THE
BLK
SWN

THE BLK SWN

INTRODUCTION TO ADVERSARIAL ML

HACKING NEURAL NETWORKS - FGSM

THE BLK SWN

INTRODUCTION

Adversarial machine learning is the study of machine learning vulnerabilities in adversarial environments.

White-Box-Attack ⟷ Black-Box-Attack

White-Box-Attack	Black-Box-Attack
Attacker has access to the model's parameter	Attacker has no access to the model's parameter

Non-targeted attack	Targeted attack
Attacker has access to the model's parameter	Attacker has no access to the model's parameter

Source:

- https://pytorch.org/tutorials/beginner/fgsm_tutorial.html
- https://medium.com/onfido-tech/adversarial-attacks-and-defences-for-convolutional-neural-networks-66915ece52e7
- Machine Learning and Security by Clarence Chio and David Freeman (O'Reilly). Copyright 2018 Clarence Chio and David Freeman, ISBN 978-1-491-97990-7.

Missclassification

Attacker only wants the output classification to be wrong; he/she doesn't care what the new classification is

Source/Target Missclassification

Attacker wants to classify an image from a specific source class to be classified in a specific target class

One-shot attacks

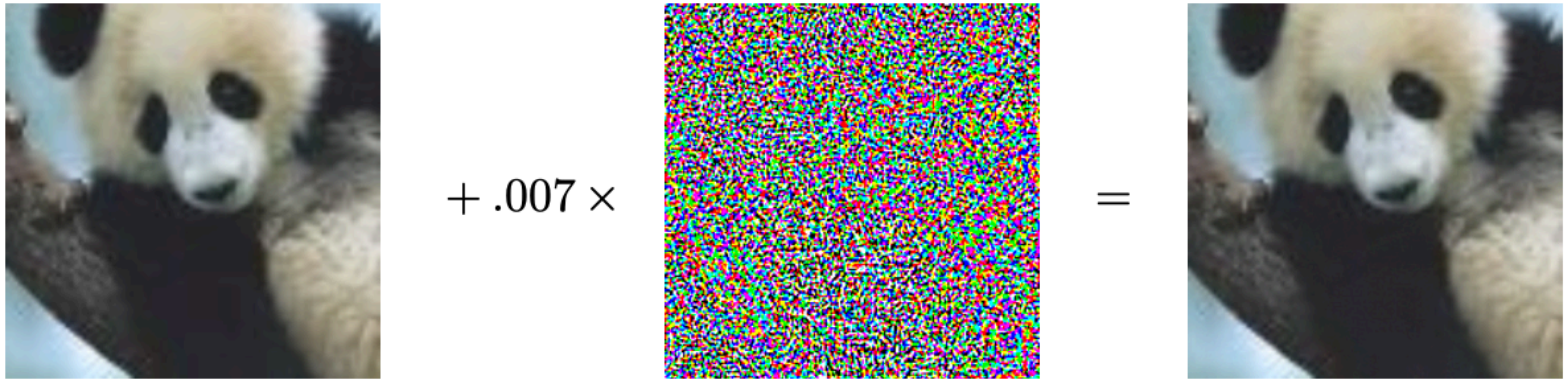
Attacker takes a single step in the direction of the gradient

Iterative attack

Attacker takes several steps

Adversarial Image

THE BLK SWN



$\boldsymbol{x}$	$\text{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y))$	$\boldsymbol{x} + \epsilon\text{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y))$
"panda" 57.7% confidence	"nematode" 8.2% confidence	"gibbon" 99.3 % confidence

Goodfellow, I. et al. (2014): Explaining and harnessing adversarial examples. In Proceedings of the International Conference on Learning Representations, 2015. [https://arxiv.org/abs/1911.07658]

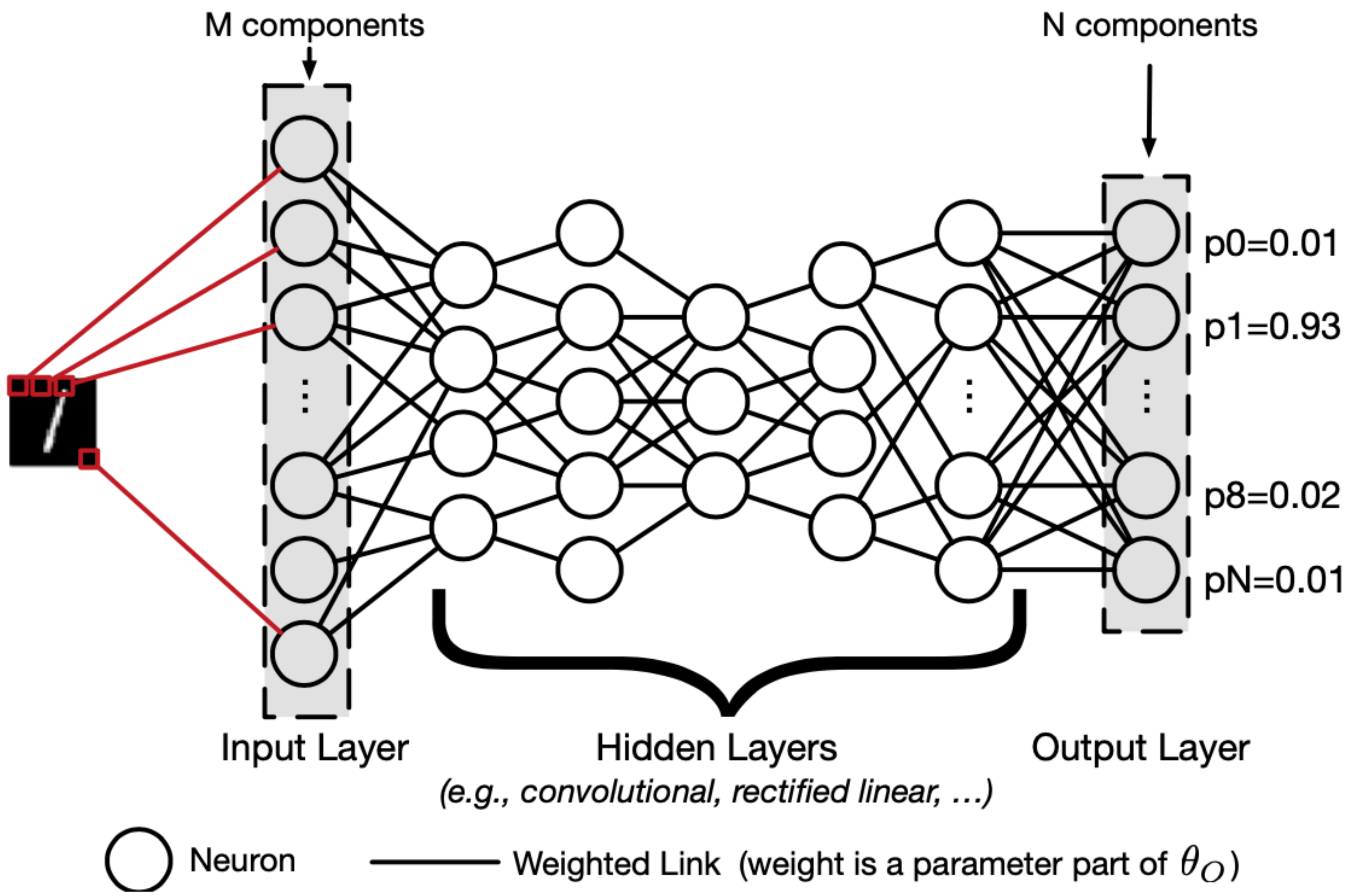


Figure 1: **DNN Classifier:** the model processes an image of a handwritten digit and outputs the probility of it being in one of the $N = 10$ classes for digits 0 to 9 (from [10]).

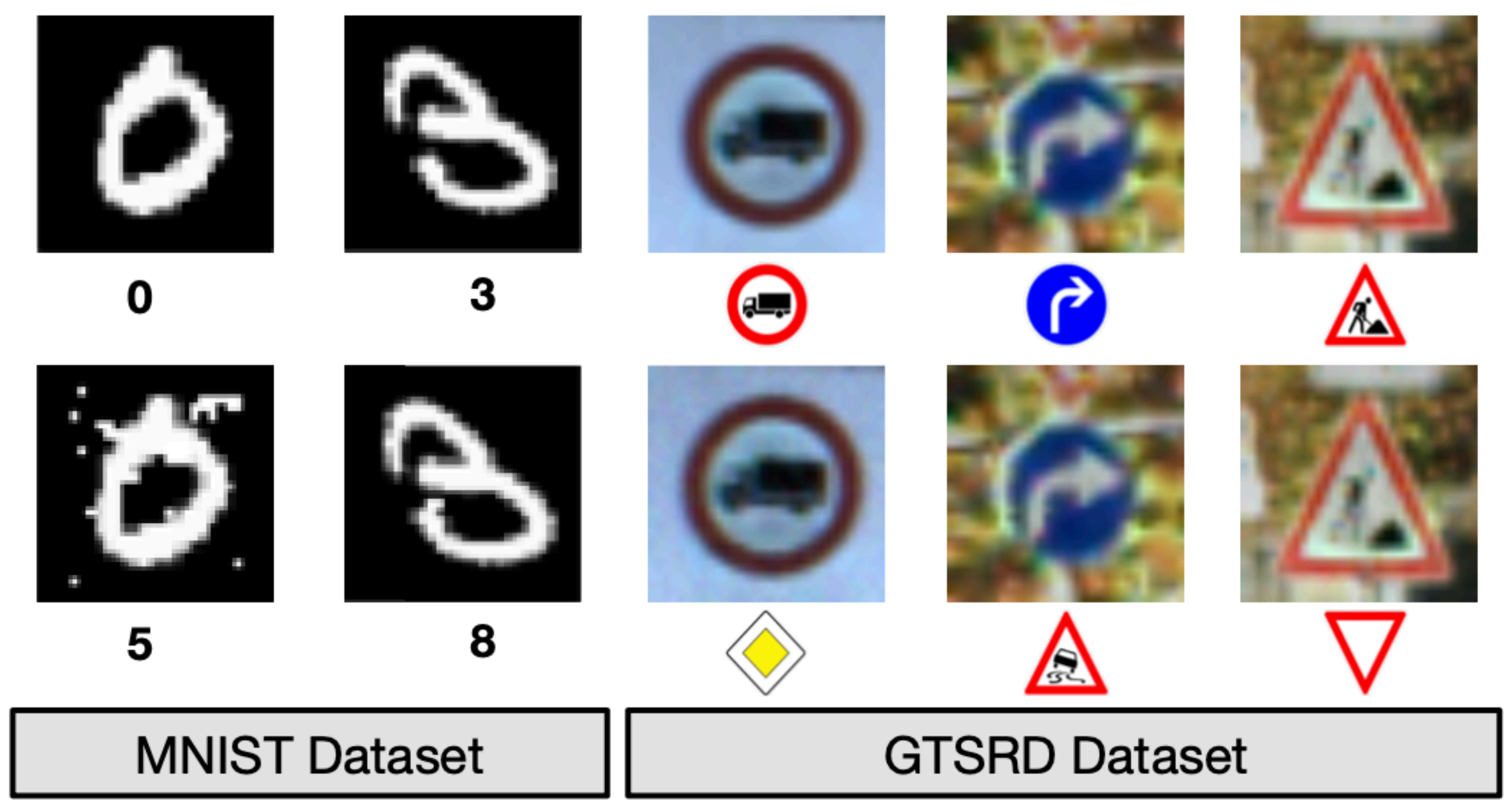


Figure 2: **Adversarial samples** (misclassified) in the bottom row are created from the legitimate samples [7, 13] in the top row. The DNN outputs are identified below the samples.

Goodfellow, I. et al. (2014): Practical Black-Box Attacks against Machine Learning. In Proceedings of the 2017 ACM Asia Conference on Computer and Communications Security, Abu Dhabi, UAE, 2017. [https://arxiv.org/abs/1602.02697]

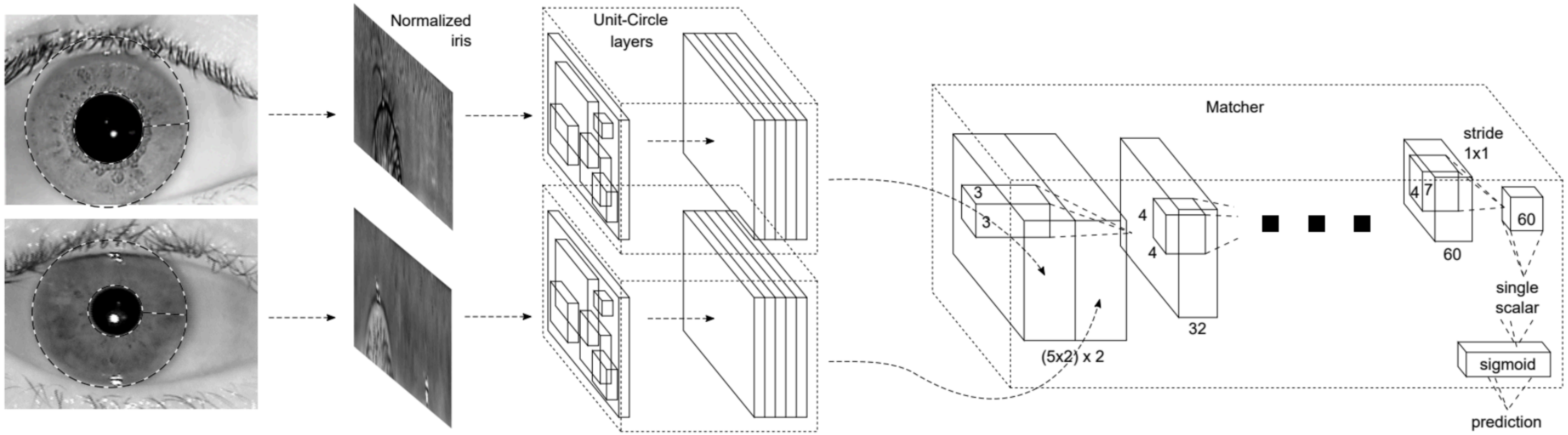


Figure 13: Iris verification with IrisMatch-CNN. Two irises are detected and normalized. The normalized irises are fed into the Unit-Circle (U-C) layers. The responses from the U-C layers are concatenated and fed into the Matcher convolutional network. A single scalar is produced – the probability of a match. Two irises match if the probability is greater than a given threshold. Figure and description from [57].

Kissner, M. (2019): Hacking Neural Networks: A Short Introduction
[https://arxiv.org/abs/1911.07658]

THE BLK SWN

FAST GRADIENT SIGN METHOD (FGSM)

- white-box-attack & missclassification
- attack adjusts the input data to maximize the loss based on the same backpropagated gradients
- the attack uses the gradient of the loss with respect to the input data, then adjusts the input data to maximize the loss
- <u>goal</u>: create an image which maximize the loss
- <u>important</u>: model is not trained anymore; parameters of the model are fix, hence model is already trained

$$adv\_x = x + \epsilon * \text{sign}(\nabla_x J(\theta, x, y))$$

where

- adv_x : Adversarial image.
- x : Original input image.
- y : Original input label.
- ϵ : Multiplier to ensure the perturbations are small.
- θ : Model parameters.
- J : Loss.

https://pytorch.org/tutorials/beginner/fgsm_tutorial.html

OTHER TYPES OF ADVERSARIAL ML

THE BLK SWN

1. SUPPLY CHAIN ATTACK
2. BACKDOORING NEURAL NETWORKS
3. EXTRACTING INFORMATION
4. BRUTE-FORCING
5. NEURAL OVERFLOW
6. NEURAL MALWARE INJECTION
7. NEURAL OBFUSCATION
8. BUG HUNTING
9. GPU ATTACK

THE
BLK
SWN

TIN VOTAN

CEO & FOUNDER

Software Engineer
Machine Learning Engineer
M.Sc. Wirtschaftsingenieurwesen

linkedin.com/in/tinvotan/

@tin_votan