Fooling automated surveillance cameras: adversarial patches to attack person detection

Danush S
s.danush@niser.ac.in

NISER

January 01, 2021 (Happy New Year!)

Introduction

- Where are Machine Learning detection/classification algorithms used?
- Security and Surveillance
- As [1] states "an arms race between the designers of learning systems and their adversaries"

Previous Research

Back in 2014, [1] showed the existence of adversial attacks in classification problems. One example they looked into was modifying input images of the MNIST database to mislead classification. And an interesting reference they cited looked into using Nash Equilibrium in these zero-sum problems. [2] cites a paper wherein the authors generate a sticker that can be applied to a stop sign to make it unrecognizable to a stop sign classifier.

Adversial patches against person-detectors

Similar to generating patches for stop-sign detection, this paper focussed on generating patches for person-detectors.
The YOLOv2 object detector was targeted (trained on the MS COCO dataset, but testing for the patch problem was done using the Inria dataset). Once an input image is passed, the algorithm generates an output grid and each cell in this output grid contains five anchor points. Each anchor point contains:
$[x_{offset}, y_{offset}, w, h, p_{obj}, p_{cls1}, p_{cls2}, ..., p_{clsn}]$.
x_{offset} and y_{offset} is the position of the center of the bounding box compared to the current anchor point, p_{obj} is the probability that this anchor point contains an object, and p_{clsi} are class scores of the object learned using cross entropy loss.

First the person-detector is run over the dataset of images. This yields bounding boxes that show where people occur in the image according to the detector. On a fixed position relative to these bounding boxes, the patch is applied to the image. The resulting image is then fed (in a batch together with other images) into the detector. The score of the persons that are still detected is then measured and used to calculate the loss function. Using back propagation over the entire network, the optimiser then changes the pixels in the patch further in order to fool the detector even more.

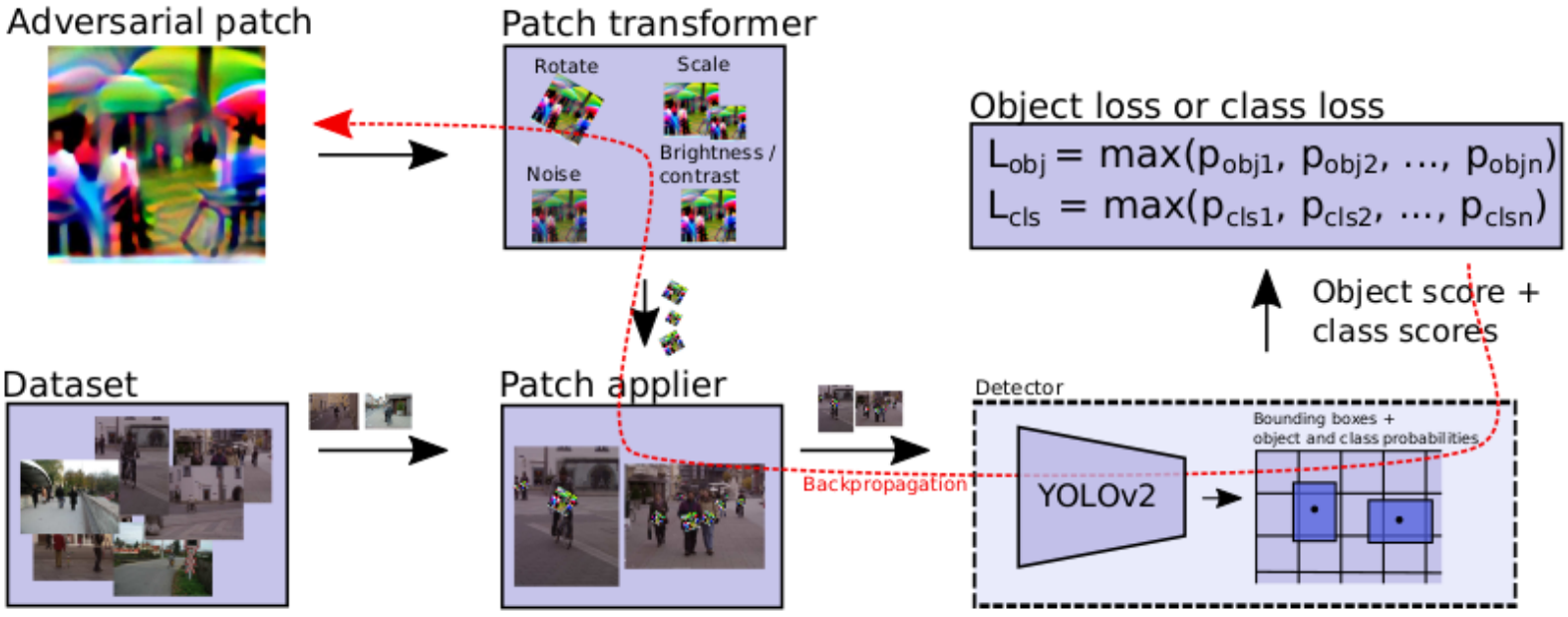

Figure: Pipeline

The authors also perform the following random transformations on the patch before applying:

- The patch is rotated up to 20 degrees.
- The patch is scaled up and down randomly.
- Random noise is put on top of the patch.
- The brightness and contrast of the patch is changed randomly.

Can you guess why?

Loss Function

- Non-printability score.
  $L_{nps} = \sum_{p_{\text{patch}} \in P} \min_{c_{\text{print}} \in C} \left| p_{\text{patch}} - c_{\text{print}} \right|$
- Loss function to reduce abrupt colour transitions.
  $L_{tv} = \sum_{i,j} \sqrt{\left((p_{i,j} - p_{i+1,j})^2 + (p_{i,j} - p_{i,j+1})^2 \right.}$
- Minimizing the object or class score outputted by the detector (L_{obj}).

Total loss function: $L = \alpha * L_{nps} + \beta * L_{tv} + L_{obj}$

Evaluation

The patches were initialised with random pixel values and there were 3 different approaches:

- Minimize object score (OBJ)
- Minimize class score (CLS)
- Minimize object and class score (product of object and class score) (OBJ-CLS)

As a control, the authors also compare results to a patch containing random noise (NOISE) that was evaluated in the exact same way as the random patches. Results of CLEAN is the baseline with no patch applied.

Results

Approach	Recall (%)
CLEAN	100
NOISE	87.14
OBJ	26.46
CLS	77.58
OBJ-CLS	39.31

Table: Comparison of different approaches in recall

Figure: Some photos attached by the authors

Bibliography I

Battista Biggio et al. "Evasion attacks against machine learning at test time". In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases. Springer, Berlin, Heidelberg, 2013, pp. 387–402. ISBN: 978-3-642-40994-3. DOI: https://doi.org/10.1007/978-3-642-40994-3_25.

Simen Thys, Wiebe Van Ranst, and Toon Goedemé. "Fooling automated surveillance cameras: adversarial patches to attack person detection". In: *CoRR* abs/1904.08653 (2019). _eprint: 1904.08653. URL: http://arxiv.org/abs/1904.08653.