CNN Robustness research

Application to face detectors and face ID systems

Aleksandr Petiushko

Lomonosov Moscow State University, Ph.D.
Huawei, Video Intelligence Team Leader

4th of February, 2021

Plan

1. Intelligent Systems and Data Science Technology Center

Plan

1. Intelligent Systems and Data Science Technology Center
2. CNN great success

Plan

1. Intelligent Systems and Data Science Technology Center
2. CNN great success
3. CNN lack of robustness

Plan

1. Intelligent Systems and Data Science Technology Center
2. CNN great success
3. CNN lack of robustness
4. ℓ_0-based adversaries

Plan

1. Intelligent Systems and Data Science Technology Center
2. CNN great success
3. CNN lack of robustness
4. ℓ_0-based adversaries
5. Adversarial examples in real world

Plan

1. Intelligent Systems and Data Science Technology Center
2. CNN great success
3. CNN lack of robustness
4. ℓ_0-based adversaries
5. Adversarial examples in real world
6. Adversarial attack on face detection

Plan

1. Intelligent Systems and Data Science Technology Center
2. CNN great success
3. CNN lack of robustness
4. ℓ_0-based adversaries
5. Adversarial examples in real world
6. Adversarial attack on face detection
7. Adversarial attack on face ID

Plan

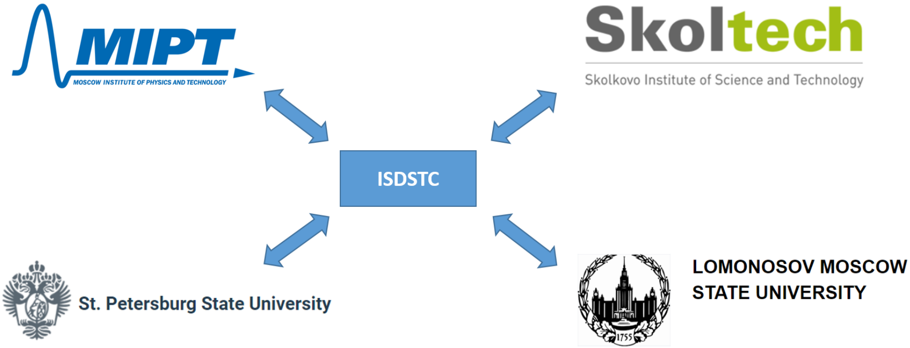
1. Intelligent Systems and Data Science Technology Center
2. CNN great success
3. CNN lack of robustness
4. ℓ_0-based adversaries
5. Adversarial examples in real world
6. Adversarial attack on face detection
7. Adversarial attack on face ID
8. Defense from adversarial examples in real world

Plan

1. Intelligent Systems and Data Science Technology Center
2. CNN great success
3. CNN lack of robustness
4. ℓ_0-based adversaries
5. Adversarial examples in real world
6. Adversarial attack on face detection
7. Adversarial attack on face ID
8. Defense from adversarial examples in real world
9. Black-box face restoration

Intelligent Systems and Data Science Technology Center: scientific collaboration

Russian Research Institute → Moscow Research Center → Intelligent Systems and Data Science Technology Center



ImageNet[1] (1000-class image DB)

- Human expert top-5 error[2]: 5.1%
- CNN top-5 error[3]: 2.0%

---

[1] http://www.image-net.org/
[2] Andrej Karpathy blog
[3] Touvron H. et al. "Fixing the train-test resolution discrepancy." 2019
[4] http://vis-www.cs.umass.edu/lfw/
[5] Kumar N. et al. "Attribute and simile classifiers for face verification." 2009
[6] Deng J. et al. "Arcface: Additive angular margin loss for deep face recognition." 2018

Human expert VS CNN

ImageNet[1] (1000-class image DB)

- Human expert top-5 error[2]: 5.1%
- CNN top-5 error[3]: 2.0%

Labeled Faces in the Wild[4] (famous faces DB)

- Human expert verification error[5]: 2.47%
- CNN verification error[6]: 0.17%

---

[1] http://www.image-net.org/
[2] Andrej Karpathy blog
[3] Touvron H. et al. "Fixing the train-test resolution discrepancy." 2019
[4] http://vis-www.cs.umass.edu/lfw/
[5] Kumar N. et al. "Attribute and simile classifiers for face verification." 2009
[6] Deng J. et al. "Arcface: Additive angular margin loss for deep face recognition." 2018

Human expert VS CNN

ImageNet[1] (1000-class image DB)

- Human expert top-5 error[2]: 5.1%
- CNN top-5 error[3]: 2.0%

Labeled Faces in the Wild[4] (famous faces DB)

- Human expert verification error[5]: 2.47%
- CNN verification error[6]: 0.17%

LFW

[1] http://www.image-net.org/
[2] Andrej Karpathy blog
[3] Touvron H. et al. "Fixing the train-test resolution discrepancy." 2019
[4] http://vis-www.cs.umass.edu/lfw/
[5] Kumar N. et al. "Attribute and simile classifiers for face verification." 2009
[6] Deng J. et al. "Arcface: Additive angular margin loss for deep face recognition." 2018

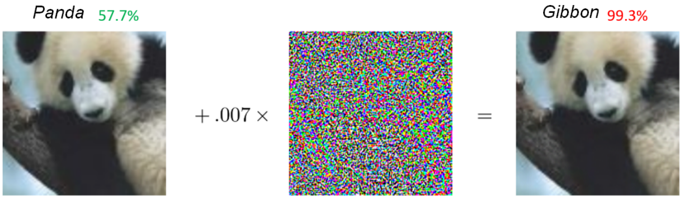
CNN instability

- It turned out that one can add to the input almost invisible to the human eye perturbation in such a way that this perturbation completely changes the CNN output

[7] Image credit: `https://arxiv.org/pdf/1412.6572.pdf`

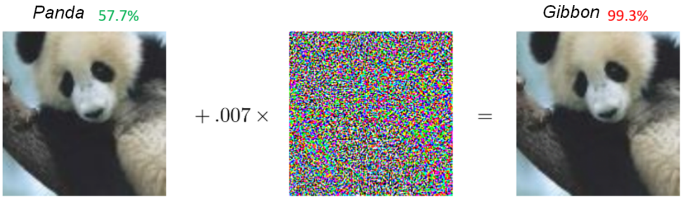
CNN instability

- It turned out that one can add to the input almost invisible to the human eye perturbation in such a way that this perturbation completely changes the CNN output
- E.g. classification result from “Panda” changes to “Gibbon”[7]



[7] Image credit: `https://arxiv.org/pdf/1412.6572.pdf`

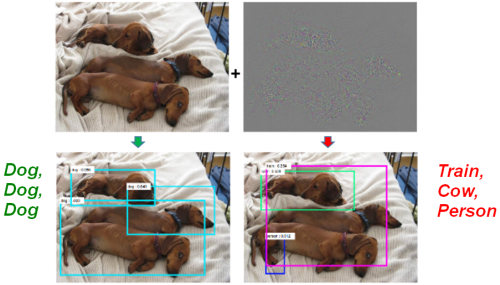
- It turned out that one can add to the input almost invisible to the human eye perturbation in such a way that this perturbation completely changes the CNN output
- E.g. classification result from "Panda" changes to "Gibbon"[7]



Such almost invisible perturbations leading to changing of the CNN output are called **adversarial examples** (or **adversarial attacks** on CNN)

[7] Image credit: `https://arxiv.org/pdf/1412.6572.pdf`

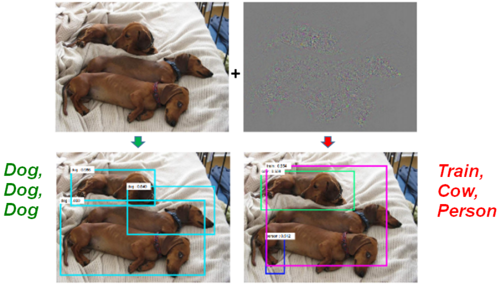
Detection and segmentation[8] CNNs:



[8] Xie C. et al. "Adversarial examples for semantic segmentation and object detection." 2017
[9] Jia R. et al. "Adversarial examples for evaluating reading comprehension systems." 2017

Different types of NN to attack

Detection and segmentation[8] CNNs:



And even NN for text processing (question answering systems)[9]:

Article: Super Bowl 50
Paragraph: "*Peyton Manning became the first quarterback ever to lead two different teams to multiple Super Bowls. He is also the oldest quarterback ever to play in a Super Bowl at age 39. The past record was held by John Elway, who led the Broncos to victory in Super Bowl XXXIII at age 38 and is currently Denver's Executive Vice President of Football Operations and General Manager. Quarterback Jeff Dean had jersey number 37 in Champ Bowl XXXIV.*"
Question: "*What is the name of the quarterback who was 38 in Super Bowl XXXIII?*"
Original Prediction: John Elway
Prediction under adversary: Jeff Dean

[8] Xie C. et al. "Adversarial examples for semantic segmentation and object detection." 2017
[9] Jia R. et al. "Adversarial examples for evaluating reading comprehension systems." 2017

Definitions

- $x \in B = [0,1]^{C \times M \times N}$ — input image $C \times M \times N$, where C — number of color channels (1 for grayscale, 3 for RGB)
- y — correct class label for input x
- θ — parameters of CNN-classifier
- $L(\theta, x, y)$ — loss function
- $f(x)$ — output of classifier (recognized class), and we are trying to make $f(x) = y$ when training

Definitions

- $x \in B = [0,1]^{C \times M \times N}$ — input image $C \times M \times N$, where C — number of color channels (1 for grayscale, 3 for RGB)
- y — correct class label for input x
- θ — parameters of CNN-classifier
- $L(\theta, x, y)$ — loss function
- $f(x)$ — output of classifier (recognized class), and we are trying to make $f(x) = y$ when training
- $r \in B = [0,1]^{C \times M \times N}$ — the additive perturbation for the input x

Definition of adversarial example and robustness

Goal of adversarial attack

To change the output of the classifier f from the correct class label to the incorrect one by means of minimal in terms of some norm ℓ_p perturbation r:

1. $||r||_p \to \min$ so as:

Definition of adversarial example and robustness

Goal of adversarial attack

To change the output of the classifier f from the correct class label to the incorrect one by means of minimal in terms of some norm ℓ_p perturbation r:

1. $||r||_p \to \min$ so as:
2. $f(x) = y$ (initially the output is correct)
3. $f(x + r) \neq y$ ("break" the CNN output with perturbation r)
4. $x + r \in B$ (still in the space of correct images)

Definition of adversarial example and robustness

Goal of adversarial attack

To change the output of the classifier f from the correct class label to the incorrect one by means of minimal in terms of some norm ℓ_p perturbation r:

1. $||r||_p \to \min$ so as:
2. $f(x) = y$ (initially the output is correct)
3. $f(x+r) \neq y$ ("break" the CNN output with perturbation r)
4. $x + r \in B$ (still in the space of correct images)

Classifier robustness

To find the perturbation class $S(x, f) \subseteq B$ so as the classifier will not change its output:

$$f(x+r) = f(x) = y \quad \forall r \in S(x, f)$$

Digital space attacks: FGSM, iterative methods

Most attacks on CNN are done in terms of ℓ_∞-norm which is correlated with the process of how a human eye perceive the visual information:

$$||x||_\infty = \max_i |x_i|, x = (x_1, \ldots, x_n) \in \mathbb{R}^n$$

[10] Goodfellow I. et al. "Explaining and harnessing adversarial examples." 2014
[11] Kurakin A. et al. "Adversarial examples in the physical world." 2016
[12] Madry A. et al. "Towards deep learning models resistant to adversarial attacks." 2017

Digital space attacks: FGSM, iterative methods

Most attacks on CNN are done in terms of ℓ_∞-norm which is correlated with the process of how a human eye perceive the visual information:

$$||x||_\infty = \max_i |x_i|, x = (x_1, \ldots, x_n) \in \mathbb{R}^n$$

- **F**ast **G**radient **S**ign **M**ethod[10] (FGSM): $r = \epsilon \cdot \text{sign}(\nabla_x L(\theta, x, y))$

[10] Goodfellow I. et al. "Explaining and harnessing adversarial examples." 2014
[11] Kurakin A. et al. "Adversarial examples in the physical world." 2016
[12] Madry A. et al. "Towards deep learning models resistant to adversarial attacks." 2017

Digital space attacks: FGSM, iterative methods

Most attacks on CNN are done in terms of ℓ_∞-norm which is correlated with the process of how a human eye perceive the visual information:

$$||x||_\infty = \max_i |x_i|, x = (x_1, \dots, x_n) \in \mathbb{R}^n$$

- **F**ast **G**radient **S**ign **M**ethod[10] (FGSM): $r = \epsilon \cdot \text{sign}(\nabla_x L(\theta, x, y))$
- **I**terative FGSM (I-FGSM)[11] / **P**rojected **G**radient **D**escent (PGD)[12] (Π_B — the projection operation on B): $x^{t+1} = \Pi_B(x^t + \alpha \cdot \text{sign}\, \nabla_x L(\theta, x^t, y)), \quad x^0 = x, \alpha = \epsilon / T, t \leq T$

[10] Goodfellow I. et al. "Explaining and harnessing adversarial examples." 2014
[11] Kurakin A. et al. "Adversarial examples in the physical world." 2016
[12] Madry A. et al. "Towards deep learning models resistant to adversarial attacks." 2017

Digital space attacks: FGSM, iterative methods

Most attacks on CNN are done in terms of ℓ_∞-norm which is correlated with the process of how a human eye perceive the visual information:

$$||x||_\infty = \max_i |x_i|, x = (x_1, \ldots, x_n) \in \mathbb{R}^n$$

- **F**ast **G**radient **S**ign **M**ethod[10] (FGSM): $r = \epsilon \cdot \text{sign}(\nabla_x L(\theta, x, y))$
- **I**terative FGSM (I-FGSM)[11] / **P**rojected **G**radient **D**escent (PGD)[12] (Π_B — the projection operation on B): $x^{t+1} = \Pi_B(x^t + \alpha \cdot \text{sign}\, \nabla_x L(\theta, x^t, y)), \quad x^0 = x, \alpha = \epsilon / T, t \leq T$
- **M**omentum I-FGSM (MI-FGSM):
$g^{t+1} = \mu \cdot g^t + \frac{\nabla_x L(\theta, x^t, y)}{||\nabla_x L(\theta, x^t, y)||_1}, x^{t+1} = \Pi_B(x^t + \alpha \cdot \text{sign}(g^{t+1})), \quad x^0 = x, g^0 = 0$

[10] Goodfellow I. et al. "Explaining and harnessing adversarial examples." 2014
[11] Kurakin A. et al. "Adversarial examples in the physical world." 2016
[12] Madry A. et al. "Towards deep learning models resistant to adversarial attacks." 2017

Comparison of FGSM-like attacks

	Attack	Inc-v3	Inc-v4	IncRes-v2	Res-152	Inc-v3$_{ens3}$	Inc-v3$_{ens4}$	IncRes-v2$_{ens}$
Inc-v3	FGSM	72.3*	28.2	26.2	25.3	11.3	10.9	4.8
	I-FGSM	**100.0***	22.8	19.9	16.2	7.5	6.4	4.1
	MI-FGSM	**100.0***	**48.8**	**48.0**	**35.6**	**15.1**	**15.2**	**7.8**
Inc-v4	FGSM	32.7	61.0*	26.6	27.2	13.7	11.9	6.2
	I-FGSM	35.8	**99.9***	24.7	19.3	7.8	6.8	4.9
	MI-FGSM	**65.6**	**99.9***	**54.9**	**46.3**	**19.8**	**17.4**	**9.6**
IncRes-v2	FGSM	32.6	28.1	55.3*	25.8	13.1	12.1	7.5
	I-FGSM	37.8	20.8	**99.6***	22.8	8.9	7.8	5.8
	MI-FGSM	**69.8**	**62.1**	99.5*	**50.6**	**26.1**	**20.9**	**15.7**
Res-152	FGSM	35.0	28.2	27.5	72.9*	14.6	13.2	7.5
	I-FGSM	26.7	22.7	21.2	**98.6***	9.3	8.9	6.2
	MI-FGSM	**53.6**	**48.9**	**44.7**	98.5*	**22.1**	**21.7**	**12.9**

Based on it, MI-FGSM is one of the most successful ones.

ℓ_0-based adversaries

- ℓ_∞-based adversaries are imperceptible, but require all pixels to change

[13] Papernot N. et al. "The limitations of deep learning in adversarial settings." 2015
[14] Su J. et al. "One pixel attack for fooling deep neural networks." 2017

ℓ_0-based adversaries

- ℓ_∞-based adversaries are imperceptible, but require all pixels to change
- In the physical world it is not realistic — we can only change a part of the scene

[13] Papernot N. et al. "The limitations of deep learning in adversarial settings." 2015
[14] Su J. et al. "One pixel attack for fooling deep neural networks." 2017

ℓ_0-based adversaries

- ℓ_∞-based adversaries are imperceptible, but require all pixels to change
- In the physical world it is not realistic — we can only change a part of the scene
- So the ℓ_0-based attack could be more appropriate: $||x||_0 = \sum_{i=1}^{n} \mathbf{1}_{x_i \neq 0}$

[13] Papernot N. et al. "The limitations of deep learning in adversarial settings." 2015

[14] Su J. et al. "One pixel attack for fooling deep neural networks." 2017

ℓ_0-based adversaries

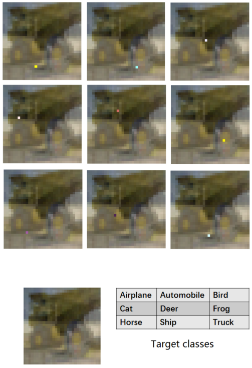
- ℓ_∞-based adversaries are imperceptible, but require all pixels to change
- In the physical world it is not realistic — we can only change a part of the scene
- So the ℓ_0-based attack could be more appropriate: $||x||_0 = \sum_{i=1}^{n} \mathbf{1}_{x_i \neq 0}$
- **J**acobian-based **S**aliency **M**ap **A**ttack (JSMA)[13] and even more extreme case — One Pixel attack[14] — are the examples of such ℓ_0-based attacks where the maximal amount of pixels to be changed is minimized

[13] Papernot N. et al. "The limitations of deep learning in adversarial settings." 2015
[14] Su J. et al. "One pixel attack for fooling deep neural networks." 2017

ℓ_0-based adversaries

- ℓ_∞-based adversaries are imperceptible, but require all pixels to change
- In the physical world it is not realistic — we can only change a part of the scene
- So the ℓ_0-based attack could be more appropriate: $||x||_0 = \sum_{i=1}^{n} \mathbf{1}_{x_i \neq 0}$
- Jacobian-based **S**aliency **M**ap **A**ttack (JSMA)[13] and even more extreme case — One Pixel attack[14] — are the examples of such ℓ_0-based attacks where the maximal amount of pixels to be changed is minimized

One Pixel attack



Original image (dog)

[13] Papernot N. et al. “The limitations of deep learning in adversarial settings.” 2015

[14] Su J. et al. “One pixel attack for fooling deep neural networks.” 2017

Adversarial examples in real world: EOT

- Don't have the control on the image pixels after the photo $\Rightarrow$ the only option is to change the object appearance itself
- **E**xpectation **O**ver **T**ransformation (EOT)[15] to the rescue — takes into account the transformations of objects in the real world, e.g.:
  - Different scaling factors
  - Random translation and rotation
  - Luminosity / contrast variation, noise etc

[15] Athalye A. et al. "Synthesizing robust adversarial examples." 2017

Adversarial examples in real world: EOT

- Don't have the control on the image pixels after the photo $\Rightarrow$ the only option is to change the object appearance itself
- **E**xpectation **O**ver **T**ransformation (EOT)[15] to the rescue — takes into account the transformations of objects in the real world, e.g.:
  - Different scaling factors
  - Random translation and rotation
  - Luminosity / contrast variation, noise etc
- So for the object x in the real world the task is to find the adversarial perturbation r taking into account transformation $g \in T$:

EOT

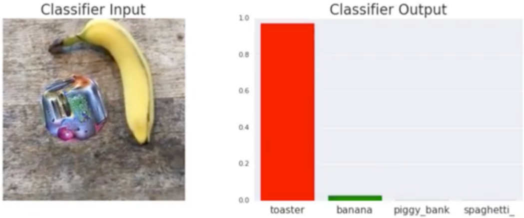
Find $\arg\min_r \mathbb{E}_{g\sim T}[P(y|g(x+r))]$ w.r.t.:

1. $\mathbb{E}_{g\sim T}[d(g(x+r), g(x))] < \epsilon$, where $d(a,b)$ – some distance function (e.g. $d(a,b) = ||a-b||_p$)
2. $x + r \in B$

[15] Athalye A. et al. "Synthesizing robust adversarial examples." 2017

Examples of physical adversarial examples

Attack on ImageNet obects[16]:

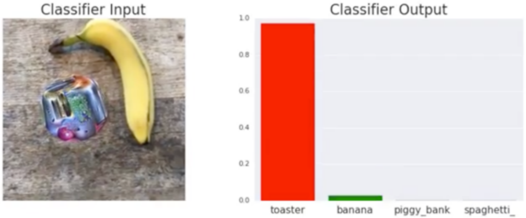


[16] Brown T. et al. "Adversarial patch." 2017

[17] Eykholt K. et al. "Robust physical-world attacks on deep learning models." 2017

Examples of physical adversarial examples

Attack on ImageNet obects[16]:



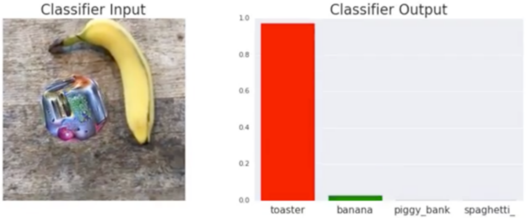
Attack on road signs[17]:

[16] Brown T. et al. "Adversarial patch." 2017

[17] Eykholt K. et al. "Robust physical-world attacks on deep learning models." 2017

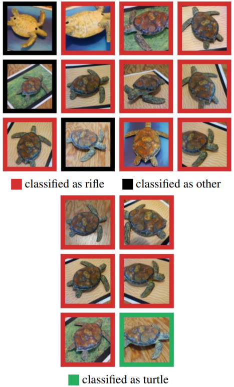
Examples of physical adversarial examples

Attack on ImageNet obects[16]:



Attack on road signs[17]:

3D adversarial objects:



[16] Brown T. et al. "Adversarial patch." 2017

[17] Eykholt K. et al. "Robust physical-world attacks on deep learning models." 2017

Physical adversarial examples: key ingredients

- ℓ_0-optimization (mask-based) + EOT: the must

Physical adversarial examples: key ingredients

- ℓ_0-optimization (mask-based) + EOT: the must
- **T**otal **V**ariation (TV) loss — penalty for the perturbation to be non-smooth (in the real world there is no distinct pixel gradients):

$$TV(x) = \sum_{i,j} \sqrt{(x_{i,j+1} - x_{i,j})^2 + (x_{i+1,j} - x_{i,j})^2}$$

Physical adversarial examples: key ingredients

- ℓ_0-optimization (mask-based) + EOT: <u>the must</u>
- **T**otal **V**ariation (TV) loss — penalty for the perturbation to be non-smooth (in the real world there is no distinct pixel gradients):

$$TV(x) = \sum_{i,j} \sqrt{(x_{i,j+1} - x_{i,j})^2 + (x_{i+1,j} - x_{i,j})^2}$$

- **N**on-**P**rintability **S**core (NPS) — penalty for the perturbation colors that are out of the generator device (e.g., printer) limited gamut. E.g. if $G \subset [0,1]^3$ — limited device gamut, then the loss for using the pixel $q_0 \in [0,1]^3$:

$$NPS(q_0) = \Pi_{q \in G} ||q - q_0||_2$$

Physical adversarial examples: key ingredients

- ℓ_0-optimization (mask-based) + EOT: the must
- **T**otal **V**ariation (TV) loss — penalty for the perturbation to be non-smooth (in the real world there is no distinct pixel gradients):

$$TV(x) = \sum_{i,j} \sqrt{(x_{i,j+1} - x_{i,j})^2 + (x_{i+1,j} - x_{i,j})^2}$$

- **N**on-**P**rintability **S**core (NPS) — penalty for the perturbation colors that are out of the generator device (e.g., printer) limited gamut. E.g. if $G \subset [0,1]^3$ — limited device gamut, then the loss for using the pixel $q_0 \in [0,1]^3$:

$$NPS(q_0) = \Pi_{q \in G} ||q - q_0||_2$$

- Additional color adjustments (e.g. generator device provides not color c, but some its modification $m(c)$)

Prior art: Face Det and ID attack

- Initially the so called **Camouflage Art**[18] was used to avoid the leading at that time Viola-Jones face detection system

[18] Feng R. et al. "Facilitating fashion camouflage art." 2013

[19] Sharif M. et al. "Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition." 2016

Prior art: Face Det and ID attack

- Initially the so called **Camouflage Art**[18] was used to avoid the leading at that time Viola-Jones face detection system
- It was just the makeup crafted manually to fool the Haar detector

[18] Feng R. et al. "Facilitating fashion camouflage art." 2013
[19] Sharif M. et al. "Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition." 2016

Prior art: Face Det and ID attack

- Initially the so called **Camouflage Art**[18] was used to avoid the leading at that time Viola-Jones face detection system
- It was just the makeup crafted manually to fool the Haar detector
- Pioneering work by Sharif et al.[19] proposed to use printed adversarial glasses

[18] Feng R. et al. "Facilitating fashion camouflage art." 2013
[19] Sharif M. et al. "Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition." 2016

Prior art: Face Det and ID attack

- Initially the so called **Camouflage Art**[18] was used to avoid the leading at that time Viola-Jones face detection system
- It was just the makeup crafted manually to fool the Haar detector
- Pioneering work by Sharif et al.[19] proposed to use printed adversarial glasses
- It uses ℓ_0-optimization + EOT + TV + NPS + color adjustments

[18] Feng R. et al. "Facilitating fashion camouflage art." 2013

[19] Sharif M. et al. "Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition." 2016

Prior art: Face Det and ID attack

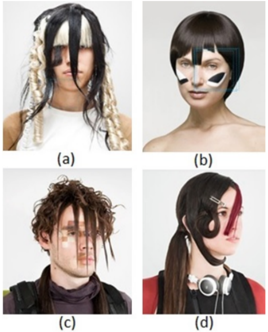
- Initially the so called **Camouflage Art**[18] was used to avoid the leading at that time Viola-Jones face detection system
- It was just the makeup crafted manually to fool the Haar detector
- Pioneering work by Sharif et al.[19] proposed to use printed adversarial glasses
- It uses ℓ_0-optimization + EOT + TV + NPS + color adjustments
- But it was used for closed-set recognition (a few predefined person ID for training) and for old generation FaceID NN

[18] Feng R. et al. "Facilitating fashion camouflage art." 2013
[19] Sharif M. et al. "Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition." 2016

Prior art: Face Det and ID attack

- Initially the so called **Camouflage Art**[18] was used to avoid the leading at that time Viola-Jones face detection system
- It was just the makeup crafted manually to fool the Haar detector
- Pioneering work by Sharif et al.[19] proposed to use printed adversarial glasses
- It uses ℓ_0-optimization + EOT + TV + NPS + color adjustments
- But it was used for closed-set recognition (a few predefined person ID for training) and for old generation FaceID NN

cvdazzle.com



[18] Feng R. et al. "Facilitating fashion camouflage art." 2013

[19] Sharif M. et al. "Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition." 2016

Prior art: Face Det and ID attack

- Initially the so called **Camouflage Art**[18] was used to avoid the leading at that time Viola-Jones face detection system
- It was just the makeup crafted manually to fool the Haar detector
- Pioneering work by Sharif et al.[19] proposed to use printed adversarial glasses
- It uses ℓ_0-optimization + EOT + TV + NPS + color adjustments
- But it was used for closed-set recognition (a few predefined person ID for training) and for old generation FaceID NN

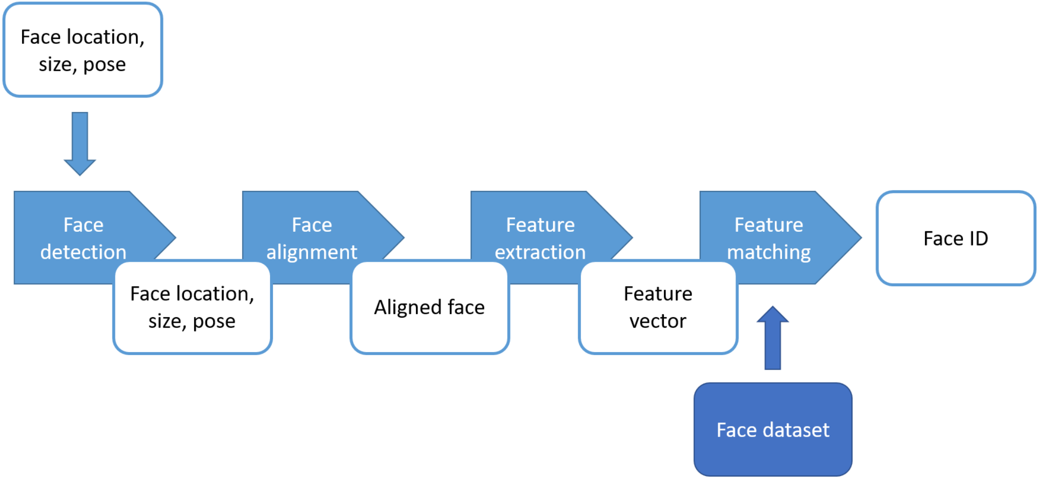
cvdazzle.com

Adversarial glasses

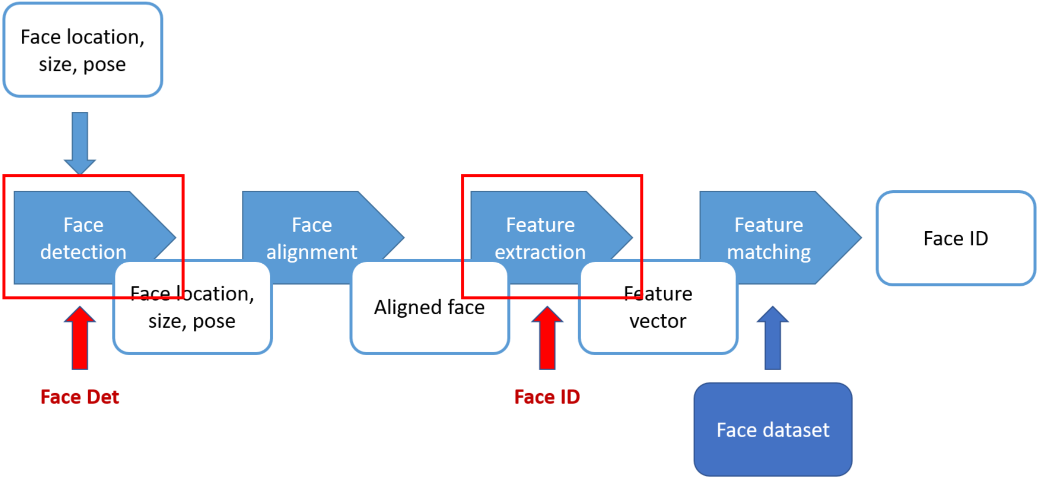
[18] Feng R. et al. "Facilitating fashion camouflage art." 2013

[19] Sharif M. et al. "Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition." 2016

Face processing pipeline



Face processing pipeline



Face detection: MTCNN[20]

- Unlike modern and heavy detectors based on Faster RCNN and YOLO the MTCNN detector is quite shallow $\Rightarrow$ smaller perception field, harder to change the detection conclusion

[20] Zhang K. et al. "Joint face detection and alignment using multitask cascaded convolutional networks." 2016

Face detection: MTCNN[20]

- Unlike modern and heavy detectors based on Faster RCNN and YOLO the MTCNN detector is quite shallow ⇒ smaller perception field, harder to change the detection conclusion
- MTCNN is cascade-based: in the beginning the rough approximation is provided (P-Net), then its tuning (R-Net and O-Net) is performed

[20] Zhang K. et al. "Joint face detection and alignment using multitask cascaded convolutional networks." 2016

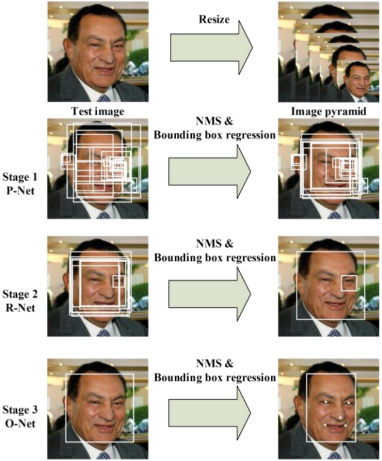
Face detection: MTCNN[20]

- Unlike modern and heavy detectors based on Faster RCNN and YOLO the MTCNN detector is quite shallow ⇒ smaller perception field, harder to change the detection conclusion
- MTCNN is cascade-based: in the beginning the rough approximation is provided (P-Net), then its tuning (R-Net and O-Net) is performed
- Based on our experiments, the most appropriate place for attack is the first P-Net and its classification loss (not bounding boxes or key points regression losses)

[20] Zhang K. et al. "Joint face detection and alignment using multitask cascaded convolutional networks." 2016

- Unlike modern and heavy detectors based on Faster RCNN and YOLO the MTCNN detector is quite shallow ⇒ smaller perception field, harder to change the detection conclusion
- MTCNN is cascade-based: in the beginning the rough approximation is provided (P-Net), then its tuning (R-Net and O-Net) is performed
- Based on our experiments, the most appropriate place for attack is the first P-Net and its classification loss (not bounding boxes or key points regression losses)

MTCNN scheme



[20] Zhang K. et al. "Joint face detection and alignment using multitask cascaded convolutional networks." 2016

Adversarial attack on MTCNN face detector

- EOT: Gaussian noise, patch size, brightness, batch of different face images

Adversarial attack on MTCNN face detector

- EOT: Gaussian noise, patch size, brightness, batch of different face images
- TV loss: used, NPS: not used

Adversarial attack on MTCNN face detector

- EOT: Gaussian noise, patch size, brightness, batch of different face images
- TV loss: used, NPS: not used
- Color adjustment: push the color to be the black one ($x_{i,j} = 1$) $\Rightarrow$ new additive loss part: $L_{BLK}(x) = \sum_{i,j}(1 - x_{i,j})$

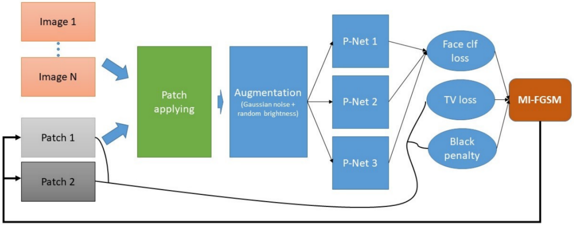
Adversarial attack on MTCNN face detector

- EOT: Gaussian noise, patch size, brightness, batch of different face images
- TV loss: used, NPS: not used
- Color adjustment: push the color to be the black one ($x_{i,j} = 1$) $\Rightarrow$ new additive loss part: $L_{BLK}(x) = \sum_{i,j}(1 - x_{i,j})$
- MI-FGSM as the optimizer

Adversarial attack on MTCNN face detector

- EOT: Gaussian noise, patch size, brightness, batch of different face images
- TV loss: used, NPS: not used
- Color adjustment: push the color to be the black one ($x_{i,j} = 1$) $\Rightarrow$ new additive loss part: $L_{BLK}(x) = \sum_{i,j}(1 - x_{i,j})$
- MI-FGSM as the optimizer

MTCNN adversarial attack



Adversarial attack on MTCNN face detector

- ℓ_0-based optimization: two versions of adversarial patches

Adversarial attack on MTCNN face detector

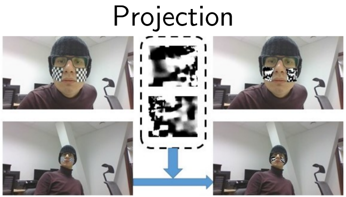
- ℓ_0-based optimization: two versions of adversarial patches
  1. two distinct patches on cheeks
  2. the whole medicine mask

Adversarial attack on MTCNN face detector

- ℓ_0-based optimization: two versions of adversarial patches
  1. two distinct patches on cheeks
  2. the whole medicine mask
- MTCNN has small perceptive field $\Rightarrow$ patches are not semantical (unlike for FaceID, see next)

Adversarial attack on MTCNN face detector

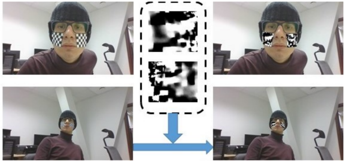
- ℓ_0-based optimization: two versions of adversarial patches
  1. two distinct patches on cheeks
  2. the whole medicine mask
- MTCNN has small perceptive field $\Rightarrow$ patches are not semantical (unlike for FaceID, see next)
- Need to estimate the local affine projections parameters based on the prepared special grid



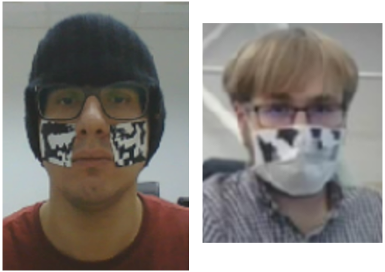
Adversarial attack on MTCNN face detector

- ℓ_0-based optimization: two versions of adversarial patches
  1. two distinct patches on cheeks
  2. the whole medicine mask
- MTCNN has small perceptive field $\Rightarrow$ patches are not semantical (unlike for FaceID, see next)
- Need to estimate the local affine projections parameters based on the prepared special grid

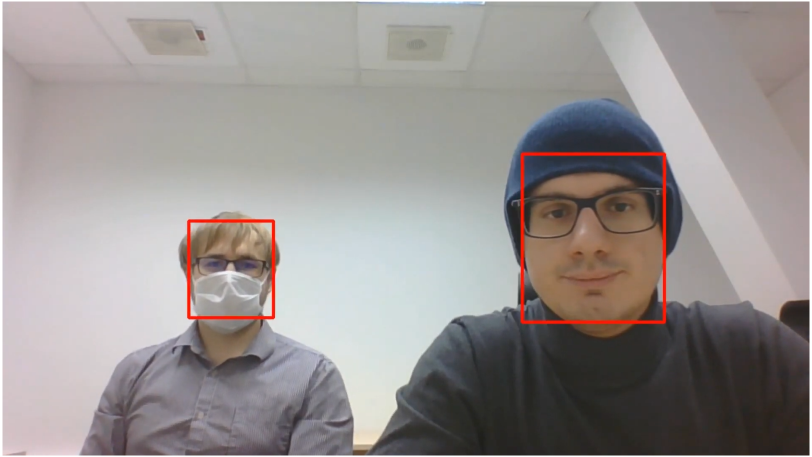
Projection



Patches



Details: paper[21] (IEEE-2019) and video[22].



[21]Kaziakhmedov E. et al. "Real-world attack on MTCNN face detection system." 2019

[22]https://www.youtube.com/watch?v=0Y700IS8bxs

FaceID: ArcFace[23]

- For face ID adversarial attack the best public face ID system was chosen: ArcFace

[23] Deng J. et al. "Arcface: Additive angular margin loss for deep face recognition." 2018

FaceID: ArcFace[23]

- For face ID adversarial attack the best public face ID system was chosen: ArcFace
- Main idea of ArcFace — to use angle margin (aligned with cosine similarity)

[23] Deng J. et al. "Arcface: Additive angular margin loss for deep face recognition." 2018

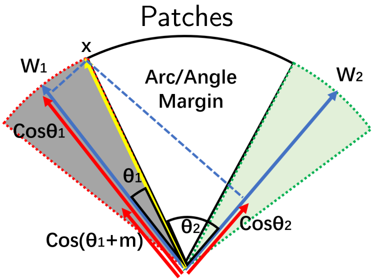
FaceID: ArcFace[23]

- For face ID adversarial attack the best public face ID system was chosen: ArcFace
- Main idea of ArcFace — to use angle margin (aligned with cosine similarity)
- Huge training dataset (MS1M) and deep CNN (ResNet-100) are used

[23] Deng J. et al. "Arcface: Additive angular margin loss for deep face recognition." 2018

FaceID: ArcFace[23]

- For face ID adversarial attack the best public face ID system was chosen: ArcFace
- Main idea of ArcFace — to use angle margin (aligned with cosine similarity)
- Huge training dataset (MS1M) and deep CNN (ResNet-100) are used



[23] Deng J. et al. "Arcface: Additive angular margin loss for deep face recognition." 2018

Adversarial attack on ArcFace face ID

- EOT: Different patch projection parameters, single face image

Adversarial attack on ArcFace face ID

- EOT: Different patch projection parameters, single face image
- TV loss: used, NPS: not used, Color adjustment: not used

Adversarial attack on ArcFace face ID

- EOT: Different patch projection parameters, single face image
- TV loss: used, NPS: not used, Color adjustment: not used
- Additive similarity loss to work in open-set setting: $L_{sim}(x, x_{gt}) = cos(emb(x), emb(x_{gt}))$, where x_{gt} — template image for the person, $emb(x)$ — feature vector of x

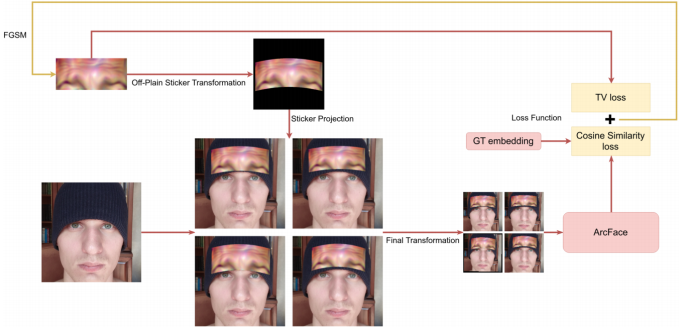
Adversarial attack on ArcFace face ID

- EOT: Different patch projection parameters, single face image
- TV loss: used, NPS: not used, Color adjustment: not used
- Additive similarity loss to work in open-set setting: $L_{sim}(x, x_{gt}) = cos(emb(x), emb(x_{gt}))$, where x_{gt} — template image for the person, $emb(x)$ — feature vector of x
- MI-FGSM as the optimizer

Adversarial attack on ArcFace face ID

- EOT: Different patch projection parameters, single face image
- TV loss: used, NPS: not used, Color adjustment: not used
- Additive similarity loss to work in open-set setting: $L_{sim}(x, x_{gt}) = cos(emb(x), emb(x_{gt}))$, where x_{gt} — template image for the person, $emb(x)$ — feature vector of x
- MI-FGSM as the optimizer

MTCNN adversarial attack



Adversarial attack on ArcFace face ID

- ℓ_0-based optimization: color patch on the forehead

[24] Jaderberg M. et al. "Spatial transformer networks." 2015

Adversarial attack on ArcFace face ID

- ℓ_0-based optimization: color patch on the forehead
- Deep NN $\Rightarrow$ large perception field $\Rightarrow$ patch is semantical

[24] Jaderberg M. et al. "Spatial transformer networks." 2015

Adversarial attack on ArcFace face ID

- ℓ_0-based optimization: color patch on the forehead
- Deep NN $\Rightarrow$ large perception field $\Rightarrow$ patch is semantical
- Nonlinear off-plane projection: $(x, y, 0) \rightarrow (x', y, z'), z' = a \cdot x'^2$

[24] Jaderberg M. et al. "Spatial transformer networks." 2015

Adversarial attack on ArcFace face ID

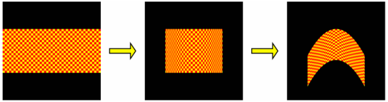
- ℓ_0-based optimization: color patch on the forehead
- Deep NN $\Rightarrow$ large perception field $\Rightarrow$ patch is semantical
- Nonlinear off-plane projection: $(x, y, 0) \rightarrow (x', y, z'), z' = a \cdot x'^2$
- All image transformation done by differentiable Spatial Transformer Layer[24]

[24] Jaderberg M. et al. "Spatial transformer networks." 2015

Adversarial attack on ArcFace face ID

- ℓ_0-based optimization: color patch on the forehead
- Deep NN $\Rightarrow$ large perception field $\Rightarrow$ patch is semantical
- Nonlinear off-plane projection: $(x, y, 0) \rightarrow (x', y, z'), z' = a \cdot x'^2$
- All image transformation done by differentiable Spatial Transformer Layer[24]

Off-plane projection

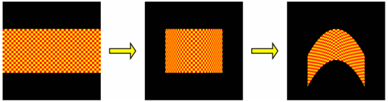


$$x' = a \cdot \left(|x| \cdot \sqrt{x^2 + \frac{1}{4 \cdot a^2}} + \frac{1}{4 \cdot a^2} \cdot \ln \left(|x| + \sqrt{x^2 + \frac{1}{4 \cdot a^2}} \right) - \frac{1}{4 \cdot a^2} \cdot \ln \left(\frac{1}{2 \cdot a} \right) \right)$$

[24] Jaderberg M. et al. "Spatial transformer networks." 2015

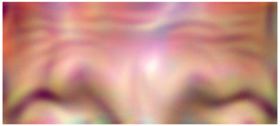
- ℓ_0-based optimization: color patch on the forehead
- Deep NN $\Rightarrow$ large perception field $\Rightarrow$ patch is semantical
- Nonlinear off-plane projection: $(x, y, 0) \to (x', y, z'), z' = a \cdot x'^2$
- All image transformation done by differentiable Spatial Transformer Layer[24]

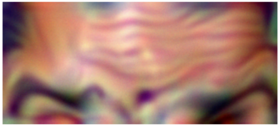
Off-plane projection



$$x' = a \cdot \left(|x| \cdot \sqrt{x^2 + \frac{1}{4 \cdot a^2}} + \frac{1}{4 \cdot a^2} \cdot \ln \left(|x| + \sqrt{x^2 + \frac{1}{4 \cdot a^2}} \right) - \frac{1}{4 \cdot a^2} \cdot \ln \left(\frac{1}{2 \cdot a} \right) \right)$$

Semantical patch examples



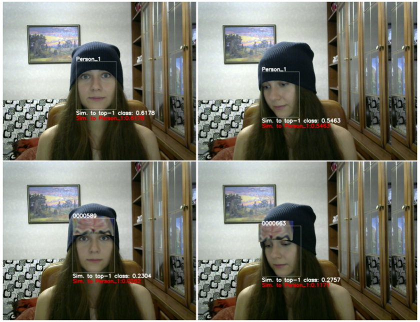


[24] Jaderberg M. et al. "Spatial transformer networks." 2015

Due to the better projection procedure and richer color information, the attack is robust to rotations and brightness variation

Frontal face (advhat: no)
Similarity to origin: 0.61

Frontal face (advhat: yes)
Similarity to origin: 0.02
Similarity to other: 0.23

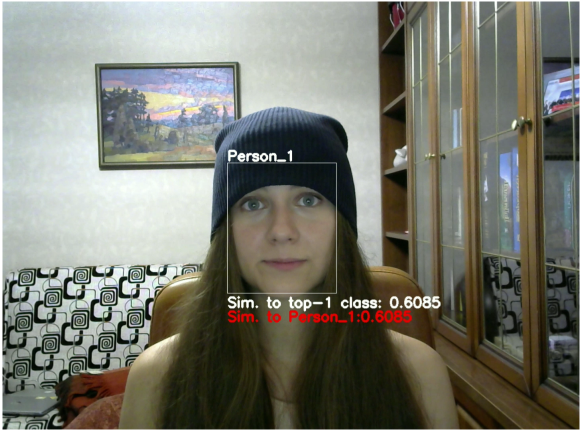


Rotated face (advhat: no)
Similarity to origin: 0.54

Rotated face (advhat: yes)
Similarity to origin: 0.11
Similarity to other: 0.27

Adversarial attack on ArcFace face ID: outcome

Details: paper[25] (ICPR-2020) and video[26].



[25] Komkov S. et al. "Advhat: Real-world adversarial attack on arcface face id system." 2019

[26] `https://www.youtube.com/watch?v=a4iNg0wWBsQ`

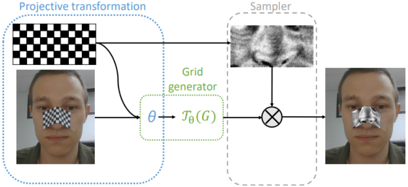
Adversarial attack on ArcFace face ID: grayscale patch[27] (IEEE-2019)

- Combination of two previous approaches:
  - Grayscale color loss adjustment
  - Local affine grid projection

[27] Pautov M. et al. "On adversarial patches: real-world attack on ArcFace-100 face recognition system." 2019

- Combination of two previous approaches:
  - Grayscale color loss adjustment
  - Local affine grid projection
- Patch is also semantical

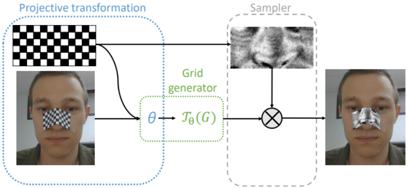
Projection



[27] Pautov M. et al. "On adversarial patches: real-world attack on ArcFace-100 face recognition system." 2019

- Combination of two previous approaches:
  - Grayscale color loss adjustment
  - Local affine grid projection
- Patch is also semantical

Projection



Patches

[27] Pautov M. et al. "On adversarial patches: real-world attack on ArcFace-100 face recognition system." 2019

FaceID adversarial defense in real world[29]

- Almost all of the real world attacks are patch-based
  - Proposal: **A**dversarial **T**raining (AT)[28] in the pixel space with patch-based augmentation

[28] Goodfellow I. et al. "Explaining and harnessing adversarial examples." 2014
[29] Wu T. et al. "Defending Against Physically Realizable Attacks on Image Classification." 2019

FaceID adversarial defense in real world[29]

- Almost all of the real world attacks are patch-based
  - Proposal: **A**dversarial **T**raining (AT)[28] in the pixel space with patch-based augmentation
- AT decoupling:
  1. Best (=max loss) location of gray patch by:
     - Exhaustive search
     - Max gradient locations w.r.t. input

[28] Goodfellow I. et al. "Explaining and harnessing adversarial examples." 2014

[29] Wu T. et al. "Defending Against Physically Realizable Attacks on Image Classification." 2019

FaceID adversarial defense in real world[29]

- Almost all of the real world attacks are patch-based
  - Proposal: **A**dversarial **T**raining (AT)[28] in the pixel space with patch-based augmentation
- AT decoupling:
  1. Best (=max loss) location of gray patch by:
     - Exhaustive search
     - Max gradient locations w.r.t. input
  2. PGD inside this patch

[28] Goodfellow I. et al. "Explaining and harnessing adversarial examples." 2014

[29] Wu T. et al. "Defending Against Physically Realizable Attacks on Image Classification." 2019

FaceID adversarial defense in real world[29]

- Almost all of the real world attacks are patch-based
  - Proposal: **A**dversarial **T**raining (AT)[28] in the pixel space with patch-based augmentation
- AT decoupling:
  1. Best (=max loss) location of gray patch by:
     - Exhaustive search
     - Max gradient locations w.r.t. input
  2. PGD inside this patch

- Common training procedure:

$$\min_\theta \mathbb{E}_{x,y}[L(\theta, x, y)]$$

- Adversarial Training:

$$\min_\theta \mathbb{E}_{x,y}[\max_{r \in \Delta} L(\theta, x + r, y)]$$

[28] Goodfellow I. et al. "Explaining and harnessing adversarial examples." 2014
[29] Wu T. et al. "Defending Against Physically Realizable Attacks on Image Classification." 2019

FaceID adversarial defense in real world[29]

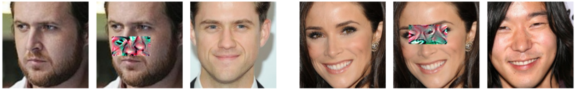
- Almost all of the real world attacks are patch-based
  - Proposal: **A**dversarial **T**raining (AT)[28] in the pixel space with patch-based augmentation
- AT decoupling:
  1. Best (=max loss) location of gray patch by:
     - Exhaustive search
     - Max gradient locations w.r.t. input
  2. PGD inside this patch

- Common training procedure:

$$\min_{\theta} \mathbb{E}_{x,y}[L(\theta, x, y)]$$

- Adversarial Training:

$$\min_{\theta} \mathbb{E}_{x,y}[\max_{r \in \Delta} L(\theta, x + r, y)]$$



[28] Goodfellow I. et al. "Explaining and harnessing adversarial examples." 2014

[29] Wu T. et al. "Defending Against Physically Realizable Attacks on Image Classification." 2019

Black-box face restoration

- Black-box model M: $M(x) = y$, where
  - x — input image of the face
  - y — its feature representation (embedding)

[30] Mai G. et al. "On the reconstruction of face images from deep face templates." 2018
[31] Schroff F. et al. "Facenet: A unified embedding for face recognition and clustering." 2015.

Black-box face restoration

- Black-box model M: $M(x) = y$, where
  - x — input image of the face
  - y — its feature representation (embedding)
- **Task**: to recover x' to preserve the person identity

[30] Mai G. et al. "On the reconstruction of face images from deep face templates." 2018
[31] Schroff F. et al. "Facenet: A unified embedding for face recognition and clustering." 2015.

Black-box face restoration

- Black-box model M: $M(x) = y$, where
  - x — input image of the face
  - y — its feature representation (embedding)
- **Task**: to recover x' to preserve the person identity
- **Prior art**: using reconstruction MSE and perceptual metrics / GANs (NBNet[30])

[30] Mai G. et al. "On the reconstruction of face images from deep face templates." 2018
[31] Schroff F. et al. "Facenet: A unified embedding for face recognition and clustering." 2015.

Black-box face restoration

- Black-box model M: $M(x) = y$, where
  - x — input image of the face
  - y — its feature representation (embedding)
- **Task**: to recover x' to preserve the person identity
- **Prior art**: using reconstruction MSE and perceptual metrics / GANs (NBNet[30])
- **Our approach**: use zero-order optimization to find such x' so as FaceID $M(x') \approx M(x)$

[30] Mai G. et al. "On the reconstruction of face images from deep face templates." 2018
[31] Schroff F. et al. "Facenet: A unified embedding for face recognition and clustering." 2015.

Black-box face restoration

- Black-box model M: $M(x) = y$, where
  - x — input image of the face
  - y — its feature representation (embedding)
- **Task**: to recover x' to preserve the person identity
- **Prior art**: using reconstruction MSE and perceptual metrics / GANs (NBNet[30])
- **Our approach**: use zero-order optimization to find such x' so as FaceID $M(x') \approx M(x)$
  - Use as M the public SotA in FaceID: ArcFace

[30] Mai G. et al. "On the reconstruction of face images from deep face templates." 2018

[31] Schroff F. et al. "Facenet: A unified embedding for face recognition and clustering." 2015.

Black-box face restoration

- Black-box model M: $M(x) = y$, where
  - x — input image of the face
  - y — its feature representation (embedding)
- **Task**: to recover x' to preserve the person identity
- **Prior art**: using reconstruction MSE and perceptual metrics / GANs (NBNet[30])
- **Our approach**: use zero-order optimization to find such x' so as FaceID $M(x') \approx M(x)$
  - Use as M the public SotA in FaceID: ArcFace
  - Test by using independent critic: FaceNET[31]

[30] Mai G. et al. "On the reconstruction of face images from deep face templates." 2018

[31] Schroff F. et al. "Facenet: A unified embedding for face recognition and clustering." 2015.

Black-box face restoration

- Black-box model M: $M(x) = y$, where
  - x — input image of the face
  - y — its feature representation (embedding)
- **Task**: to recover x' to preserve the person identity
- **Prior art**: using reconstruction MSE and perceptual metrics / GANs (NBNet[30])
- **Our approach**: use zero-order optimization to find such x' so as FaceID $M(x') \approx M(x)$
  - Use as M the public SotA in FaceID: ArcFace
  - Test by using independent critic: FaceNET[31]
  - Similarity loss: $1 - cos(M(x'), M(x))$

[30] Mai G. et al. "On the reconstruction of face images from deep face templates." 2018

[31] Schroff F. et al. "Facenet: A unified embedding for face recognition and clustering." 2015.

Black-box face restoration

- Black-box model M: $M(x) = y$, where
  - x — input image of the face
  - y — its feature representation (embedding)
- **Task**: to recover x' to preserve the person identity
- **Prior art**: using reconstruction MSE and perceptual metrics / GANs (NBNet[30])
- **Our approach**: use zero-order optimization to find such x' so as FaceID $M(x') \approx M(x)$
  - Use as M the public SotA in FaceID: ArcFace
  - Test by using independent critic: FaceNET[31]
  - Similarity loss: $1 - cos(M(x'), M(x))$
  - Additional term: $(||M(x')||_2 - ||M(x)||_2)^2$

[30] Mai G. et al. "On the reconstruction of face images from deep face templates." 2018
[31] Schroff F. et al. "Facenet: A unified embedding for face recognition and clustering." 2015.

Black-box face restoration

- Black-box model M: $M(x) = y$, where
  - x — input image of the face
  - y — its feature representation (embedding)
- **Task**: to recover x' to preserve the person identity
- **Prior art**: using reconstruction MSE and perceptual metrics / GANs (NBNet[30])
- **Our approach**: use zero-order optimization to find such x' so as FaceID $M(x') \approx M(x)$
  - Use as M the public SotA in FaceID: ArcFace
  - Test by using independent critic: FaceNET[31]
  - Similarity loss: $1 - cos(M(x'), M(x))$
  - Additional term: $(||M(x')||_2 - ||M(x)||_2)^2$
- **Main difficulty**: huge search space in pixel domain

[30] Mai G. et al. "On the reconstruction of face images from deep face templates." 2018
[31] Schroff F. et al. "Facenet: A unified embedding for face recognition and clustering." 2015.

Black-box face restoration

- Black-box model M: $M(x) = y$, where
  - x — input image of the face
  - y — its feature representation (embedding)
- **Task**: to recover x' to preserve the person identity
- **Prior art**: using reconstruction MSE and perceptual metrics / GANs (NBNet[30])
- **Our approach**: use zero-order optimization to find such x' so as FaceID $M(x') \approx M(x)$
  - Use as M the public SotA in FaceID: ArcFace
  - Test by using independent critic: FaceNET[31]
  - Similarity loss: $1 - cos(M(x'), M(x))$
  - Additional term: $(||M(x')||_2 - ||M(x)||_2)^2$

- **Main difficulty**: huge search space in pixel domain
- **Solution**: to use prior knowledge about face — 2D Gaussians

$$G(x, y) = A \cdot e^{\frac{(x-x_0)^2}{2\sigma_1^2} + \frac{(y-y_0)^2}{2\sigma_2^2}}$$

[30] Mai G. et al. "On the reconstruction of face images from deep face templates." 2018

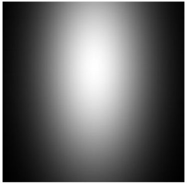
[31] Schroff F. et al. "Facenet: A unified embedding for face recognition and clustering." 2015.

Black-box face restoration

- Black-box model M: $M(x) = y$, where
  - x — input image of the face
  - y — its feature representation (embedding)
- **Task**: to recover x' to preserve the person identity
- **Prior art**: using reconstruction MSE and perceptual metrics / GANs (NBNet[30])
- **Our approach**: use zero-order optimization to find such x' so as FaceID $M(x') \approx M(x)$
  - Use as M the public SotA in FaceID: ArcFace
  - Test by using independent critic: FaceNET[31]
  - Similarity loss: $1 - cos(M(x'), M(x))$
  - Additional term: $(||M(x')||_2 - ||M(x)||_2)^2$

- **Main difficulty**: huge search space in pixel domain
- **Solution**: to use prior knowledge about face — 2D Gaussians

$$G(x, y) = A \cdot e^{\frac{(x-x_0)^2}{2\sigma_1^2} + \frac{(y-y_0)^2}{2\sigma_2^2}}$$



$$(x_0, y_0, \sigma_1, \sigma_2, A) = (56, 72, 22, 42, 1)$$

[30] Mai G. et al. "On the reconstruction of face images from deep face templates." 2018

[31] Schroff F. et al. "Facenet: A unified embedding for face recognition and clustering." 2015.

Black-box face restoration: successful tricks

- Even prior info about face is not enough

Black-box face restoration: successful tricks

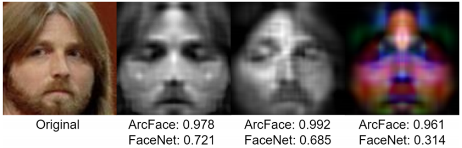
- Even prior info about face is not enough
- **Trick1**: Use vertical face symmetry $\Rightarrow$ use only half of the face to search

Black-box face restoration: successful tricks

- Even prior info about face is not enough
- **Trick1**: Use vertical face symmetry $\Rightarrow$ use only half of the face to search
- **Trick2**: For identity preservation usually no need in color $\Rightarrow$ use only a single color channel instead of 3

Black-box face restoration: successful tricks

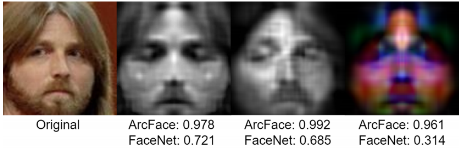
- Even prior info about face is not enough
- **Trick1**: Use vertical face symmetry $\Rightarrow$ use only half of the face to search
- **Trick2**: For identity preservation usually no need in color $\Rightarrow$ use only a single color channel instead of 3



Symmetrical, non-symmetrical, color restoration

Black-box face restoration: successful tricks

- Even prior info about face is not enough
- **Trick1**: Use vertical face symmetry $\Rightarrow$ use only half of the face to search
- **Trick2**: For identity preservation usually no need in color $\Rightarrow$ use only a single color channel instead of 3



Symmetrical, non-symmetrical, color restoration

- **Initialization**: What to use as the starting point?

Black-box face restoration: successful tricks

- Even prior info about face is not enough
- **Trick1**: Use vertical face symmetry $\Rightarrow$ use only half of the face to search
- **Trick2**: For identity preservation usually no need in color $\Rightarrow$ use only a single color channel instead of 3

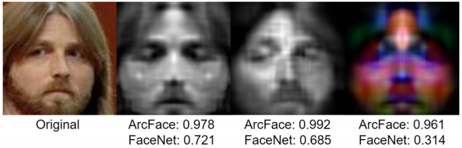
Original	ArcFace: 0.978 FaceNet: 0.721	ArcFace: 0.992 FaceNet: 0.685	ArcFace: 0.961 FaceNet: 0.314

Symmetrical, non-symmetrical, color restoration

- **Initialization**: What to use as the starting point?
- **Common approach**: to use other face (can be biased)

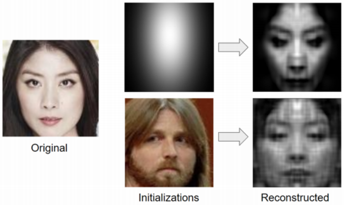
Black-box face restoration: successful tricks

- Even prior info about face is not enough
- **Trick1**: Use vertical face symmetry ⇒ use only half of the face to search
- **Trick2**: For identity preservation usually no need in color ⇒ use only a single color channel instead of 3



Symmetrical, non-symmetrical, color restoration

- **Initialization**: What to use as the starting point?
- **Common approach**: to use other face (can be biased)
- **Our approach**: optimal Gaussian blob (additional loss term is needed)



Algorithm

Algorithm 1 Face recovery algorithm

INPUT: target face embedding y, black-box model M, loss function L, $N_{queries}$

1: $X \leftarrow 0$
2: Initialize G_0
3: **for** $i \leftarrow 0$ to $N_{queries}$ **do**:
4: Allocate image batch $\mathbf{X}$
5: Sample batch $\mathbf{G}$ of random gaussians
6: $\mathbf{X}_j = X + G_0 + \mathbf{G}_j$
7: $\mathbf{y}' = M(\mathbf{X})$
8: $\text{ind} = \text{argmin}\Big(L(\mathbf{y}'_i, y)\Big)$
9: $X \leftarrow X + \mathbf{G}_{\text{ind}}$
10: $G_0 \leftarrow 0.99 \cdot G_0$
11: $i \leftarrow i + \text{batchsize}$
12: **end for**
13: $X \leftarrow X + G_0$

OUTPUT: reconstructed face X

Black-box face restoration: algorithm and results

Algorithm

Algorithm 1 Face recovery algorithm

INPUT: target face embedding y, black-box model M, loss function L, $N_{queries}$

$X \leftarrow 0$
Initialize G_0
for $i \leftarrow 0$ to $N_{queries}$ **do**:
  Allocate image batch $\mathbf{X}$
  Sample batch $\mathbf{G}$ of random gaussians
  $\mathbf{X}_j = X + G_0 + \mathbf{G}_j$
  $\mathbf{y}' = M(\mathbf{X})$
  $\text{ind} = \text{argmin}\left(L(\mathbf{y}'_i, y)\right)$
  $X \leftarrow X + \mathbf{G}_{\text{ind}}$
  $G_0 \leftarrow 0.99 \cdot G_0$
  $i \leftarrow i + \text{batchsize}$
end for
$X \leftarrow X + G_0$

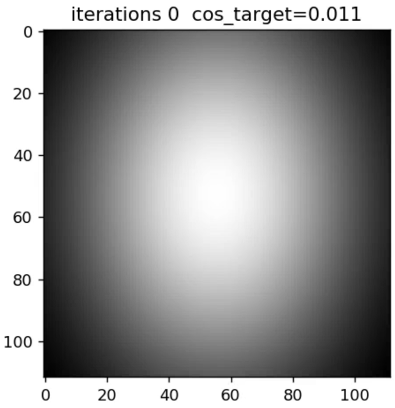
OUTPUT: reconstructed face X

Results

Our method:						
ArcFace:	0.97	0.97	0.94	0.97	0.85	0.73
FaceNet:	0.70	0.75	0.72	0.78	0.38	-0.09
NBNet (WB):						
ArcFace:	0.17	0.21	0.12	0.26	0.06	0.09
FaceNet:	0.02	0.32	0.25	0.46	-0.01	0.35
NBNet (RGB):						
ArcFace:	0.28	0.46	0.34	0.54	0.12	0.21
FaceNet:	0.59	0.53	0.44	0.74	0.18	0.41
Original:						

Black-box face restoration: outcome

Details: paper[32] (ECCV-2020) and video presentation[33].



[32] Razzhigaev A. et al. "Black-Box Face Recovery from Identity Features." 2020
[33] https://www.youtube.com/watch?v=s0rTcqRTw2A

Takeway notes

- CNNs for now are much better than human expert in controlled conditions

Takeway notes

- CNNs for now are much better than human expert in controlled conditions
- CNNs are unstable w.r.t. its input

Takeway notes

- CNNs for now are much better than human expert in controlled conditions
- CNNs are unstable w.r.t. its input
- Digital $\rightarrow$ physical domain attack translation is hard

Takeway notes

- CNNs for now are much better than human expert in controlled conditions
- CNNs are unstable w.r.t. its input
- Digital $\rightarrow$ physical domain attack translation is hard
- But even the most successful face ID systems can be fooled by a simple grayscale patch from common printer

Takeway notes

- CNNs for now are much better than human expert in controlled conditions
- CNNs are unstable w.r.t. its input
- Digital $\rightarrow$ physical domain attack translation is hard
- But even the most successful face ID systems can be fooled by a simple grayscale patch from common printer
- ℓ_0-based local attack + TV loss are the must

Takeway notes

- CNNs for now are much better than human expert in controlled conditions
- CNNs are unstable w.r.t. its input
- Digital $\to$ physical domain attack translation is hard
- But even the most successful face ID systems can be fooled by a simple grayscale patch from common printer
- ℓ_0-based local attack + TV loss are the must
- Need to use projection schemes allowing gradient backpropagation

Takeway notes

- CNNs for now are much better than human expert in controlled conditions
- CNNs are unstable w.r.t. its input
- Digital $\rightarrow$ physical domain attack translation is hard
- But even the most successful face ID systems can be fooled by a simple grayscale patch from common printer
- ℓ_0-based local attack + TV loss are the must
- Need to use projection schemes allowing gradient backpropagation
- Adversarial training in practice (or certified robustness in theory) can help to defense

Takeway notes

- CNNs for now are much better than human expert in controlled conditions
- CNNs are unstable w.r.t. its input
- Digital $\rightarrow$ physical domain attack translation is hard
- But even the most successful face ID systems can be fooled by a simple grayscale patch from common printer
- ℓ_0-based local attack + TV loss are the must
- Need to use projection schemes allowing gradient backpropagation
- Adversarial training in practice (or certified robustness in theory) can help to defense
- Face image can be restored even in black-box setting using its embedding

Присоединяйтесь к нам!

Кого мы ждем:

- *Выпускники аспирантуры 2019-2021 годов*
- *Победители и призеры таких международных соревнований как ICPC, IMC, CTF, Kaggle, IMO, IOI, ICHO, IPHO etc.*
- *Техническое образование (информационные технологии, математика, физика, радиотехника, системы связи, информационная безопасность и др.)*
- *Английский на уровне "intermediate" и выше*

Наши направления:

- *Nonlinear algorithm development*
- *Wireless communication technologies*
- *Computer Vision with Deep Learning*
- *Math Library optimization*
- *Automatic program repair*
- *Compiler optimizations*
- *Automatic speech recognition*
- *AI databases and AI enabled systems*
- *Distributed and Parallel software*
- *Image/Video signal processing*
- *Software engineering and innovation*
- *Automated machine learning & Model optimization*
- *Computer architecture*

Резюме можно выслать на почту: *rrihr@huawei.com*

Москва Санкт-Петербург Нижний Новгород Новосибирск

https://career.huawei.ru/rri/

Inspired by science to connect the world!

Looking forward to seeing your application

Thank you!