WEIGHTED AVERAGE PRECISION: ADVERSARIAL EXAMPLE DETECTION FOR VISUAL PERCEPTION OF AUTONOMOUS VEHICLES

Weiheng Chai, Yantao Lu, Senem Velipasalar

Electrical Engineering and Computer Science Dept., Syracuse University, NY, USA
{wchai01, ylu25, svelipas}@syr.edu

ABSTRACT

Recent works have shown that neural networks are vulnerable to carefully crafted adversarial examples (AE). By adding small perturbations to original images, AEs are able to deceive victim models, and result in incorrect outputs. Research work in adversarial machine learning started to focus on the detection of AEs in autonomous driving applications. However, existing studies either use simplifying assumptions on the outputs of object detectors or ignore the tracking system in the perception pipeline. In this paper, we first propose a novel similarity distance metric for object detection outputs in autonomous driving applications. Then, we bridge the gap between the current AE detection research and the real-world autonomous systems by providing a temporal AE detection algorithm, which takes the impact of tracking system into consideration. We perform evaluations on Berkeley Deep Drive and CityScapes datasets, by using different white-box and black-box attacks, which show that our approach outperforms the mean-average-precision and mean intersection-over-union based AE detection baselines by significantly increasing the detection accuracy.

Index Terms— Adversarial Attack, Neural Networks

1. INTRODUCTION

Significant progress in machine learning (ML) techniques, such as Deep Neural Networks (DNNs), has enabled the development of safety-critical ML systems like autonomous vehicles. Neural network-based object detection models are widely employed as an important part of autonomous driving perception systems. Since the control successors are highly dependent on the outputs of object detectors, the reliability of object detection is very important for safe autonomous driving. However, neural network-based object detectors have been shown to be vulnerable to adversarial examples (AEs) [1, 2, 3, 4],which are designed to deceive the models. To address this problem, researchers have been focusing on defending against adversarial attacks.

The information, data, or work presented herein was funded in part by National Science Foundation (NSF) under Grant 1739748, Grant 1816732 and by the Advanced Research Projects Agency-Energy (ARPA-E), U.S. Department of Energy, under Award Number DE-AR0000940. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

Defense mechanisms aim to strengthen the DNN models, and are classified into two broad categories [5]: adversarial training and gradient masking. Kurakin et al. [6] introduced the idea of adversarial training, which tries to integrate existing AE generation methods into the training process. However, this approach requires prior knowledge of possible attacks, and the robustness of adversarially trained models usually overfits to the choice of norms [1]. The idea behind gradient masking is to enhance the training process by training the model with small gradients so that it is not sensitive to small changes in the input [7]. However, Papernot et al. [5] concluded that controlling gradient information in training has limited effects in defending against adversarial attacks.

Since attack approaches are more advanced compared to defense techniques, some researchers have focused on 'detecting' AEs, instead of hardening the model itself. One of the successful AE detection methods is perturbation-based detection [8], which hypothesizes that, the robustness of DNNs to local changes (e.g., squeezing, scale etc.) does not generalize to the perturbations added by AEs. Therefore, researchers add perturbations to an input image, then do inferences for both the original and perturbed images. If the outputs are different, the input image is concluded to be an AE.

Although existing AE detection methods provide promising results for image classification, much less attention has been paid to object detection. One significant challenge is the design/use of the right similarity metric for the object detector outputs. Unlike image classification, for which the output is a 1-D vector representing probabilities of each class, the output of object detection contains bounding box location, label index and confidence scores. To handle this, some prior work assumes that there is only one bounding box in the image [9]. However, in real world, an image often contains multiple objects of interest. Thus, the aforementioned assumption cannot be made, especially in autonomous driving applications. Mean average precision (mAP) is the most commonly used metric to evaluate the performance of object detectors, such as in the Open Images 2019 challenge [10]. Some existing works directly employ mAP to calculate the distance between the detection outputs of two images. However, when average precision is used, all bounding boxes are treated the same. Also, when a perturbation-based AE detection method is used, adding perturbations to benign images

will cause differences in the output boxes as well, and this will mostly affect the smaller objects. Thus, the mAP-based metric is not suitable for perturbation-based AE detection.

In this paper, we present a unified framework to bridge the gap between AE research and real-world autonomous driving systems. Our framework extends the AE detection mechanisms developed for image classifiers into the object detection domain by designing a novel similarity metric to detect inconsistencies between dense object detection results of two video frames, and by adding temporal information into the detection pipeline to further improve accuracy. We first evaluate the impact of AEs on end-to-end visual perceptual autonomous driving systems, and then propose a defense leveraging the evaluation results to effectively prevent the threat from AEs in self-driving tasks. Our framework allows different auxiliary transformations, which have been demonstrated effective in unveiling AEs in image classification and face recognition, to be easily transferred to the object detection domain without sacrificing effectiveness. To evaluate our approach, we perform adversarial attacks against state-of-the-art object detectors by using the most effective attacks in the image classification domain. We conduct a large scale evaluation on diverse adversarial targets on different datasets, namely Cityscapes [11] and Berkeley Deep Drive (BDD) [12]. The results show high AE detection accuracy with only 2.86% false positive (FP) and 3.16% false negative (FN) rates. The code is available at: https://github.com/anony4papers/wAP.

2. REEVALUATION OF ADVERSARIAL THREAT

Autonomous driving systems employ tracking algorithms to post-process the object detection results before using them to produce control actions. A tracking algorithm creates a unique ID for each detected object, and tracks them in a video. An ideal tracking algorithm can handle cases of tracked objects 'disappearing', pick back up objects it has 'lost' in between frames, and is robust to merges and occlusions.

Kalman filter [13], also known as linear quadratic estimation (LQE), is one of the most commonly used object tracking algorithms, and employed by the popular open-source self-driving platforms including Apollo [14] and Autoware [15]. The Kalman filter algorithm is recursive, and can run in real-time, using only the present input measurements and previously calculated state and its uncertainty matrix.

Recent works [16] study adversarial ML attacks by considering the complete visual perception pipeline in autonomous driving, and taking both object detection and tracking into consideration. It was shown that [16], due to the existence of Kalman filter-based tracking system, which has been proven effective in tolerating missed and occasionally inaccurate object detection results [17], only temporal adversarial attacks, which can successfully attack several consecutive frames, can fool the perception pipeline in autonomous driving. However, current physical adversarial attack approaches are not effective enough to generate high success rate AEs in consecutive frames. Our threat reevaluation results suggest that although object trackers like Kalman filter raise the bar for adversarial attacks, strong and robust AEs can still fool the detectors and cause wrong perception results.

3. PROPOSED METHOD

The overview of our proposed AE detection framework is shown in Fig. 1. First, for each frame x_t from time t, a transformed image x'_t is generated using an auxiliary transformation $T(x_t)$. This is based on a hypothesis that the robustness of DNNs to local changes (e.g., squeezing, scale, position) does not generalize to the perturbations added by AEs, which has been validated by previous works [18, 19, 20]. Thus, if x is an AE, it is highly likely that the object detection result $f(x)$ will be very different from that of $f(x')$.

However, there is no suitable metric to describe/quantify the similarity of two object detection results. Existing AE detection methods, designed for image classification tasks, simply compare the predicted classes and their confidences. Yet, in object detection, we need to deal with dense and sometimes overlapping bounding boxes, and it is non-trivial to design a metric, which provides a good balance between precision and recall. As shown below, the number of false positives can be prohibitively high when using traditional mAP as the distance metric, since it assigns equal weights to all the detected objects regardless of their sizes and classes. Also, calculating the mAP between two frames instead of a set of test images provides no statistical significance.

Thus, we propose a novel distance metric, referred to as the weighted average precision (wAP) and detailed in §3.1, to describe and quantify the differences between two object detection results. In addition, by monitoring the variance of a temporal inconsistency metric $I(D_t)$, AEs can be detected in real-time, and mitigation actions, such as requiring human control, or rolling back to the predictions of Kalman filter, can be taken on time. Intersection over Union (IoU) measures the overlap between the predicted and the ground truth bounding boxes. Mean IoU (mIoU) is another metric we used in our experiments for comparison purposes, where mIoU is the mean of all the IoU values across all classes.

3.1. Proposed Frame-wise Distance Metric

The pseudo code for the computation of our proposed frame-wise distance metric $D(x, y)$ is provided in Algorithm 1, where x and y refer to the object detection results from the original image and its transformation, respectively. Two images are run through an object detector to obtain bounding boxes ($bbox$), confidence scores (cs) and bounding box predicted classes (cl) for each image. By considering x as the ground truth, and based on the IoU value between the bounding boxes of x and y and an overlap threshold ($MIN_{overlap}$), all bounding boxes are classified into true positive (TP), false positive (FP) and false negative (FN) sets. Let tp be the set of index pairs of TP bounding boxes, fp be the set of indices

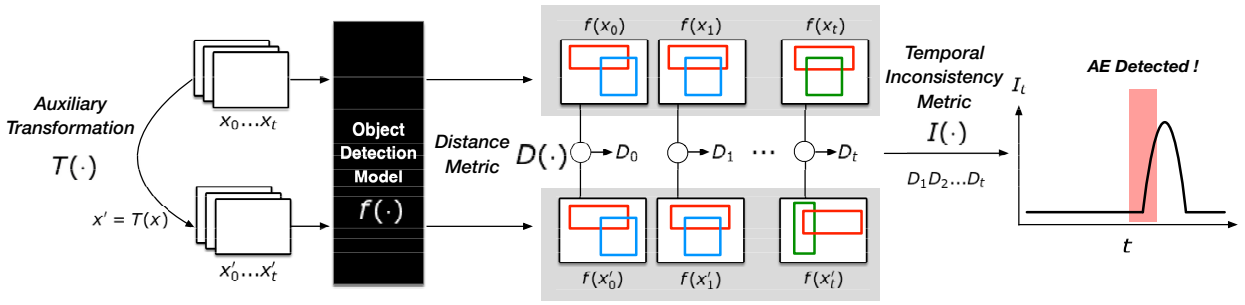

Fig. 1. The proposed framework for detecting AEs. The object detection model is used on both the original images and the images generated from an auxiliary transformation $T(\cdot)$. Distance $D(\cdot)$ is calculated between the object detection results of each pair. The distances from consecutive frames are used to obtain a temporal inconsistency metric $I(\cdot)$, and the adversarial attack can be detected on-the-fly by monitoring the variance of I over time.

Algorithm 1: Frame-wise Distance Metric $D(x, y)$

Data: x: {bounding boxes, confidence score, object class},
y: {bounding boxes, confidence score, object class}
tp = []; fn = [] ;
for $i \leftarrow 0$ **to** N_x **do**
  for $j \leftarrow 0$ **to** N_y **do**
    $IoU_{ij} \leftarrow getIoU(x_i.bbox, y_j.bbox)$;
    if $IoU_{ij} > MIN_{overlap}$ && $x_i.cl == y_j.cl$ **then**
      tp append (i, j) ;
      break;
    end
  end
  fn append i ;
end
fp = {0, 1, ..., N_y} ;
for $i \leftarrow 0$ **to** N_{tp} **do**
  del fp[tp[i,1]] ;
end
$\mathcal{D}_{tp} = \frac{\sum_{(i,j)\in tp} \mathcal{F}(\mathcal{DA}(x_i.bbox, y_j.bbox))}{A_y + A_x} + \gamma_{cs} \sum_{(i,j)\in tp} | cs_{x_i} - cs_{y_j} |$;
$\mathcal{D}_{fp} = \frac{\sum_{i\in fp} \mathcal{F}(A_{y_i})}{A_y} + \gamma_{cs} \sum_{i\in fp} cs_{y_i}$;
$\mathcal{D}_{fn} = \frac{\sum_{i\in fn} \mathcal{F}(A_{x_i})}{A_x} + \gamma_{cs} \sum_{i\in fn} cs_{x_i}$;
$\mathcal{D}_{error} = \frac{len(fp)+len(fn)}{len(fp)+len(fn)+len(tp)}$;
$D(x, y) = \frac{\alpha_{tp}\mathcal{D}_{tp} + \alpha_{fp}\mathcal{D}_{fp} + \alpha_{fn}\mathcal{D}_{fn} + \alpha_{er}\mathcal{D}_{error}}{\alpha_{tp} + \alpha_{fp} + \alpha_{fn} + \alpha_{er}}$

of FP bounding boxes of y, and fn be the set of indices of FN bounding boxes of x. The distance metric is composed of area-based and confidence score-based distance. For the bounding box pairs in the TP set, sum of the difference areas ($\mathcal{DA}(A, B) = (A - B) \cup (B - A)$) is calculated, and divided by the sum of areas of bounding boxes of x and y. For FP and FN set, sum of the bounding box areas are calculated and divided by the sum of bounding boxes areas of y and x, respectively. All the area-based values are fed into the weight function $\mathcal{F}(m) = \frac{m}{m+\alpha}$ to make small bounding boxes contribute less. For confidence scores part, we obtain difference of confidence scores of the TP pairs, confidence scores of FP samples and FN samples. All the confidence score-based values are multiplied by a constant weight parameter (γ_{cs}). The weighted sum of $\mathcal{D}_{tp}$, $\mathcal{D}_{fp}$, $\mathcal{D}_{fn}$ and $\mathcal{D}_{error}$ is obtained to get the final distance $D(x, y)$.

3.2. Temporal Consistency

In a perception system, a tracker will be deleted if it cannot be associated to an object for a duration of R frames. Thus, for AE detection, only attacks that last longer than R frames should be regarded as valid AEs. In other words, in a set of R consecutive frames, AEs that can successfully attack all R frames are regarded as an effective attack. Intermittent attacks, which do not affect consecutive frames will likely fail, since trackers save information from the benign samples, and if attacks are not able to retain sufficient duration, trackers will be calibrated back to benign objects. Motivated by this fact, we propose a temporal detection approach, which can be expressed as

$$I(D_0 \ldots D_t) = \prod_{i=0}^{R} (\mathbb{1}(D_i - \mu)), \tag{1}$$

where $\mathbb{1}(\cdot)$ is the indicator function returning 1 if $(D_i - \mu) > 0$, and 0 otherwise, and μ is the threshold for the distance metric $D(\cdot)$. This is an extension of single-frame distance metric $D(\cdot)$ to a set of consecutive frames. Only when distance metric values for all R single frame attacks are higher than the threshold μ, the temporal metric $I(\ldots)$ outputs a 1.

4. EXPERIMENTAL RESULTS

We have performed evaluations by using BDD10k [12] and Cityscapes [11] datasets, implemented both white-box and black-box attacks, and inserted adversarial frames into the videos to build our adversarial evaluation set. For white-box attack, we used the CW_{inf}, which is a state-of-the-art white-box attack approach. For black-box attacks, we employed momentum iterative fast gradient sign method (MI-FGSM) [4], translation invariant momentum diverse inputs (TI-DIM) [21], Momentum Diverse Inputs Fast Gradient Sign Method (DIM) [22] and dispersion reduction (DR) [23]. For auxiliary image transformation $T(.)$, we employ bit-wise squeeze as one of the best performing feature squeezing methods. For generating object detection outputs, we use

Yolo-v3 [24], which is widely employed in autonomous driving perception systems. We compare the AE detection performance of our proposed wAP-based method with that of mAP and mIoU baselines.

4.1. Evaluation on Single-frame AE Detection

For single-frame detection, we compute the distance $D(\cdot)$ between an input image and its version transformed with $T()$, and use a threshold to decide whether the image is an AE. The results of single-frame AE detection are summarized in Table 1 for BDD10k and CityScapes datasets. We used Bit-wise (4, 5, 6, 7) squeeze as the transformation. Our proposed wAP-based approach outperforms the mAP-based and mIoU-based algorithms on both datasets and for five different attack methods. On 38 out of 40 experiment configurations, including black-box and white-box attacks, our approach achieves higher AE detection accuracy. For instance, for the C&W attack and Bit7 squeeze, our method provides 72.28% and 73.93% detection accuracy on Bdd10k and Cityscapes datasets, respectively. For the DR attack, which was shown to be an effective and transferable attack [23], and with Bit7 squeeze, our proposed wAP approach achieves a detection accuracy of 72.53% compared to 65.72% and 67.29% provided by mAP and mIoU, respectively, on the Cityscapes dataset For the TI-DIM attack, on the Cityscapes dataset with Bit7 squeeze, our wAP approach increases the detection accuracy from 65.97% and 68.74% to 72.7%, compared to using mAP and mIoU, respectively.

Accuracy(%)		Bdd10k			Cityscapes		
Attack method	Squeeze method	mAP	mIoU	wAP (ours)	mAP	mIoU	wAP (ours)
C&W [1]	bit4	59.95	62.76	**64.55**	**71.20**	65.84	69.47
	bit5	67.35	67.62	**68.87**	72.24	71.99	**74.05**
	bit6	68.18	**71.82**	71.60	73.68	74.10	**74.18**
	bit7	69.41	71.79	**72.28**	73.68	73.27	**73.93**
DR [23]	bit4	68.43	63.15	**73.50**	56.51	62.20	**66.39**
	bit5	70.84	69.62	**75.14**	59.03	65.09	**71.46**
	bit6	70.67	72.10	**75.65**	67.91	68.74	**72.63**
	bit7	71.33	72.75	**73.20**	65.72	67.29	**72.53**
MI-FGSM [4]	bit4	68.02	61.23	**73.21**	71.20	62.15	**74.20**
	bit5	65.05	66.86	**69.43**	71.78	67.22	**75.00**
	bit6	69.80	70.99	**72.17**	70.78	69.21	**72.12**
	bit7	68.28	72.19	**72.90**	72.18	69.04	**72.92**
TI-DIM [21]	bit4	68.07	61.54	**73.62**	69.71	60.96	**73.45**
	bit5	67.47	68.09	**69.43**	67.29	65.42	**71.41**
	bit6	68.78	70.47	**72.38**	66.37	69.06	**71.75**
	bit7	69.87	71.83	**72.39**	65.97	68.74	**72.70**
DIM [22]	bit4	69.20	59.90	**72.75**	70.76	62.6	**74.94**
	bit5	64.54	67.63	**69.28**	69.16	67.44	**70.88**
	bit6	68.36	71.48	**71.91**	70.70	69.81	**72.50**
	bit7	69.72	71.00	**71.88**	68.66	69.26	**72.03**

Table 1. Comparison of AE detection accuracy against different attacks, with different-bit feature squeezing, when using mAP, mIoU and the proposed wAP metric.

4.2. Evaluation on Temporal AE Detection

The state-of-the-art temporal attack can successfully attack a tracking system in 3 consecutive frames [16]. Thus, for this experiment, we set the number of frames for temporal consistency to $i = 3$. We randomly chose multiples of 3 consecutive frames from videos to insert AEs. Since these intervals can overlap or neighbor each other, the number of consecutive AEs can be greater than or equal to 3. Table 2 shows the detection accuracy values for single-frame and temporal detection of AEs by the proposed wAP method, and the mAP and mIoU baselines when bit7 squeezing is used. As seen in Tab. 2, our proposed temporal wAP-based detection provides the highest AE detection accuracy for all the attack methods. For example, with C&W attack, proposed wAP increases the detection accuracy from 93.4% and 89.8% to *97.6%* compared to mAP and mIoU, respectively. For MI-FGSM attack, wAP increases the detection accuracy from 88.2% and 94.2% to *97.2%*. The table also shows that temporal detection provides much better accuracy than single-frame detection.

	Single Frame			Temporal		
Attack method	mAP	mIoU	wAP (proposed)	mAP	mIoU	wAP (proposed)
C&W [1]	69.41	71.79	72.28	93.40	89.80	**97.60**
DR [23]	71.33	72.75	73.20	84.20	90.20	**91.20**
MI-FGSM [4]	68.28	72.19	72.90	88.20	94.20	**97.20**
TI-DIM [21]	69.87	71.83	72.39	88.80	94.60	**97.40**
DIM [22]	69.72	71.00	71.88	86.77	91.72	**95.94**

Table 2. Comparison of AE detection accuracy on BDD10k dataset, using both single-frame and temporal detection, against different attacks when using proposed wAP, mAP and mIoU metrics.

In summary, compared to the mAP and mIoU baselines, our proposed wAP metric is optimized, focuses on single image instead of the whole dataset, and introduces weights to bounding boxes. Moreover, since the proposed wAP metric is calculated by making use of IoU, our AE detection method can be applied to any perception task that is based on overlapping of detection results, such as object detection, semantic segmentation, and text detection and recognition. As for the proposed temporal algorithm, any application that is based on temporal information or tracking, such as video and audio processing, can benefit from our approach.

5. CONCLUSION

We have proposed a new weighted average precision (wAP) distance metric, and temporal optimization method to improve the detection of AEs for object detection in autonomous driving perception systems. Our proposed wAP metric focuses on bounding boxes in individual images, and can be applied to a sequence of frames to fit into a tracking system. The motivation behind the temporal wAP is that attacking a single image frame is not enough to successfully deceive a vehicle's perception system, since it also involves tracking, which is able to make predictions when bounding boxes of objects cannot be detected or missed for a small number of frames. Evaluation on different autonomous driving datasets, and with a variety of white-box and black-box attacks shows that our proposed pipeline greatly enhances the AE detection performance compared to the mAP-based and mIoU-based baselines.

6. REFERENCES

[1] Nicholas Carlini and David Wagner, "Towards evaluating the robustness of neural networks," in *2017 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2017, pp. 39–57.

[2] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus, "Intriguing properties of neural networks," *arXiv preprint arXiv:1312.6199*, 2013.

[3] Wen Zhou, Xin Hou, Yongjun Chen, Mengyun Tang, Xiangqi Huang, Xiang Gan, and Yong Yang, "Transferable adversarial perturbations," *In Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 452–467, 2018.

[4] Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Xiaolin Hu, and Jun Zhu, "Discovering adversarial examples with momentum," *CoRR, abs/1710.06081, 2017*, 2017.

[5] Nicolas Papernot, Patrick McDaniel, Arunesh Sinha, and Michael Wellman, "Towards the science of security and privacy in machine learning," *arXiv preprint arXiv:1611.03814*, 2016.

[6] Alexey Kurakin, Ian Goodfellow, and Samy Bengio, "Adversarial examples in the physical world," *arXiv preprint arXiv:1607.02533*, 2016.

[7] Shixiang Gu and Luca Rigazio, "Towards deep neural network architectures robust to adversarial examples," *arXiv preprint arXiv:1412.5068*, 2014.

[8] Kathrin Grosse, Praveen Manoharan, Nicolas Papernot, Michael Backes, and Patrick McDaniel, "On the (statistical) detection of adversarial examples," *arXiv preprint arXiv:1702.06280*, 2017.

[9] Kevin Eykholt, Ivan Evtimov andEarlence Fernandes, Bo Li, Amir Rahmati, Chaowei Xiao, Atul Prakash, Tadayoshi Kohno, and Dawn Song, "Robust physical-world attacks on deep learning visual classification," *arXiv preprint arXiv:1707.08945*, 2018.

[10] "Open images 2019 - object detection," `https://www.kaggle.com/c/open-images-2019-object-detection/overview`, 2019.

[11] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 3213–3223.

[12] Fisher Yu, Wenqi Xian, Yingying Chen, Fangchen Liu, Mike Liao, Vashisht Madhavan, and Trevor Darrell, "Bdd100k: A diverse driving video database with scalable annotation tooling," *arXiv preprint arXiv:1805.04687*, 2018.

[13] Robert Grover Brown, Patrick YC Hwang, et al., *Introduction to random signals and applied Kalman filtering*, vol. 3, Wiley New York, 1992.

[14] "Baidu Apollo Platform," `https://github.com/ApolloAuto/apollo`.

[15] Shinpei Kato, Eijiro Takeuchi, Yoshio Ishiguro, Yoshiki Ninomiya, Kazuya Takeda, and Tsuyoshi Hamada, "An open approach to autonomous vehicles," *IEEE Micro*, vol. 35, no. 6, pp. 60–68, 2015.

[16] Yunhan Jia, Yantao Lu, Junjie Shen, Qi Alfred Chen, Zhenyu Zhong, and Tao Wei, "Fooling detection alone is not enough: First adversarial attack against multiple object tracking," *CoRR*, vol. abs/1905.11026, 2019.

[17] Shu-Li Sun and Zi-Li Deng, "Multi-sensor optimal information fusion kalman filter," *Automatica*, vol. 40, no. 6, pp. 1017–1023, 2004.

[18] Yan Luo, Xavier Boix, Gemma Roig, Tomaso Poggio, and Qi Zhao, "Foveation-based mechanisms alleviate adversarial examples," *arXiv preprint arXiv:1511.06292*, 2015.

[19] Weilin Xu, David Evans, and Yanjun Qi, "Feature squeezing: Detecting adversarial examples in deep neural networks," *arXiv preprint arXiv:1704.01155*, 2017.

[20] Dongyu Meng and Hao Chen, "Magnet: a two-pronged defense against adversarial examples," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2017, pp. 135–147.

[21] Yinpeng Dong, Tianyu Pang, Hang Su, and Jun Zhu, "Evading defenses to transferable adversarial examples by translation-invariant attacks," *arXiv preprint arXiv:1904.02884*, 2019.

[22] Cihang Xie, Zhishuai Zhang, Jianyu Wang, Yuyin Zhou, Zhou Ren, and Alan Yuille, "Improving transferability of adversarial examples with input diversity," *arXiv preprint arXiv:1803.06978*, 2018.

[23] Yantao Lu, Yunhan Jia, Jianyu Wang, Bai Li, Weiheng Chai, Lawrence Carin, and Senem Velipasalar, "Enhancing cross-task black-box transferability of adversarial examples with dispersion reduction," *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 940–949, 2020.

[24] Joseph Redmon and Ali Farhadi, "Yolov3: An incremental improvement," *arXiv*, 2018.