boldsymbol{\Delta \tilde{u}}, \boldsymbol{\Delta \tilde{v}}) \leftarrow (\boldsymbol{\Delta u}, \boldsymbol{\Delta v}) + \boldsymbol{\alpha}$;
    /* generate pseudo frame $\boldsymbol{X}_{T-k}$     */;
    $\hat{\boldsymbol{X}}_{s\to t} \leftarrow$ warp$((\boldsymbol{\Delta \tilde{u}}, \boldsymbol{\Delta \tilde{v}}), \boldsymbol{X}_s)$;
    $\hat{\boldsymbol{Y}}_{s\to t} \leftarrow g(\hat{\boldsymbol{X}}_{s\to t})$;
    /* measure consistency information     */;
    cs $\overset{+}{\leftarrow} f(\hat{\boldsymbol{Y}}_{s\to t}, \boldsymbol{Y}_t)$;
end
$c \leftarrow$ Mean(cs);
Return: c
```

3.5 Temporal Consistency Based Test

consistency measurement function f for various learning tasks:
(1) Segmentation: Pixel-wise accuracy1.
(2) Human Pose Estimation: Average L2 distance over all key joints.
(3) Object Detection: mIoU between bounding boxes of pseudo frames and the current frame.
(4) Action Recognition: the average of forward and backward KL divergence between the two categorical distributions.

Imperial College
London

Content

1. Introduction

2. Literature Review

3. Adversarial frame identifier via temporal consistency

▷ 4. Experiments

5. Disccussion and conclusion

4.1 Implementation Details

4.1.1 Semantic segmentation

- Dataset: CityScapes dataset which consists of high-resolution (1024x2048) outdoor videos captured from a moving car.
- Model : adopted Dilated Residual Network [45] model with DRN-D-22 architecture.The mean Intersection Over Union (mIoU) of this model on pristine data is 66.7.
- attack methods: Hou- dini [10] and DAG [44]
- adversarial targets: “Remapping”, “Stripe”, and “ICCV 2019”

4.1 Implementation Details

4.1.2 Human Pose Estimation

- Dataset: MPII human pose dataset [3]
- Model: Stacked Hourglass Network model [25]
- Attack methods:Houdini algorithm
- Attack targets: “Transpose” and “Shuffle”.

4.1 Implementation Details

4.1.3 Object Detection

- Dataset: DAVIS Challenge 17 dataset
- Model: YOLOv3
- Attack methods:DAG algorithm
- Attack targets: “All” and “Person”

4.1 Implementation Details

4.1.4 Action Recognition

- Dataset: UCF-101 dataset [33]
- Model: CNN+RNN model used in [38]
- Attack methods:Sparse
- Feature extraction: Inception V2 model [34]

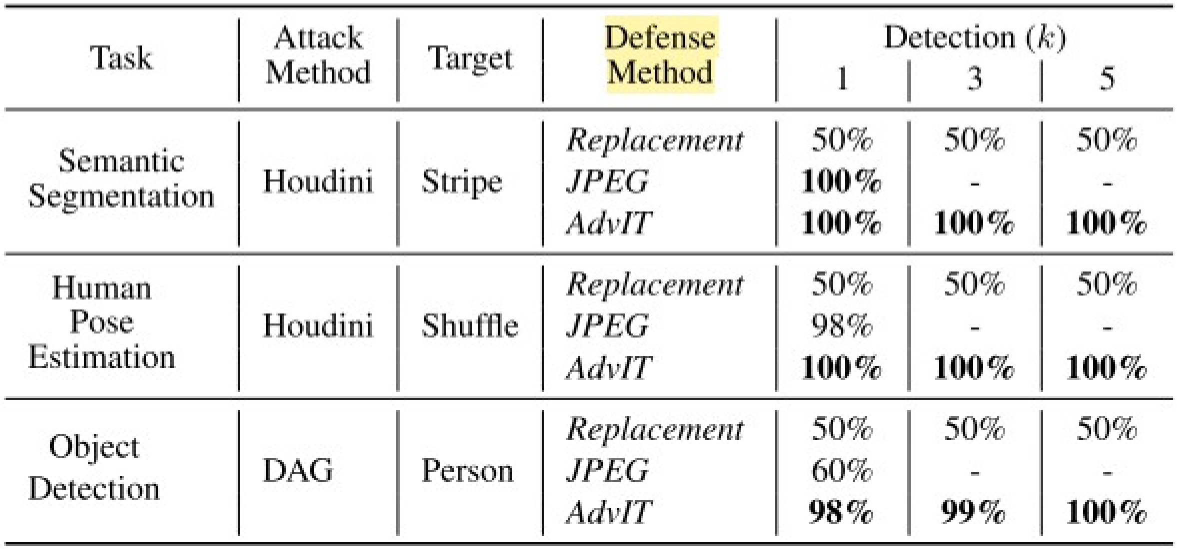
4.2 Temporal Consistency Based Detection

4.2.1 Detecting independent frame attack

- Independent frame attack includes Houdini [10] and DAG [44] on three video tasks: semantic segmentation, human pose estimate and object detection.
- test the method and report the results under various conditions: previous frames are purely benign, adversarial, or mixture.

4.2 Temporal Consistency Based Detection

4.2.1 Detecting independent frame attack

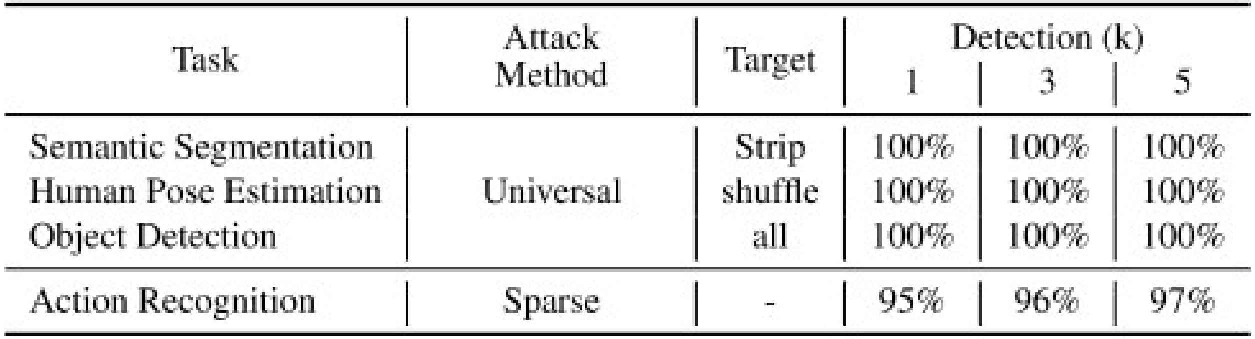


Task	Attack Method	Target	Defense Method	Detection (k)		
				1	3	5
Semantic Segmentation	Houdini	Stripe	*Replacement*	50%	50%	50%
			JPEG	**100%**	-	-
			AdvIT	**100%**	**100%**	**100%**
Human Pose Estimation	Houdini	Shuffle	*Replacement*	50%	50%	50%
			JPEG	98%	-	-
			AdvIT	**100%**	**100%**	**100%**
Object Detection	DAG	Person	*Replacement*	50%	50%	50%
			JPEG	60%	-	-
			AdvIT	**98%**	**99%**	**100%**

Table 1: Comparison of detection results (AUC) against different attacks for *AdvIT* and baseline methods.

4.2 Temporal Consistency Based Detection

4.2.2 Detecting temporal continuity attack



Task	Attack Method	Target	Detection (k)		
			1	3	5
Semantic Segmentation	Universal	Strip	100%	100%	100%
Human Pose Estimation		shuffle	100%	100%	100%
Object Detection		all	100%	100%	100%
Action Recognition	Sparse	-	95%	96%	97%

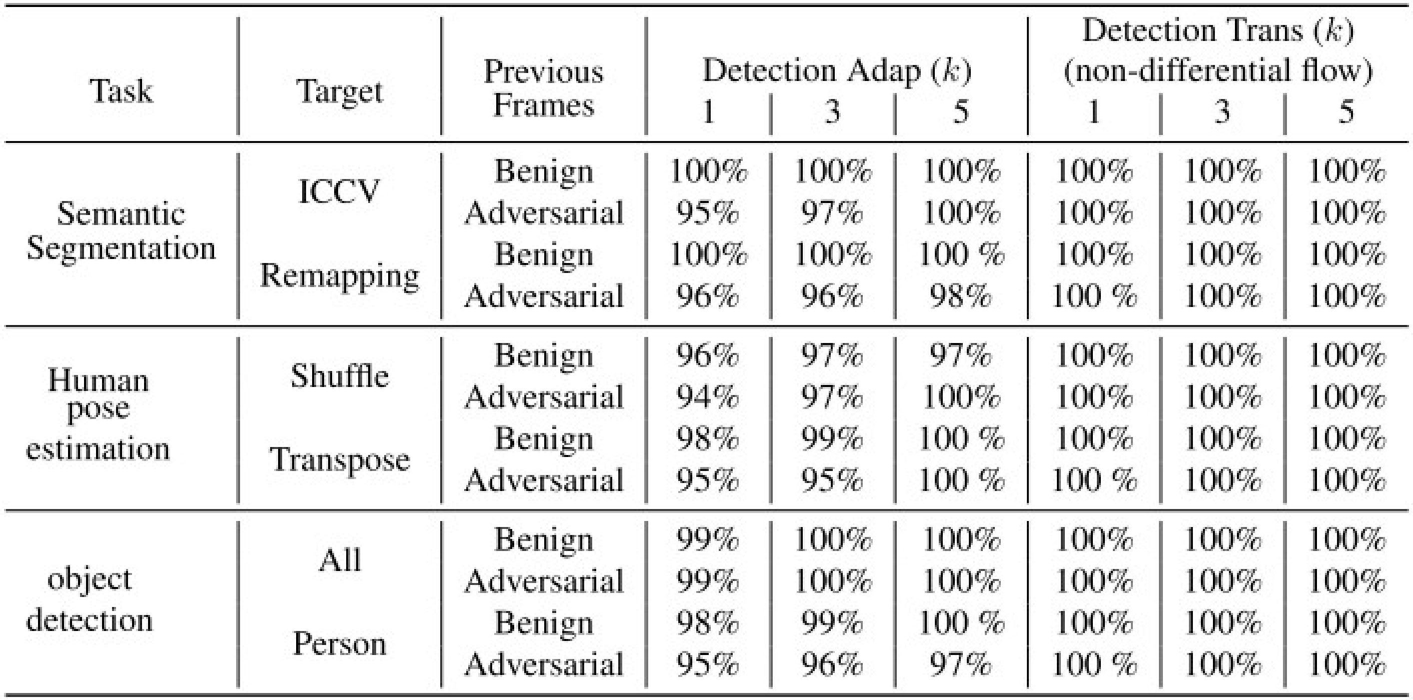
Table 2: Detection results (AUC) against *temporal continuity attack*

4.3 Analysis of Adaptive Attacks

- the attacker generates a perturbation that considers both the current and generated pseudo frames.
- allow the attacker to use the state of the art adaptive attack estimation method expectation of transformation to approximate potential randomness.
- Follow the setting in [4], and randomly select 30 possible α in each iteration to optimize the perturbation.

Imperial College London

4.3 Analysis of Adaptive Attacks

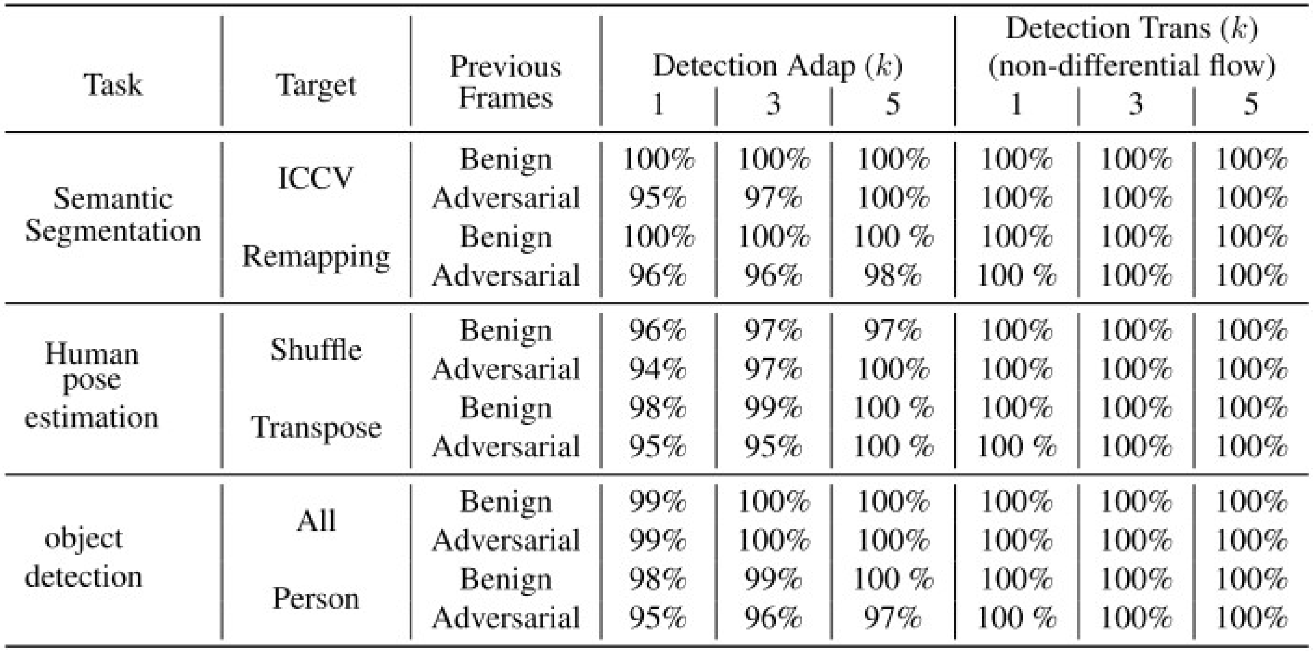


Task	Target	Previous Frames	Detection Adap (k)			Detection Trans (k) (non-differential flow)		
			1	3	5	1	3	5
Semantic Segmentation	ICCV	Benign	100%	100%	100%	100%	100%	100%
		Adversarial	95%	97%	100%	100%	100%	100%
	Remapping	Benign	100%	100%	100 %	100%	100%	100%
		Adversarial	96%	96%	98%	100 %	100%	100%
Human pose estimation	Shuffle	Benign	96%	97%	97%	100%	100%	100%
		Adversarial	94%	97%	100%	100%	100%	100%
	Transpose	Benign	98%	99%	100 %	100%	100%	100%
		Adversarial	95%	95%	100 %	100 %	100%	100%
object detection	All	Benign	99%	100%	100%	100%	100%	100%
		Adversarial	99%	100%	100%	100%	100%	100%
	Person	Benign	98%	99%	100 %	100%	100%	100%
		Adversarial	95%	96%	97%	100 %	100%	100%

Table 3: Detection results (AUC) of adaptive attacks and transferability analysis.

4.3 Analysis of Adaptive Attacks

4.3.1 Transferbility analysis

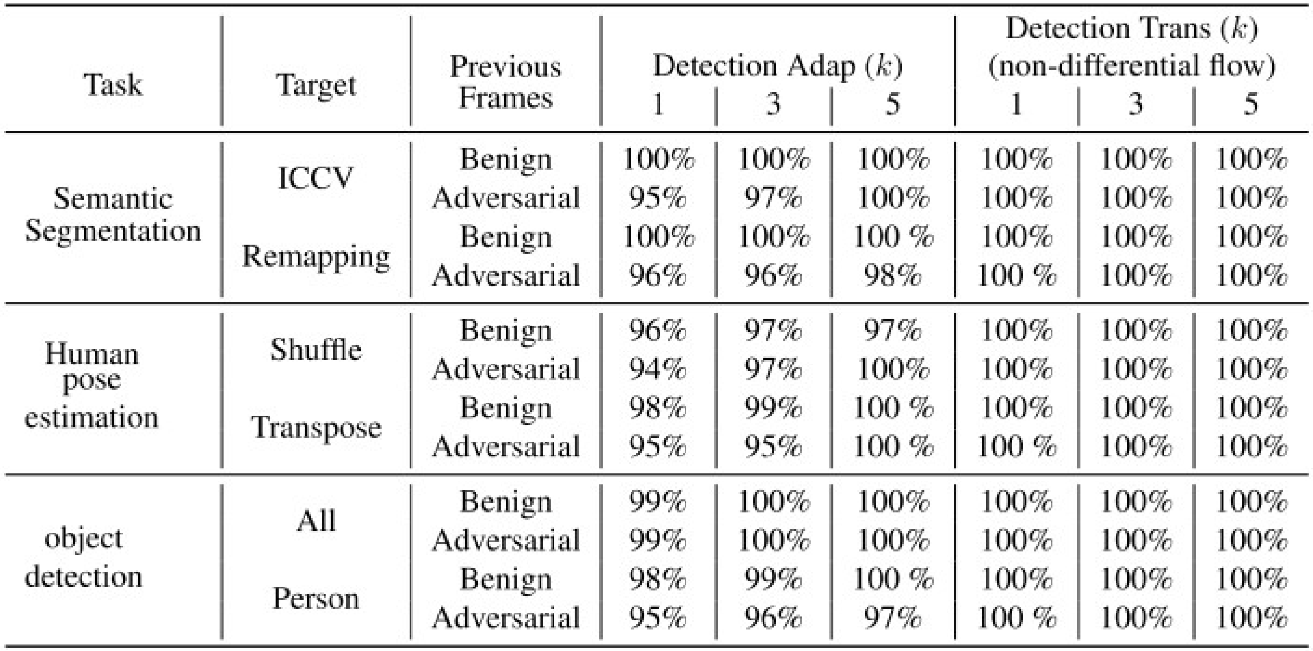


Task	Target	Previous Frames	Detection Adap (k)			Detection Trans (k) (non-differential flow)		
			1	3	5	1	3	5
Semantic Segmentation	ICCV	Benign	100%	100%	100%	100%	100%	100%
		Adversarial	95%	97%	100%	100%	100%	100%
	Remapping	Benign	100%	100%	100 %	100%	100%	100%
		Adversarial	96%	96%	98%	100 %	100%	100%
Human pose estimation	Shuffle	Benign	96%	97%	97%	100%	100%	100%
		Adversarial	94%	97%	100%	100%	100%	100%
	Transpose	Benign	98%	99%	100 %	100%	100%	100%
		Adversarial	95%	95%	100 %	100 %	100%	100%
object detection	All	Benign	99%	100%	100%	100%	100%	100%
		Adversarial	99%	100%	100%	100%	100%	100%
	Person	Benign	98%	99%	100 %	100%	100%	100%
		Adversarial	95%	96%	97%	100 %	100%	100%

Table 3: Detection results (AUC) of adaptive attacks and transferability analysis.

4.3 Analysis of Adaptive Attacks

4.3.2 Optical Flow Estimator

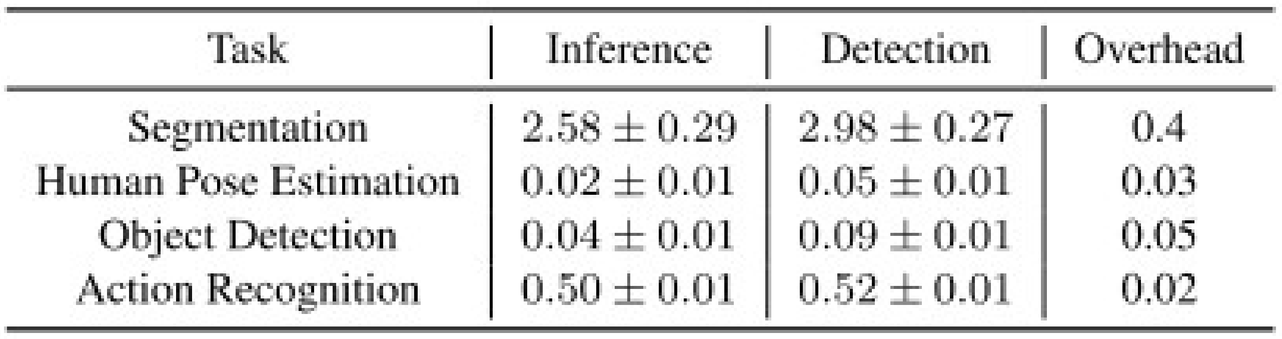


Task	Target	Previous Frames	Detection Adap (k)			Detection Trans (k) (non-differential flow)		
			1	3	5	1	3	5
Semantic Segmentation	ICCV	Benign	100%	100%	100%	100%	100%	100%
		Adversarial	95%	97%	100%	100%	100%	100%
	Remapping	Benign	100%	100%	100 %	100%	100%	100%
		Adversarial	96%	96%	98%	100 %	100%	100%
Human pose estimation	Shuffle	Benign	96%	97%	97%	100%	100%	100%
		Adversarial	94%	97%	100%	100%	100%	100%
	Transpose	Benign	98%	99%	100 %	100%	100%	100%
		Adversarial	95%	95%	100 %	100 %	100%	100%
object detection	All	Benign	99%	100%	100%	100%	100%	100%
		Adversarial	99%	100%	100%	100%	100%	100%
	Person	Benign	98%	99%	100 %	100%	100%	100%
		Adversarial	95%	96%	97%	100 %	100%	100%

Table 3: Detection results (AUC) of adaptive attacks and transferability analysis.

4.3 Analysis of Adaptive Attacks

4.3.3 Run-time Analysis



Task	Inference	Detection	Overhead
Segmentation	2.58 ± 0.29	2.98 ± 0.27	0.4
Human Pose Estimation	0.02 ± 0.01	0.05 ± 0.01	0.03
Object Detection	0.04 ± 0.01	0.09 ± 0.01	0.05
Action Recognition	0.50 ± 0.01	0.52 ± 0.01	0.02

Table 4: Detection overhead of *AdvIT* (in seconds).

Imperial College
London

Content

1. Introduction

2. Literature Review

3. Adversarial frame identifier via temporal consistency

4. Experiments

5. Disccussion and conclusion

4.3 Analysis of Adaptive Attacks

- Our experiments rely on video tasks where video is taken
- from continuous sequence. An video that contains “jump cuts" would likely introduce a small number of false posi- tive frames into our detector, as these cuts would be unpredictable by the optical flow algorithm.
- not aimed at remediating or repairing them. Future work could include using pseudo frames as surrogates for suspicious frames or using pseudo frames along with detected adversarial frames to reform attacks
-

Thank you