Состязательное обучение в реальном мире

Петюшко А. А.
petyushko.alexander1@huawei.com

МГУ им. М.В.Ломоносова, механико-математический факультет, кафедра МаТИС
Huawei, Intelligence Systems and Data Science Technnology Center

29 мая 2020 г.

План доклада

1. Intelligence Systems and Data Science Technology Center

План доклада

1. Intelligence Systems and Data Science Technology Center
2. Потрясающие успехи CHC в компьютерном зрении

План доклада

1. Intelligence Systems and Data Science Technology Center
2. Потрясающие успехи СНС в компьютерном зрении
3. (Не) устойчивость СНС в компьютерном зрении

План доклада

1. Intelligence Systems and Data Science Technology Center
2. Потрясающие успехи CHC в компьютерном зрении
3. (Не) устойчивость CHC в компьютерном зрении
4. Классификация состязательных атак

План доклада

1. Intelligence Systems and Data Science Technology Center
2. Потрясающие успехи СНС в компьютерном зрении
3. (Не) устойчивость СНС в компьютерном зрении
4. Классификация состязательных атак
5. Методы состязательных атак в цифровой области

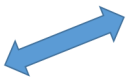
План доклада

1. Intelligence Systems and Data Science Technology Center
2. Потрясающие успехи СНС в компьютерном зрении
3. (Не) устойчивость СНС в компьютерном зрении
4. Классификация состязательных атак
5. Методы состязательных атак в цифровой области
6. Методы состязательных атак в реальном мире

План доклада

1. Intelligence Systems and Data Science Technology Center
2. Потрясающие успехи СНС в компьютерном зрении
3. (Не) устойчивость СНС в компьютерном зрении
4. Классификация состязательных атак
5. Методы состязательных атак в цифровой области
6. Методы состязательных атак в реальном мире
7. Состязательные атаки на системы детекции и распознавания лиц в реальном мире

Intelligence Systems and Data Science Technology Center: научное сотрудничество



ISDSTC

Санкт-Петербургский государственный университет

МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ М.В.ЛОМОНОСОВА

SHARE

В 2019 году в Huawei стартовала образовательная программа **SHARE**: Школа опережающего научного образования Хуавэй (School of Huawei Advanced Research Education).

SHARE

В 2019 году в Huawei стартовала образовательная программа **SHARE**: Школа опережающего научного образования Хуавэй (School of Huawei Advanced Research Education).
Intelligence Systems and Data Science Technology Center проводит занятия в МГУ им. М.В. Ломоносова:

SHARE

В 2019 году в Huawei стартовала образовательная программа **SHARE**: Школа опережающего научного образования Хуавэй (School of Huawei Advanced Research Education).
Intelligence Systems and Data Science Technology Center проводит занятия в МГУ им. М.В. Ломоносова:

- 2 года длится программа;

SHARE

В 2019 году в Huawei стартовала образовательная программа **SHARE**: Школа опережающего научного образования Хуавэй (School of Huawei Advanced Research Education).
Intelligence Systems and Data Science Technology Center проводит занятия в МГУ им. М.В. Ломоносова:

- 2 года длится программа;
- 12 полносеместровых курсов;

SHARE

В 2019 году в Huawei стартовала образовательная программа **SHARE**: Школа опережающего научного образования Хуавэй (School of Huawei Advanced Research Education).
Intelligence Systems and Data Science Technology Center проводит занятия в МГУ им. М.В. Ломоносова:

- 2 года длится программа;
- 12 полносеместровых курсов;
- 2 направления:

SHARE

В 2019 году в Huawei стартовала образовательная программа **SHARE**: Школа опережающего научного образования Хуавэй (School of Huawei Advanced Research Education).
Intelligence Systems and Data Science Technology Center проводит занятия в МГУ им. М.В. Ломоносова:

- 2 года длится программа;
- 12 полносеместровых курсов;
- 2 направления:
  - Специализация “Компьютерное зрение и машинное обучение”;

SHARE

В 2019 году в Huawei стартовала образовательная программа **SHARE**: Школа опережающего научного образования Хуавэй (School of Huawei Advanced Research Education).
Intelligence Systems and Data Science Technology Center проводит занятия в МГУ им. М.В. Ломоносова:

- 2 года длится программа;
- 12 полносеместровых курсов;
- 2 направления:
  - Специализация “Компьютерное зрение и машинное обучение”;
  - Специализация “Большие данные и теория информации”.

Источники доклада

- Курс “Современное компьютерное зрение”[1];

[1] https://github.com/mlcoursemm/cvcoursemm2019autumn
[2] https://www.youtube.com/watch?v=tKPSqKYuLTc

Источники доклада

- Курс “Современное компьютерное зрение”[1];
- Открытая лекция на Фестивале Науке-2019[2];

[1] https://github.com/mlcoursemm/cvcoursemm2019autumn
[2] https://www.youtube.com/watch?v=tKPSqKYuLTc

Источники доклада

- Курс “Современное компьютерное зрение”[1];
- Открытая лекция на Фестивале Науке-2019[2];
- Внутренние обсуждения внутри Intelligence Systems and Data Science Technology Center, Moscow Research Center, Huawei.

[1] https://github.com/mlcoursemm/cvcoursemm2019autumn
[2] https://www.youtube.com/watch?v=tKPSqKYuLTc

Развитие нейросетей

Начиная с 1943 года, когда впервые была предложена математическая формализация МакКалоком и Питтсом понятия “искусственного нейрона”, нейросети становились[3]:

[3] Image credits: https://arxiv.org/abs/1409.4842

Развитие нейросетей

Начиная с 1943 года, когда впервые была предложена математическая формализация МакКалоком и Питтсом понятия “искусственного нейрона”, нейросети становились[3]:

- Объемнее (содержали больше параметров),

[3] Image credits: https://arxiv.org/abs/1409.4842

Развитие нейросетей

Начиная с 1943 года, когда впервые была предложена математическая формализация МакКалоком и Питтсом понятия “искусственного нейрона”, нейросети становились[3]:

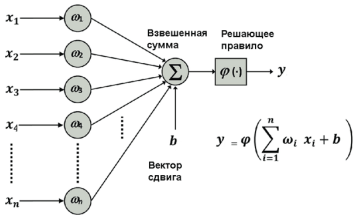
- Объемнее (содержали больше параметров),
- Глубже (содержали больше блоков вычислений),

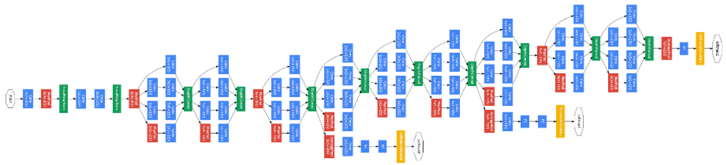
[3] Image credits: https://arxiv.org/abs/1409.4842

Развитие нейросетей

Начиная с 1943 года, когда впервые была предложена математическая формализация МакКалоком и Питтсом понятия “искусственного нейрона”, нейросети становились[3]:

- Объемнее (содержали больше параметров),
- Глубже (содержали больше блоков вычислений),
- Лучше! (более правильно решали поставленные перед ними задачи)





[3]Image credits: `https://arxiv.org/abs/1409.4842`

Сверточные нейросети[4]

- Для работы с фотографиями и видео лучше всего подходят сверточные нейронные сети (CHC),

[4] Image credits: https://adeshpande3.github.io/, https://stepupanalytics.com

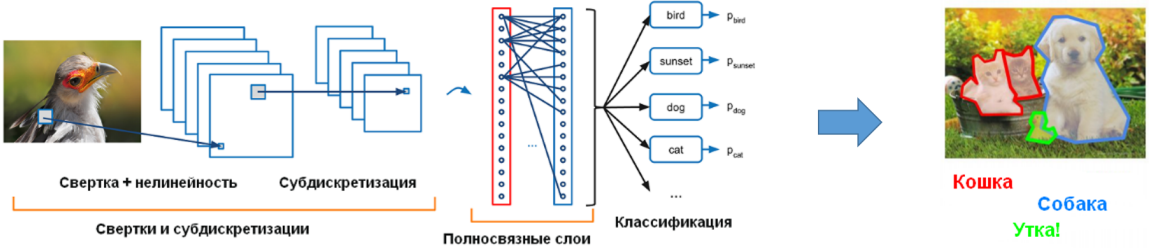
Сверточные нейросети[4]

- Для работы с фотографиями и видео лучше всего подходят сверточные нейронные сети (СНС),
- Например, позволяют выделять объекты и определять их класс,

[4] Image credits: https://adeshpande3.github.io/, https://stepupanalytics.com

Сверточные нейросети[4]

- Для работы с фотографиями и видео лучше всего подходят сверточные нейронные сети (СНС),
- Например, позволяют выделять объекты и определять их класс,
- Ну и отвечают на главный вопрос – кошка или собака?



[4]Image credits: https://adeshpande3.github.io/, https://stepupanalytics.com

Интересные вопросы

Давайте разберемся, так ли уж хороши сверточные нейросети, действительно ли оправдано все то внимание, которое им уделяют?

[5] Image credit: `https://spectrum.ieee.org`

Интересные вопросы

Давайте разберемся, так ли уж хороши сверточные нейросети, действительно ли оправдано все то внимание, которое им уделяют?

Вопрос1

Как сейчас соотносится качество распознавания человеком и СНС для известных баз данных?

[5] Image credit: `https://spectrum.ieee.org`

Интересные вопросы

Давайте разберемся, так ли уж хороши сверточные нейросети, действительно ли оправдано все то внимание, которое им уделяют?

Вопрос1

Как сейчас соотносится качество распознавания человеком и СНС для известных баз данных?

Вопрос2

Насколько устойчивы СНС по отношению к входным данным? Легко ли их сломать?

[5] Image credit: `https://spectrum.ieee.org`

Интересные вопросы

Давайте разберемся, так ли уж хороши сверточные нейросети, действительно ли оправдано все то внимание, которое им уделяют?

Вопрос1

Как сейчас соотносится качество распознавания человеком и CHC для известных баз данных?

Вопрос2

Насколько устойчивы CHC по отношению к входным данным? Легко ли их сломать?

CNN vs Human[5]

[5]Image credit: https://spectrum.ieee.org

Человек или СНС?

ImageNet[6] (1000-классовая база данных изображений)

- Top-5 ошибка для человека[7]: 5.1%
- Top-5 ошибка для СНС[8]: 2.0%

[6] http://www.image-net.org/

[7] http://karpathy.github.io/2014/09/02/what-i-learned-from-competing-against-a-convnet-on-imagenet/

[8] Touvron, Hugo, et al. "Fixing the train-test resolution discrepancy." 2019

[9] http://vis-www.cs.umass.edu/lfw/

[10] Kumar, Neeraj, et al. "Attribute and simile classifiers for face verification." 2009

[11] Deng, Jiankang, et al. "Arcface: Additive angular margin loss for deep face recognition." 2018

Человек или СНС?

ImageNet[6] (1000-классовая база данных изображений)

- Top-5 ошибка для человека[7]: 5.1%
- Top-5 ошибка для СНС[8]: 2.0%

Labeled Faces in the Wild[9] (база данных лиц)

- Ошибка верификации для человека[10]: 2.47%
- Ошибка верификации для СНС[11]: 0.17%

[6] http://www.image-net.org/

[7] http://karpathy.github.io/2014/09/02/what-i-learned-from-competing-against-a-convnet-on-imagenet/

[8] Touvron, Hugo, et al. "Fixing the train-test resolution discrepancy." 2019

[9] http://vis-www.cs.umass.edu/lfw/

[10] Kumar, Neeraj, et al. "Attribute and simile classifiers for face verification." 2009

[11] Deng, Jiankang, et al. "Arcface: Additive angular margin loss for deep face recognition." 2018

Человек или СНС?

ImageNet[6] (1000-классовая база данных изображений)

- Top-5 ошибка для человека[7]: 5.1%
- Top-5 ошибка для СНС[8]: 2.0%

Labeled Faces in the Wild[9] (база данных лиц)

- Ошибка верификации для человека[10]: 2.47%
- Ошибка верификации для СНС[11]: 0.17%

LFW

[6] http://www.image-net.org/

[7] http://karpathy.github.io/2014/09/02/what-i-learned-from-competing-against-a-convnet-on-imagenet/

[8] Touvron, Hugo, et al. "Fixing the train-test resolution discrepancy." 2019

[9] http://vis-www.cs.umass.edu/lfw/

[10] Kumar, Neeraj, et al. "Attribute and simile classifiers for face verification." 2009

[11] Deng, Jiankang, et al. "Arcface: Additive angular margin loss for deep face recognition." 2018

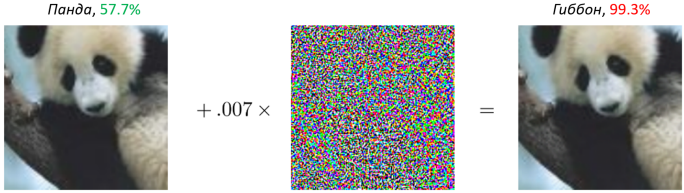
Такие неустойчивые СНС

- Можно внести практически незаметные для глаза человека возмущения во входные данные, которые, тем не менее, полностью поменяют выход нейронной сети

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

Такие неустойчивые СНС

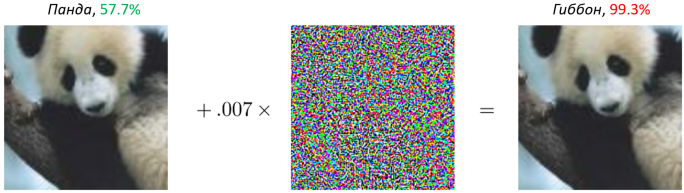
- Можно внести практически незаметные для глаза человека возмущения во входные данные, которые, тем не менее, полностью поменяют выход нейронной сети
- Например, результат классификации с “панды” поменяется на “гиббона”[12]



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

Такие неустойчивые СНС

- Можно внести практически незаметные для глаза человека возмущения во входные данные, которые, тем не менее, полностью поменяют выход нейронной сети
- Например, результат классификации с “панды” поменяется на “гиббона”[12]



Такое возмущение называется **состязательной атакой** (adversarial attack)

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

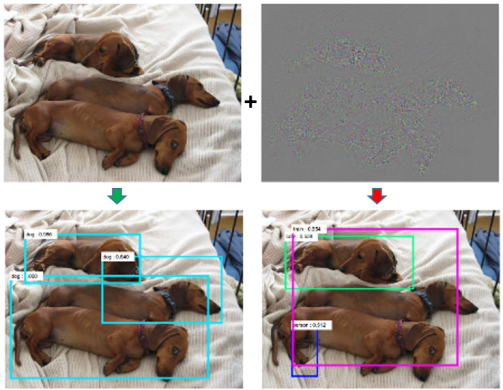
Атака СНС, предназначенных для сегментации или обнаружения

- Можно атаковать также СНС, которые не предназначены для классификации — например, для обнаружения и сегментации изображений[13]

[13] Xie, Cihang, et al. "Adversarial examples for semantic segmentation and object detection." 2017

Атака СНС, предназначенных для сегментации или обнаружения

- Можно атаковать также СНС, которые не предназначены для классификации — например, для обнаружения и сегментации изображений[13]



[13] Xie, Cihang, et al. "Adversarial examples for semantic segmentation and object detection." 2017

Атака нейросетей, не предназначенных для изображений

- Можно атаковать даже НС, которые вообще не работают с изображениями — например, НС для вопросно-ответных систем (QA, question answering systems)[14]

[14] Jia, Robin, and Percy Liang. "Adversarial examples for evaluating reading comprehension systems." 2017

Атака нейросетей, не предназначенных для изображений

- Можно атаковать даже НС, которые вообще не работают с изображениями — например, НС для вопросно-ответных систем (QA, question answering systems)[14]

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

[14] Jia, Robin, and Percy Liang. "Adversarial examples for evaluating reading comprehension systems." 2017

Одна из главных причин существования атак

- Одна из основных причин такого поведения CHC на похожих изображениях —- неустойчивость CHC

[15] Image credit: `https://secml.github.io/`

[16] Fawzi, Alhussein, Seyed-Mohsen Moosavi-Dezfooli, and Pascal Frossard. “Robustness of classifiers: from adversarial to random noise.” 2016

Одна из главных причин существования атак

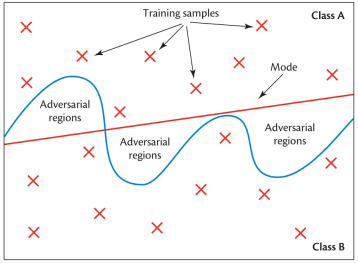
- Одна из основных причин такого поведения СНС на похожих изображениях —- неустойчивость СНС
- А именно, разделяющие границы классификатора часто проходят очень близко к обучающим данным, и легко “заступить” за такую границу[15,16]

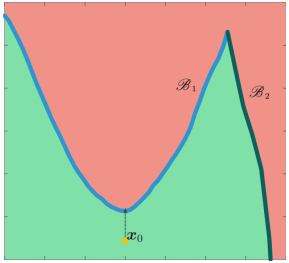
[15] Image credit: `https://secml.github.io/`

[16] Fawzi, Alhussein, Seyed-Mohsen Moosavi-Dezfooli, and Pascal Frossard. “Robustness of classifiers: from adversarial to random noise.” 2016

Одна из главных причин существования атак

- Одна из основных причин такого поведения CHC на похожих изображениях —- неустойчивость CHC
- А именно, разделяющие границы классификатора часто проходят очень близко к обучающим данным, и легко “заступить” за такую границу[15,16]





[15] Image credit: `https://secml.github.io/`

[16] Fawzi, Alhussein, Seyed-Mohsen Moosavi-Dezfooli, and Pascal Frossard. “Robustness of classifiers: from adversarial to random noise.” 2016

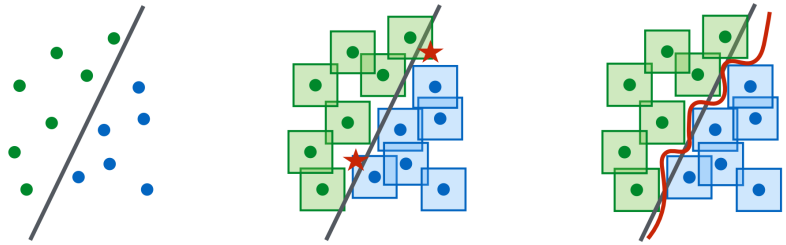
Простой метод защиты

- Поскольку можно обмануть CHC путем небольшого пиксельного возмущения, то почему бы во время обучения для каждого обучающего примера не добавлять и всю его попиксельную окрестность (по некоторой норме, например, ℓ_∞)[17]

[17] Madry, Aleksander, et al. "Towards deep learning models resistant to adversarial attacks." 2017

Простой метод защиты

- Поскольку можно обмануть СНС путем небольшого пиксельного возмущения, то почему бы во время обучения для каждого обучающего примера не добавлять и всю его попиксельную окрестность (по некоторой норме, например, ℓ_∞)[17]



[17] Madry, Aleksander, et al. "Towards deep learning models resistant to adversarial attacks." 2017

Простой, но не работающий метод защиты

- Предположим, что исходная картинка размера 100×100 пикселей, 3 цвета RGB

Простой, но не работающий метод защиты

- Предположим, что исходная картинка размера 100×100 пикселей, 3 цвета RGB
- Предположим, что наш глаз не сильно различает колебания цвета пикселей в 2 градации (из 256): в каждой точке для каждого цвета можем позволить ± 1 значение

Простой, но не работающий метод защиты

- Предположим, что исходная картинка размера 100×100 пикселей, 3 цвета RGB
- Предположим, что наш глаз не сильно различает колебания цвета пикселей в 2 градации (из 256): в каждой точке для каждого цвета можем позволить ± 1 значение
- Тогда для каждого обучающего примера нужно добавить следующее количество его пиксельных соседей:

$$2^{3 \times 100 \times 100} = 2^{30000} = (2^{10})^{3000} \approx (10^3)^{3000} = 10^{9000}$$

Простой, но не работающий метод защиты

- Предположим, что исходная картинка размера 100×100 пикселей, 3 цвета RGB
- Предположим, что наш глаз не сильно различает колебания цвета пикселей в 2 градации (из 256): в каждой точке для каждого цвета можем позволить ± 1 значение
- Тогда для каждого обучающего примера нужно добавить следующее количество его пиксельных соседей:

$$2^{3 \times 100 \times 100} = 2^{30000} = (2^{10})^{3000} \approx (10^3)^{3000} = 10^{9000}$$

- Это гораздо больше числа атомов в видимой части Вселенной (10^{80})!

Простой, но не работающий метод защиты

- Предположим, что исходная картинка размера 100×100 пикселей, 3 цвета RGB
- Предположим, что наш глаз не сильно различает колебания цвета пикселей в 2 градации (из 256): в каждой точке для каждого цвета можем позволить ± 1 значение
- Тогда для каждого обучающего примера нужно добавить следующее количество его пиксельных соседей:

$$2^{3 \times 100 \times 100} = 2^{30000} = (2^{10})^{3000} \approx (10^3)^{3000} = 10^{9000}$$

- Это гораздо больше числа атомов в видимой части Вселенной (10^{80})!
- В общем, не очень реалистично

Работающий метод защиты

- Давайте не перебирать всю окрестность обучающего примера, а брать только те точки из окрестности, которые ближе всего к разделяющей поверхности

[18] Goodfellow, Ian J., Jonathon Shlens, and Christian Szegedy. "Explaining and harnessing adversarial examples." 2014

Работающий метод защиты

- Давайте не перебирать всю окрестность обучающего примера, а брать только те точки из окрестности, которые ближе всего к разделяющей поверхности
- Такой метод обучения называется состязательным (adversarial training)[18]

[18] Goodfellow, Ian J., Jonathon Shlens, and Christian Szegedy. "Explaining and harnessing adversarial examples." 2014

Работающий метод защиты

- Давайте не перебирать всю окрестность обучающего примера, а брать только те точки из окрестности, которые ближе всего к разделяющей поверхности
- Такой метод обучения называется состязательным (adversarial training)[18]

Плюсы состязательного обучения

- Не нужно перебирать всю окрестность огромной мощности
- В целом, защищает от метода нахождения состязательных примеров

[18] Goodfellow, Ian J., Jonathon Shlens, and Christian Szegedy. "Explaining and harnessing adversarial examples." 2014

Работающий метод защиты

- Давайте не перебирать всю окрестность обучающего примера, а брать только те точки из окрестности, которые ближе всего к разделяющей поверхности
- Такой метод обучения называется состязательным (adversarial training)[18]

Плюсы состязательного обучения

- Не нужно перебирать всю окрестность огромной мощности
- В целом, защищает от метода нахождения состязательных примеров

Минусы состязательного обучения

- Процедура нахождения хороших состязательных примеров работает медленно (гораздо медленнее одного градиентного шага)
- Защищает **только** от того метода нахождения состязательных примеров, который исопльзовался в состязательном обучении

[18] Goodfellow, Ian J., Jonathon Shlens, and Christian Szegedy. "Explaining and harnessing adversarial examples." 2014

Состязательные атаки: необходимые обозначения

- Пусть $x \in B = [0,1]^{C \times M \times N}$ — входная картинка $C \times M \times N$, где C — количество цветов (1 для ч/б, 3 для RGB)
- y_{gt} — правильный класс для x
- θ — параметры CHC-классификатора
- $L(\theta, x, y_{gt})$ — функция потерь
- $f(x)$ — выход классификатора (распознанный класс); при обучении мы добиваемся равенства $f(x) = y_{gt}$

Состязательные атаки: необходимые обозначения

- Пусть $x \in B = [0,1]^{C \times M \times N}$ — входная картинка $C \times M \times N$, где C — количество цветов (1 для ч/б, 3 для RGB)
- y_{gt} — правильный класс для x
- θ — параметры CHC-классификатора
- $L(\theta, x, y_{gt})$ — функция потерь
- $f(x)$ — выход классификатора (распознанный класс); при обучении мы добиваемся равенства $f(x) = y_{gt}$
- $r \in B = [0,1]^{C \times M \times N}$ — аддитивная добавка ко входу x

Состязательная атака: формулировка

Цель состязательной атаки

Поменять выход классификатора f на неправильный путем добавления минимального по некоторой норме (на практике используются ℓ_0, ℓ_1, ℓ_2 и ℓ_∞ — обозначим через ℓ_p) возмущения r, а именно:

Состязательная атака: формулировка

Цель состязательной атаки

Поменять выход классификатора f на неправильный путем добавления минимального по некоторой норме (на практике используются ℓ_0, ℓ_1, ℓ_2 и ℓ_∞ — обозначим через ℓ_p) возмущения r, а именно: минимизировать $||r||_p$ т.ч.

1. $f(x) = y_{gt}$
2. $f(x + r) \neq y_{gt}$
3. $x + r \in B$

Состязательное обучение: формулировка

В обозначениях выше обычное обучение можно сформулировать как

Обучение на примерах

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

Состязательное обучение: формулировка

В обозначениях выше обычное обучение можно сформулировать как

Обучение на примерах

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

В состязательном обучении мы сначала генерируем (например, каким-нибудь методом атаки) самый сложный пример из некоторой окрестности Δ входного примера (например, по ℓ_p-норме), а уже затем минимизируем по параметрам нейросети:

Состязательное обучение: формулировка

В обозначениях выше обычное обучение можно сформулировать как

Обучение на примерах

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

В состязательном обучении мы сначала генерируем (например, каким-нибудь методом атаки) самый сложный пример из некоторой окрестности Δ входного примера (например, по ℓ_p-норме), а уже затем минимизируем по параметрам нейросети:

Состязательное обучение

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

Нормы ℓ_p: напоминание

Напомним наиболее употребительные нормы ℓ_p для $x = (x_1, \ldots, x_n) \in \mathbb{R}^n$:

Нормы ℓ_p: напоминание

Напомним наиболее употребительные нормы ℓ_p для $x = (x_1, \ldots, x_n) \in \mathbb{R}^n$:

- ℓ_2: $||x||_2 = \sqrt{\sum_{i=1}^{n} x_i^2}$

Нормы ℓ_p: напоминание

Напомним наиболее употребительные нормы ℓ_p для $x = (x_1, \ldots, x_n) \in \mathbb{R}^n$:

- ℓ_2: $||x||_2 = \sqrt{\sum_{i=1}^n x_i^2}$
- ℓ_1: $||x||_1 = \sum_{i=1}^n |x_i|$

Нормы ℓ_p: напоминание

Напомним наиболее употребительные нормы ℓ_p для $x = (x_1, \ldots, x_n) \in \mathbb{R}^n$:

- ℓ_2: $||x||_2 = \sqrt{\sum_{i=1}^n x_i^2}$
- ℓ_1: $||x||_1 = \sum_{i=1}^n |x_i|$
- ℓ_∞: $||x||_\infty = \max_i |x_i|$

Нормы ℓ_p: напоминание

Напомним наиболее употребительные нормы ℓ_p для $x = (x_1, \ldots, x_n) \in \mathbb{R}^n$:

- ℓ_2: $||x||_2 = \sqrt{\sum_{i=1}^n x_i^2}$
- ℓ_1: $||x||_1 = \sum_{i=1}^n |x_i|$
- ℓ_∞: $||x||_\infty = \max_i |x_i|$
- ℓ_0: $||x||_0 = \sum_{i=1}^n \mathbf{1}_{x_i \neq 0}$

Нормы ℓ_p: напоминание

Напомним наиболее употребительные нормы ℓ_p для $x = (x_1, \ldots, x_n) \in \mathbb{R}^n$:

- ℓ_2: $||x||_2 = \sqrt{\sum_{i=1}^n x_i^2}$
- ℓ_1: $||x||_1 = \sum_{i=1}^n |x_i|$
- ℓ_∞: $||x||_\infty = \max_i |x_i|$
- ℓ_0: $||x||_0 = \sum_{i=1}^n \mathbf{1}_{x_i \neq 0}$

Замечание. Для $0 < p < 1$ норма ℓ_p, для которой $||x||_p = \left(\sum_{i=1}^n |x_i|^p\right)^{1/p}$, не является нормой

Классификация состязательных атак

По цели атаки

- Ненаправленная (untargeted): нужно просто сменить ответ классификатора
- Направленная (targeted): нужно сменить на заранее определенный класс y_t

Классификация состязательных атак

По цели атаки

- Ненаправленная (untargeted): нужно просто сменить ответ классификатора
- Направленная (targeted): нужно сменить на заранее определенный класс y_t

По осведомленности атакующего

- Открытая (white-box): атакующий знает все о классификаторе (архитектуру и веса)
- Закрытая (black-box): атакующий имеет частичную информацию о классификаторе (обычно только информацию о выходе)

Классификация состязательных атак

По цели атаки

- Ненаправленная (untargeted): нужно просто сменить ответ классификатора
- Направленная (targeted): нужно сменить на заранее определенный класс y_t

По осведомленности атакующего

- Открытая (white-box): атакующий знает все о классификаторе (архитектуру и веса)
- Закрытая (black-box): атакующий имеет частичную информацию о классификаторе (обычно только информацию о выходе)

По условию применения

- Цифровая (digital): атака на фотографию
- Реальная (real-world): атака на реальный объект

Классификация состязательных атак

По универсальности

- Зависимая от входа (input-aware): возмущение r зависит от входа x
- Универсальная (universal): возмущение r работает для любого входа x

Классификация состязательных атак

По универсальности

- Зависимая от входа (input-aware): возмущение r зависит от входа x
- Универсальная (universal): возмущение r работает для любого входа x

По переносимости

- Непереносимая (non-transferable): атака работает только для узкого класса классификаторов
- Переносимая (transferable): атака работает для широкого класса классификаторов (но при этом может быть не универсальной)

- Наиболее сложная атака — направленная закрытая реальная универсальная переносимая атака
- Для простоты будем рассматривать открытые атаки

Эффективность состязательной атаки

Введем простой критерий успешности (success) $S(A, Z)$ алгоритма A состязательной атаки $r_A(x)$ на множестве $Z \ni (x^i, y^i_{gt})$:

Эффективность состязательной атаки

Введем простой критерий успешности (success) $S(A, Z)$ алгоритма A состязательной атаки $r_A(x)$ на множестве $Z \ni (x^i, y^i_{gt})$:

- В случае ненаправленной атаки:

$$S(A, Z) = \frac{\sum_i \mathbf{1}\{f(x^i) = y^i_{gt}\} \cdot \mathbf{1}\{f(x^i + r_A(x^i)) \neq y^i_{gt}\}}{\sum_i \mathbf{1}\{f(x^i) = y^i_{gt}\}}$$

Эффективность состязательной атаки

Введем простой критерий успешности (success) $S(A, Z)$ алгоритма A состязательной атаки $r_A(x)$ на множестве $Z \ni (x^i, y^i_{gt})$:

- В случае ненаправленной атаки:

$$S(A, Z) = \frac{\sum_i \mathbf{1}\{f(x^i) = y^i_{gt}\} \cdot \mathbf{1}\{f(x^i + r_A(x^i)) \neq y^i_{gt}\}}{\sum_i \mathbf{1}\{f(x^i) = y^i_{gt}\}}$$

- В случае направленной атаки на класс y_t:

$$S(A, Z, y_t) = \frac{\sum_i \mathbf{1}\{f(x^i) = y^i_{gt}\} \cdot \mathbf{1}\{f(x^i + r_A(x^i)) = y_t\}}{\sum_i \mathbf{1}\{f(x^i) = y^i_{gt}\}}$$

Эффективность состязательной атаки

Введем простой критерий успешности (success) $S(A, Z)$ алгоритма A состязательной атаки $r_A(x)$ на множестве $Z \ni (x^i, y^i_{gt})$:

- В случае ненаправленной атаки:

$$S(A, Z) = \frac{\sum_i \mathbf{1}\{f(x^i) = y^i_{gt}\} \cdot \mathbf{1}\{f(x^i + r_A(x^i)) \neq y^i_{gt}\}}{\sum_i \mathbf{1}\{f(x^i) = y^i_{gt}\}}$$

- В случае направленной атаки на класс y_t:

$$S(A, Z, y_t) = \frac{\sum_i \mathbf{1}\{f(x^i) = y^i_{gt}\} \cdot \mathbf{1}\{f(x^i + r_A(x^i)) = y_t\}}{\sum_i \mathbf{1}\{f(x^i) = y^i_{gt}\}}$$

Замечание. Очевидно, что $S(A, Z, y_t) \leq S(A, Z)$

Предтеча состязательных атак

- Изначально устойчивость CHC изучалась с точки зрения адекватной реакции на разные входы

[19] Nguyen, Anh, Jason Yosinski, and Jeff Clune. "Deep neural networks are easily fooled: High confidence predictions for unrecognizable images." 2014

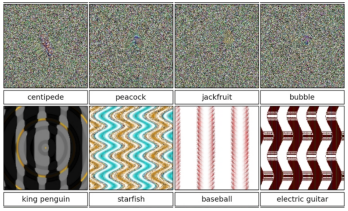
Предтеча состязательных атак

- Изначально устойчивость CHC изучалась с точки зрения адекватной реакции на разные входы
- Выяснилось, что существуют примеры (структурированные или нет), которые на выходе CHC могут давать с большой вероятностью любой класс

[19] Nguyen, Anh, Jason Yosinski, and Jeff Clune. "Deep neural networks are easily fooled: High confidence predictions for unrecognizable images." 2014

Предтеча состязательных атак

- Изначально устойчивость CHC изучалась с точки зрения адекватной реакции на разные входы
- Выяснилось, что существуют примеры (структурированные или нет), которые на выходе CHC могут давать с большой вероятностью любой класс
- Такие примеры назывались “обманными изображениями”[19] (fooling images) и строились с помощью эволюционных алгоритмов



[19] Nguyen, Anh, Jason Yosinski, and Jeff Clune. “Deep neural networks are easily fooled: High confidence predictions for unrecognizable images.” 2014

Метод атаки: L-BFGS-B

- Первая предложенная атака[20] использовала ℓ_2-норму для ограничения атаки
- Рассматривалась направленная атака на класс $y_t \neq y_{gt}$
- Функционал для минимизации с ограничением $x + r \in B$, $c = const$:

$$c||r||_2 + L(\theta, x, y_t) \to \min_r$$

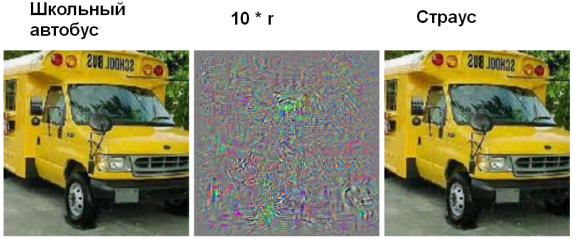
- Для оптимизации использовался метод L-BFGS-B[21] (**L**imited memory **B**royden–**F**letcher–**G**oldfarb–**S**hanno algorithm with **B**ox constraints) — квази-Ньютоновский метод минимизации с ограничением на память и на переменные
- В какой-то мере атака была переносима на другие архитектуры

[20] Szegedy, Christian, et al. "Intriguing properties of neural networks." 2013

[21] Byrd, Richard H., et al. "A limited memory algorithm for bound constrained optimization." 1995

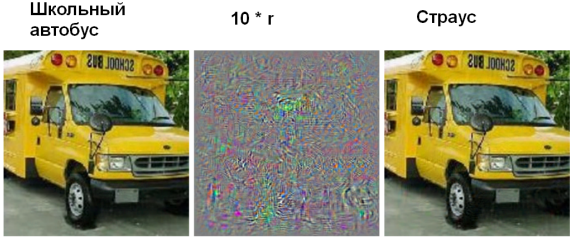
Метод атаки: L-BFGS-B

Пример работы:



Метод атаки: L-BFGS-B

Пример работы:



Переносимость:

	FC10(10^{-4})	FC10(10^{-2})	FC10(1)	FC100-100-10	FC200-200-10	AE400-10
FC10(10^{-4})	100%	11.7%	22.7%	2%	3.9%	2.7%
FC10(10^{-2})	87.1%	100%	35.2%	35.9%	27.3%	9.8%
FC10(1)	71.9%	76.2%	100%	48.1%	47%	34.4%
FC100-100-10	28.9%	13.7%	21.1%	100%	6.6%	2%
FC200-200-10	38.2%	14%	23.8%	20.3%	100%	2.7%
AE400-10	23.4%	16%	24.8%	9.4%	6.6%	100%

Метод атаки: FGSM

- Несмотря на хорошую реализацию, метод L-BFGS-B не так быстр и требует внешнего (по отношению к исследуемой СНС) оптимизатора

[22] Goodfellow, Ian J., Jonathon Shlens, and Christian Szegedy. "Explaining and harnessing adversarial examples." 2014

Метод атаки: FGSM

- Несмотря на хорошую реализацию, метод L-BFGS-B не так быстр и требует внешнего (по отношению к исследуемой СНС) оптимизатора
- **Предложение**: использовать линейную часть функции потерь в окрестности x и идти по градиенту — FGSM[22] (**F**ast **G**radient **S**ign **M**ethod):

$$r = \epsilon \cdot \text{sign}(\nabla_x L(\theta, x, y_t))$$

где $0 < \epsilon < 1$ — некоторая константа

[22] Goodfellow, Ian J., Jonathon Shlens, and Christian Szegedy. "Explaining and harnessing adversarial examples." 2014

Метод атаки: FGSM

- Несмотря на хорошую реализацию, метод L-BFGS-B не так быстр и требует внешнего (по отношению к исследуемой СНС) оптимизатора
- **Предложение**: использовать линейную часть функции потерь в окрестности x и идти по градиенту — FGSM[22] (**F**ast **G**radient **S**ign **M**ethod):

$$r = \epsilon \cdot \text{sign}(\nabla_x L(\theta, x, y_t))$$

где $0 < \epsilon < 1$ — некоторая константа

- **Напоминание**: для оптимизации весов СНС применяется метод обратного распространения ошибок, где берется градиент по весам СНС, т.е. $\nabla_\theta L(\theta, x, y_{gt})$

[22] Goodfellow, Ian J., Jonathon Shlens, and Christian Szegedy. "Explaining and harnessing adversarial examples." 2014

Метод атаки: FGSM

- Несмотря на хорошую реализацию, метод L-BFGS-B не так быстр и требует внешнего (по отношению к исследуемой СНС) оптимизатора
- **Предложение**: использовать линейную часть функции потерь в окрестности x и идти по градиенту — FGSM[22] (**F**ast **G**radient **S**ign **M**ethod):

$$r = \epsilon \cdot \text{sign}(\nabla_x L(\theta, x, y_t))$$

где $0 < \epsilon < 1$ — некоторая константа

- **Напоминание**: для оптимизации весов СНС применяется метод обратного распространения ошибок, где берется градиент по весам СНС, т.е. $\nabla_\theta L(\theta, x, y_{gt})$
- Теперь исследуется норма возмущения ℓ_∞ как наиболее близкая к тому, что использует человек

[22] Goodfellow, Ian J., Jonathon Shlens, and Christian Szegedy. "Explaining and harnessing adversarial examples." 2014

Метод атаки: I-FGSM (PGD)

- Часто линейная оценка окрестности функции достаточно грубая, и один шаг FGSM порой не приводит к хорошей атаке

[23] Kurakin, Alexey, Ian Goodfellow, and Samy Bengio. "Adversarial examples in the physical world." 2016

Метод атаки: I-FGSM (PGD)

- Часто линейная оценка окрестности функции достаточно грубая, и один шаг FGSM порой не приводит к хорошей атаке
- Для этого применяют итеративный метод I-FGSM[23] (**I**terative FGSM), который позволяет двигаться в сторону границы классификатора более точно

[23] Kurakin, Alexey, Ian Goodfellow, and Samy Bengio. "Adversarial examples in the physical world." 2016

Метод атаки: I-FGSM (PGD)

- Часто линейная оценка окрестности функции достаточно грубая, и один шаг FGSM порой не приводит к хорошей атаке
- Для этого применяют итеративный метод I-FGSM[23] (**I**terative FGSM), который позволяет двигаться в сторону границы классификатора более точно
- Если Π_B — проекция на B, то в случае ненаправленной атаки

$$x^{n+1} = \Pi_B(x^n + \operatorname{sign} \nabla_x L(\theta, x, y_{gt})), \quad x^0 = x$$

[23] Kurakin, Alexey, Ian Goodfellow, and Samy Bengio. "Adversarial examples in the physical world." 2016

Метод атаки: I-FGSM (PGD)

- Часто линейная оценка окрестности функции достаточно грубая, и один шаг FGSM порой не приводит к хорошей атаке
- Для этого применяют итеративный метод I-FGSM[23] (**I**terative FGSM), который позволяет двигаться в сторону границы классификатора более точно
- Если Π_B — проекция на B, то в случае ненаправленной атаки

$$x^{n+1} = \Pi_B(x^n + \text{sign}\,\nabla_x L(\theta, x, y_{gt})), \quad x^0 = x$$

- Если принять $||x - x_{adv}||_\infty \leq \epsilon$, то авторы предлагают делать $n = \min(256\epsilon + 4, 320\epsilon)$ шагов

[23] Kurakin, Alexey, Ian Goodfellow, and Samy Bengio. "Adversarial examples in the physical world." 2016

Метод атаки: I-FGSM (PGD)

- Часто линейная оценка окрестности функции достаточно грубая, и один шаг FGSM порой не приводит к хорошей атаке
- Для этого применяют итеративный метод I-FGSM[23] (**I**terative FGSM), который позволяет двигаться в сторону границы классификатора более точно
- Если Π_B — проекция на B, то в случае ненаправленной атаки

$$x^{n+1} = \Pi_B(x^n + \text{sign}\,\nabla_x L(\theta, x, y_{gt})), \quad x^0 = x$$

- Если принять $||x - x_{adv}||_\infty \leq \epsilon$, то авторы предлагают делать $n = \min(256\epsilon + 4, 320\epsilon)$ шагов
- Этот метод также называется PGD (**P**rojected **G**radient **D**escent)

[23] Kurakin, Alexey, Ian Goodfellow, and Samy Bengio. "Adversarial examples in the physical world." 2016

Метод атаки: MI-FGSM

- **Замечание**: Методы атаки все больше похожи на шаги оптимизатора

[24] Dong, Yinpeng, et al. "Boosting adversarial attacks with momentum." 2017

Метод атаки: MI-FGSM

- **Замечание**: Методы атаки все больше похожи на шаги оптимизатора
- **Идея**: давайте использовать сглаживание градиента — MI-FGSM[24] (**M**omentum I-FGSM)

[24] Dong, Yinpeng, et al. "Boosting adversarial attacks with momentum." 2017

Метод атаки: MI-FGSM

- **Замечание**: Методы атаки все больше похожи на шаги оптимизатора
- **Идея**: давайте использовать сглаживание градиента — MI-FGSM[24] (Momentum I-FGSM)

Algorithm 1 MI-FGSM

Input: A classifier f with loss function J; a real example $\boldsymbol{x}$ and ground-truth label y;
Input: The size of perturbation ϵ; iterations T and decay factor μ.
Output: An adversarial example $\boldsymbol{x}^*$ with $\|\boldsymbol{x}^* - \boldsymbol{x}\|_\infty \leq \epsilon$.

$\alpha = \epsilon/T$;
$\boldsymbol{g}_0 = 0$; $\boldsymbol{x}_0^* = \boldsymbol{x}$;
for $t = 0$ to $T - 1$ **do**
  Input $\boldsymbol{x}_t^*$ to f and obtain the gradient $\nabla_{\boldsymbol{x}} J(\boldsymbol{x}_t^*, y)$;
  Update $\boldsymbol{g}_{t+1}$ by accumulating the velocity vector in the gradient direction as

$$\boldsymbol{g}_{t+1} = \mu \cdot \boldsymbol{g}_t + \frac{\nabla_{\boldsymbol{x}} J(\boldsymbol{x}_t^*, y)}{\|\nabla_{\boldsymbol{x}} J(\boldsymbol{x}_t^*, y)\|_1}; \quad (6)$$

  Update $\boldsymbol{x}_{t+1}^*$ by applying the sign gradient as

$$\boldsymbol{x}_{t+1}^* = \boldsymbol{x}_t^* + \alpha \cdot \mathrm{sign}(\boldsymbol{g}_{t+1}); \quad (7)$$

end for
return $\boldsymbol{x}^* = \boldsymbol{x}_T^*$.

[24] Dong, Yinpeng, et al. "Boosting adversarial attacks with momentum." 2017

Сравнение FGSM-like атак

	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**

Метод атаки: DeepFool

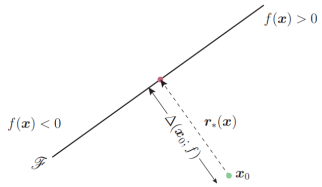
- **Идея**: проецировать точку x_0 на разделяющую поверхность

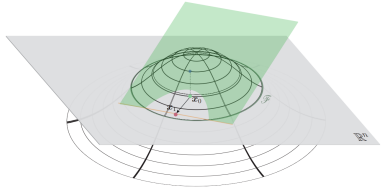
Метод атаки: DeepFool

- **Идея**: проецировать точку x_0 на разделяющую поверхность
- В случае линейного бинарного классификатора $\operatorname{sign} f(x) = \operatorname{sign}(w^T x + b)$:
  - Направление: $-\operatorname{sign} f(x_0) \frac{w}{||w||_2}$
  - Длина: $\frac{|f(x_0)|}{||w||_2}$
  - $\Rightarrow$ Атака: $r = -\frac{f(x_0)}{||w||_2^2} w$

Метод атаки: DeepFool

- **Идея**: проецировать точку x_0 на разделяющую поверхность
- В случае линейного бинарного классификатора $\operatorname{sign} f(x) = \operatorname{sign}(w^T x + b)$:
  - Направление: $-\operatorname{sign} f(x_0) \frac{w}{||w||_2}$
  - Длина: $\frac{|f(x_0)|}{||w||_2}$
  - $\Rightarrow$ Атака: $r = -\frac{f(x_0)}{||w||_2^2} w$
- В случае нелинейной разделяющей поверхности $f(x)$:
  - применяем формулу Тейлора: $f(x) \approx f(x_0) + \nabla f(x_0)(x - x_0)$
  - и подставляем в формулу для r выражение $w = \nabla f(x_0)$





Метод атаки: DeepFool

- Итеративный алгоритм DeepFool[25] для произвольного классификатора

Algorithm 1 DeepFool for binary classifiers

input: Image $\boldsymbol{x}$, classifier f.
output: Perturbation $\hat{\boldsymbol{r}}$.
Initialize $\boldsymbol{x}_0 \leftarrow \boldsymbol{x}$, $i \leftarrow 0$.
while $\operatorname{sign}(f(\boldsymbol{x}_i)) = \operatorname{sign}(f(\boldsymbol{x}_0))$ **do**
    $\boldsymbol{r}_i \leftarrow -\frac{f(\boldsymbol{x}_i)}{\|\nabla f(\boldsymbol{x}_i)\|_2^2} \nabla f(\boldsymbol{x}_i)$,
    $\boldsymbol{x}_{i+1} \leftarrow \boldsymbol{x}_i + \boldsymbol{r}_i$,
    $i \leftarrow i + 1$.
end while
return $\hat{\boldsymbol{r}} = \sum_i \boldsymbol{r}_i$.

- Существует естественное обобщение на случай многоклассового классификатора

[25] Moosavi-Dezfooli, Seyed-Mohsen, Alhussein Fawzi, and Pascal Frossard. "Deepfool: a simple and accurate method to fool deep neural networks." 2015

Метод атаки: JSMA

- JSMA[26] (**J**acobian-based **S**aliency **M**ap **A**ttack) — одна из первых ℓ_0-атак, когда важно количество задействованных в атаке пикселей, а не их значения

[26] Papernot, Nicolas, et al. "The limitations of deep learning in adversarial settings." 2015

Метод атаки: JSMA

- JSMA[26] (**J**acobian-based **S**aliency **M**ap **A**ttack) — одна из первых ℓ_0-атак, когда важно количество задействованных в атаке пикселей, а не их значения
- **Идея**: менять те пиксели, которые дают максимальный вклад в производную по входу для нужного класса для направленной атаки

[26] Papernot, Nicolas, et al. "The limitations of deep learning in adversarial settings." 2015

Метод атаки: JSMA

- JSMA[26] (**J**acobian-based **S**aliency **M**ap **A**ttack) — одна из первых ℓ_0-атак, когда важно количество задействованных в атаке пикселей, а не их значения
- **Идея**: менять те пиксели, которые дают максимальный вклад в производную по входу для нужного класса для направленной атаки
- Можно делать это итеративно, постепенно добавляя пиксели в область атаки r

[26] Papernot, Nicolas, et al. "The limitations of deep learning in adversarial settings." 2015

Метод атаки: JSMA

- JSMA[26] (**J**acobian-based **S**aliency **M**ap **A**ttack) — одна из первых ℓ_0-атак, когда важно количество задействованных в атаке пикселей, а не их значения
- **Идея**: менять те пиксели, которые дают максимальный вклад в производную по входу для нужного класса для направленной атаки
- Можно делать это итеративно, постепенно добавляя пиксели в область атаки r
- **Замечание**: $F(x)$ — выход SoftMax слоя, пиксели добавляются парами (так проще)

[26] Papernot, Nicolas, et al. "The limitations of deep learning in adversarial settings." 2015

Метод атаки: JSMA

- JSMA[26] (**J**acobian-based **S**aliency **M**ap **A**ttack) — одна из первых ℓ_0-атак, когда важно количество задействованных в атаке пикселей, а не их значения
- **Идея**: менять те пиксели, которые дают максимальный вклад в производную по входу для нужного класса для направленной атаки
- Можно делать это итеративно, постепенно добавляя пиксели в область атаки r
- **Замечание**: $F(x)$ — выход SoftMax слоя, пиксели добавляются парами (так проще)

Algorithm 3 Increasing pixel intensities saliency map
$\nabla \mathbf{F}(\mathbf{X})$ is the forward derivative, Γ the features still in the search space, and t the target class

Input: $\nabla \mathbf{F}(\mathbf{X})$, Γ, t
for each pair $(p, q) \in \Gamma$ **do**
  $\alpha = \sum_{i=p,q} \frac{\partial \mathbf{F}_t(\mathbf{X})}{\partial \mathbf{X}_i}$
  $\beta = \sum_{i=p,q} \sum_{j \neq t} \frac{\partial \mathbf{F}_j(\mathbf{X})}{\partial \mathbf{X}_i}$
  if $\alpha > 0$ and $\beta < 0$ and $-\alpha \times \beta >$ max **then**
    $p_1, p_2 \leftarrow p, q$
    $max \leftarrow -\alpha \times \beta$
  end if
end for
return p_1, p_2

[26] Papernot, Nicolas, et al. "The limitations of deep learning in adversarial settings." 2015

Метод атаки: One pixel

- Однопиксельная атака[27] — предельный случай ℓ_0-атаки

[27] Su, Jiawei, Danilo Vasconcellos Vargas, and Kouichi Sakurai. "One pixel attack for fooling deep neural networks." 2017

[28] Storn, Rainer, and Kenneth Price. "Differential evolution — a simple and efficient heuristic for global optimization over continuous spaces." 1997

Метод атаки: One pixel

- Однопиксельная атака[27] — предельный случай ℓ_0-атаки
- **Идея**: применить эволюционный алгоритм (дифференциальной эволюции[28])

[27] Su, Jiawei, Danilo Vasconcellos Vargas, and Kouichi Sakurai. "One pixel attack for fooling deep neural networks." 2017

[28] Storn, Rainer, and Kenneth Price. "Differential evolution — a simple and efficient heuristic for global optimization over continuous spaces." 1997

Метод атаки: One pixel

- Однопиксельная атака[27] — предельный случай ℓ_0-атаки
- **Идея**: применить эволюционный алгоритм (дифференциальной эволюции[28])
- Популяция состоит из 400 экземпляров, каждый из которых задается пятеркой: две координаты и три канала цвета

[27] Su, Jiawei, Danilo Vasconcellos Vargas, and Kouichi Sakurai. "One pixel attack for fooling deep neural networks." 2017

[28] Storn, Rainer, and Kenneth Price. "Differential evolution — a simple and efficient heuristic for global optimization over continuous spaces." 1997

Метод атаки: One pixel

- Однопиксельная атака[27] — предельный случай ℓ_0-атаки
- **Идея**: применить эволюционный алгоритм (дифференциальной эволюции[28])
- Популяция состоит из 400 экземпляров, каждый из которых задается пятеркой: две координаты и три канала цвета
- Генерация потомка — линейная комбинация трех случайных родителей

[27] Su, Jiawei, Danilo Vasconcellos Vargas, and Kouichi Sakurai. "One pixel attack for fooling deep neural networks." 2017

[28] Storn, Rainer, and Kenneth Price. "Differential evolution — a simple and efficient heuristic for global optimization over continuous spaces." 1997

Метод атаки: One pixel

- Однопиксельная атака[27] — предельный случай ℓ_0-атаки
- **Идея**: применить эволюционный алгоритм (дифференциальной эволюции[28])
- Популяция состоит из 400 экземпляров, каждый из которых задается пятеркой: две координаты и три канала цвета
- Генерация потомка — линейная комбинация трех случайных родителей

Original image (dog)

Airplane	Automobile	Bird
Cat	Deer	Frog
Horse	Ship	Truck

Target classes

[27] Su, Jiawei, Danilo Vasconcellos Vargas, and Kouichi Sakurai. "One pixel attack for fooling deep neural networks." 2017

[28] Storn, Rainer, and Kenneth Price. "Differential evolution — a simple and efficient heuristic for global optimization over continuous spaces." 1997

Метод универсальной атаки

- До этого все атаки строились как функция от входа x

[29] Moosavi-Dezfooli, Seyed-Mohsen, et al. "Universal adversarial perturbations." 2016

Метод универсальной атаки

- До этого все атаки строились как функция от входа x
- Однако можно строить т.н. “универсальную” атаку[29], которая будет уже функцией от всего обучающего множества X

[29] Moosavi-Dezfooli, Seyed-Mohsen, et al. “Universal adversarial perturbations.” 2016

Метод универсальной атаки

- До этого все атаки строились как функция от входа x
- Однако можно строить т.н. “универсальную” атаку[29], которая будет уже функцией от всего обучающего множества X
- При построении атаки будем искать r, примерно одинаково ломающий все классы из X

[29] Moosavi-Dezfooli, Seyed-Mohsen, et al. “Universal adversarial perturbations.” 2016

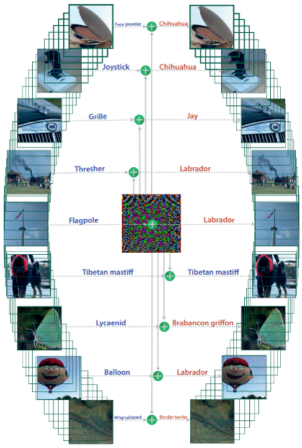
Метод универсальной атаки

- До этого все атаки строились как функция от входа x
- Однако можно строить т.н. “универсальную” атаку[29], которая будет уже функцией от всего обучающего множества X
- При построении атаки будем искать r, примерно одинаково ломающий все классы из X
- Справа — универсальное возмущение для любого входа, которое ломает классификатор

[29] Moosavi-Dezfooli, Seyed-Mohsen, et al. “Universal adversarial perturbations.” 2016

Метод универсальной атаки

- До этого все атаки строились как функция от входа x
- Однако можно строить т.н. “универсальную” атаку[29], которая будет уже функцией от всего обучающего множества X
- При построении атаки будем искать r, примерно одинаково ломающий все классы из X
- Справа — универсальное возмущение для любого входа, которое ломает классификатор



[29] Moosavi-Dezfooli, Seyed-Mohsen, et al. “Universal adversarial perturbations.” 2016

Физические атаки

- Все атаки до этого работали в т.н. цифровой области (digital domain): изменяли картинку на уровне пикселей

[30] Kurakin, Alexey, Ian Goodfellow, and Samy Bengio. "Adversarial examples in the physical world." 2016

Физические атаки

- Все атаки до этого работали в т.н. цифровой области (digital domain): изменяли картинку на уровне пикселей
- Если нет возможности проатаковать изображение непосредственно перед подачей в CHC, то такая атака бесполезна

[30] Kurakin, Alexey, Ian Goodfellow, and Samy Bengio. "Adversarial examples in the physical world." 2016

Физические атаки

- Все атаки до этого работали в т.н. цифровой области (digital domain): изменяли картинку на уровне пикселей
- Если нет возможности проатаковать изображение непосредственно перед подачей в СНС, то такая атака бесполезна
- Поэтому атаки в реальном мире (real-world), или физические атаки, наиболее универсальны

[30] Kurakin, Alexey, Ian Goodfellow, and Samy Bengio. "Adversarial examples in the physical world." 2016

Физические атаки

- Все атаки до этого работали в т.н. цифровой области (digital domain): изменяли картинку на уровне пикселей
- Если нет возможности проатаковать изображение непосредственно перед подачей в CHC, то такая атака бесполезна
- Поэтому атаки в реальном мире (real-world), или физические атаки, наиболее универсальны
- Первый пример физической атаки[30] — атака на изображение в цифровой области, затем печать на физическом носителе (бумага), затем снимок цифровой камерой и последующая обработка CHC

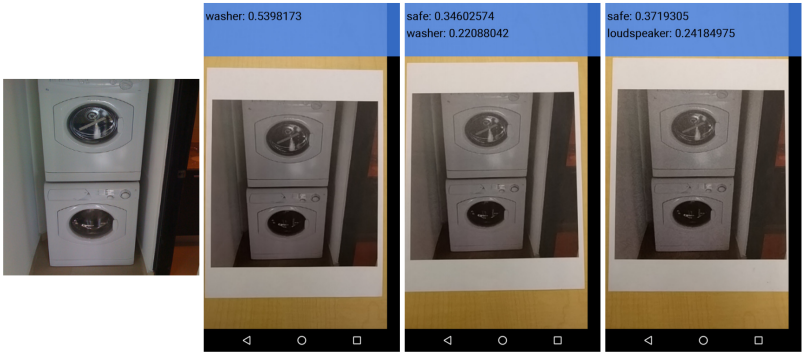
[30] Kurakin, Alexey, Ian Goodfellow, and Samy Bengio. "Adversarial examples in the physical world." 2016

Физические атаки

- Все атаки до этого работали в т.н. цифровой области (digital domain): изменяли картинку на уровне пикселей
- Если нет возможности проатаковать изображение непосредственно перед подачей в CHC, то такая атака бесполезна
- Поэтому атаки в реальном мире (real-world), или физические атаки, наиболее универсальны
- Первый пример физической атаки[30] — атака на изображение в цифровой области, затем печать на физическом носителе (бумага), затем снимок цифровой камерой и последующая обработка CHC
- Никакой специальной технологии для генерации таких атак еще не было, просто была показана их возможность

[30] Kurakin, Alexey, Ian Goodfellow, and Samy Bengio. "Adversarial examples in the physical world." 2016

Физические атаки



(a) Image from dataset (b) Clean image (c) Adv. image, $\epsilon = 4$ (d) Adv. image, $\epsilon = 8$

Физические атаки: EOT

- Подход EOT[31] (**E**xpectation **O**ver **T**ransformation) учитывает, что объект в реальном мире обычно претерпевает ряд преобразований таких как:
  - Масштабирование
  - Трансляция (тряска)
  - Изменение яркости и/или контрастности

[31] Athalye, Anish, et al. "Synthesizing robust adversarial examples." 2017

Физические атаки: EOT

- Подход EOT[31] (**E**xpectation **O**ver **T**ransformation) учитывает, что объект в реальном мире обычно претерпевает ряд преобразований таких как:
  - Масштабирование
  - Трансляция (тряска)
  - Изменение яркости и/или контрастности
- Поэтому задача — найти (направленную) состязательную атаку r с учетом множества преобразований T:

EOT

Найти $\arg\max_r \mathbb{E}_{g\sim T} P(y_t|g(x+r))$ при условии:

1. $f(x) = y_{gt} \neq y_t$
2. $\mathbb{E}_{g\sim T}||g(x+r) - g(x)||_p < \epsilon$
3. $x \in B$

[31] Athalye, Anish, et al. "Synthesizing robust adversarial examples." 2017

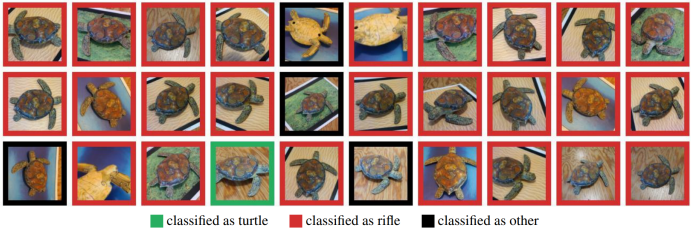
Физические атаки: EOT

- В итоге, используя широкий ряд преобразований T, удалось сделать состязательный 3D-пример

Физические атаки: EOT

- В итоге, используя широкий ряд преобразований T, удалось сделать состязательный 3D-пример

Transformation	Minimum	Maximum
Camera distance	2.5	3.0
X/Y translation	−0.05	0.05
Rotation	any	
Background	(0.1, 0.1, 0.1)	(1.0, 1.0, 1.0)
Lighten / Darken (additive)	−0.15	0.15
Lighten / Darken (multiplicative)	0.5	2.0
Per-channel (additive)	−0.15	0.15
Per-channel (multiplicative)	0.7	1.3
Gaussian Noise (stdev)	0.0	0.1



Еще примеры физических атак

- Интересны примеры атак на объекты ImageNet[32], дорожные знаки [33] и даже системы распознавания лиц[34]

[32] Brown, Tom B., et al. "Adversarial patch." 2017

[33] Eykholt, Kevin, et al. "Robust physical-world attacks on deep learning models." 2017

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

Еще примеры физических атак

- Интересны примеры атак на объекты ImageNet[32], дорожные знаки [33] и даже системы распознавания лиц[34]
- Примечательно, что все эти атаки по существу ℓ_0-атаки, а также используют NPS и TV-добавки в функцию потерь
  - NPS (**N**on **P**rintability **S**core): штраф за использование цветов, которые не может воспроизвести данный принтер
  - TV (**T**otal **V**ariation): штраф за негладкость картинки

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

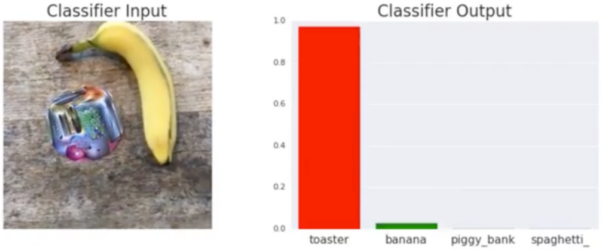
[32] Brown, Tom B., et al. "Adversarial patch." 2017

[33] Eykholt, Kevin, et al. "Robust physical-world attacks on deep learning models." 2017

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

Еще примеры физических атак

Атака на объекты ImageNet:



Атака на дорожные знаки:

Атака на FaceID:

Атаки на ведущую систему распознавания лиц

- Обычно система распознавания содержит два важных элемента: детектор и извлекатель признаков (часто называемый FaceID)

[35] Zhang, Kaipeng, et al. "Joint face detection and alignment using multitask cascaded convolutional networks." 2016

[36] Deng, Jiankang, et al. "Arcface: Additive angular margin loss for deep face recognition." 2018

Атаки на ведущую систему распознавания лиц

- Обычно система распознавания содержит два важных элемента: детектор и извлекатель признаков (часто называемый FaceID)
- Использовались: крайне легкий нейросетевой детектор MTCNN[35] и ведущая открытая система извлечения признаков ArcFace[36]

[35] Zhang, Kaipeng, et al. "Joint face detection and alignment using multitask cascaded convolutional networks." 2016

[36] Deng, Jiankang, et al. "Arcface: Additive angular margin loss for deep face recognition." 2018

Атаки на ведущую систему распознавания лиц

- Обычно система распознавания содержит два важных элемента: детектор и извлекатель признаков (часто называемый FaceID)
- Использовались: крайне легкий нейросетевой детектор MTCNN[35] и ведущая открытая система извлечения признаков ArcFace[36]
- Атаки на FaceID: с цветным патчем и черно-белым

[35] Zhang, Kaipeng, et al. "Joint face detection and alignment using multitask cascaded convolutional networks." 2016

[36] Deng, Jiankang, et al. "Arcface: Additive angular margin loss for deep face recognition." 2018

Атаки на ведущую систему распознавания лиц

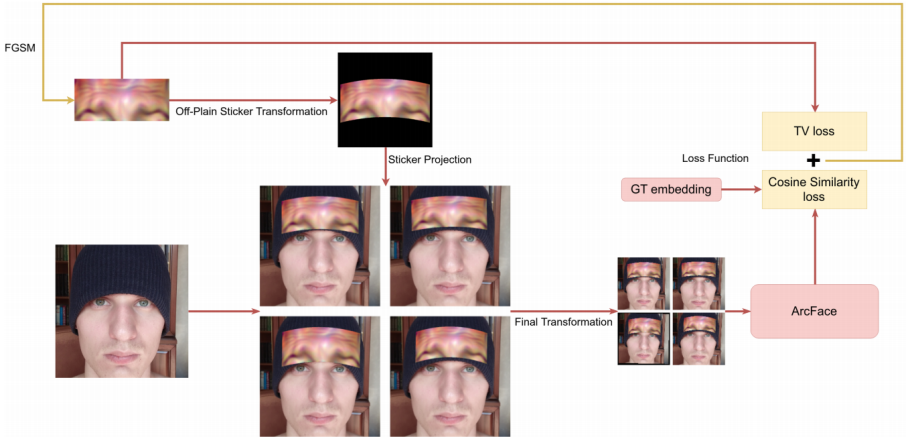
- Обычно система распознавания содержит два важных элемента: детектор и извлекатель признаков (часто называемый FaceID)
- Использовались: крайне легкий нейросетевой детектор MTCNN[35] и ведущая открытая система извлечения признаков ArcFace[36]
- Атаки на FaceID: с цветным патчем и черно-белым
- Атака на детектор: маска и черно-белый патч

[35] Zhang, Kaipeng, et al. "Joint face detection and alignment using multitask cascaded convolutional networks." 2016

[36] Deng, Jiankang, et al. "Arcface: Additive angular margin loss for deep face recognition." 2018

AdvHat[37] — шапка-невидимка

Алгоритм обучения:



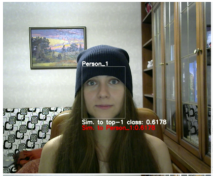
[37] Komkov, Stepan, and Aleksandr Petiushko. "AdvHat: Real-world adversarial attack on ArcFace Face ID system." 2019

AdvHat — шапка-невидимка

Устойчивость к поворотам и разной освещенности[38]:

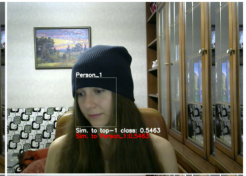
Фронтальное лицо (нет атаки)

Близость до своего эталона: 0.61



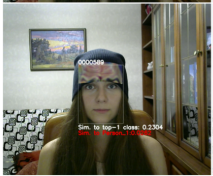
Поворот лица (нет атаки)

Близость до своего эталона: 0.54



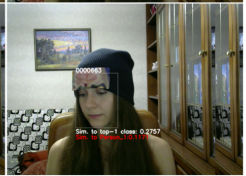
Фронтальное лицо (атака)

Близость до своего эталона: 0.02
Близость до другого эталона: 0.23



Поворот лица (атака)

Близость до своего эталона: 0.11
Близость до другого эталона: 0.27



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

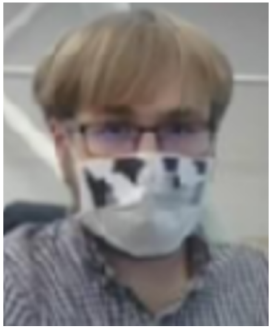
Adversarial patches[39] — черно-белые патчи

Дальнейшее развитие атак на FaceID:

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

Атака на детектор лиц[41] — черно-белые патчи

Физическая атака на неглубокий и поэтому крайне устойчивый к состязательным атакам детектор MTCNN[40]:



[40] https://www.youtube.com/watch?v=0Y70OIS8bxs

[41] Kaziakhmedov, Edgar, et al. "Real-world attack on MTCNN face detection system." 2019

Вариант защиты от состязательных атак в реальном мире[42]

- Большинство состязательных атак в реальном мире основано на том, что к объекту добавляется специальная (обычно – прямоугольная) картинка, которая и ломает распознавание.

[42] Wu, Tong, Liang Tong, and Yevgeniy Vorobeychik. "Defending Against Physically Realizable Attacks on Image Classification." 2019

Вариант защиты от состязательных атак в реальном мире[42]

- Большинство состязательных атак в реальном мире основано на том, что к объекту добавляется специальная (обычно – прямоугольная) картинка, которая и ломает распознавание.
  - А давайте будем обучать в цифровой области (на обычных картинках), используя состязательное обучение и добавляя специальную прямоугольную аугментацию!

[42] Wu, Tong, Liang Tong, and Yevgeniy Vorobeychik. "Defending Against Physically Realizable Attacks on Image Classification." 2019

Вариант защиты от состязательных атак в реальном мире[42]

- Большинство состязательных атак в реальном мире основано на том, что к объекту добавляется специальная (обычно – прямоугольная) картинка, которая и ломает распознавание.
  - А давайте будем обучать в цифровой области (на обычных картинках), используя состязательное обучение и добавляя специальную прямоугольную аугментацию!
- Состязательное обучение предлагается делать двухэтапным:

[42] Wu, Tong, Liang Tong, and Yevgeniy Vorobeychik. "Defending Against Physically Realizable Attacks on Image Classification." 2019

Вариант защиты от состязательных атак в реальном мире[42]

- Большинство состязательных атак в реальном мире основано на том, что к объекту добавляется специальная (обычно – прямоугольная) картинка, которая и ломает распознавание.
  - А давайте будем обучать в цифровой области (на обычных картинках), используя состязательное обучение и добавляя специальную прямоугольную аугментацию!
- Состязательное обучение предлагается делать двухэтапным:
  - Сначала ищем наилучшую позицию для прямоугольника (среднего серого цвета),

[42] Wu, Tong, Liang Tong, and Yevgeniy Vorobeychik. "Defending Against Physically Realizable Attacks on Image Classification." 2019

Вариант защиты от состязательных атак в реальном мире[42]

- Большинство состязательных атак в реальном мире основано на том, что к объекту добавляется специальная (обычно – прямоугольная) картинка, которая и ломает распознавание.
  - А давайте будем обучать в цифровой области (на обычных картинках), используя состязательное обучение и добавляя специальную прямоугольную аугментацию!
- Состязательное обучение предлагается делать двухэтапным:
  - Сначала ищем наилучшую позицию для прямоугольника (среднего серого цвета),
    - Либо полным перебором (скользящим окном) по всем возможным позациям,

[42] Wu, Tong, Liang Tong, and Yevgeniy Vorobeychik. "Defending Against Physically Realizable Attacks on Image Classification." 2019

Вариант защиты от состязательных атак в реальном мире[42]

- Большинство состязательных атак в реальном мире основано на том, что к объекту добавляется специальная (обычно – прямоугольная) картинка, которая и ломает распознавание.
  - А давайте будем обучать в цифровой области (на обычных картинках), используя состязательное обучение и добавляя специальную прямоугольную аугментацию!
- Состязательное обучение предлагается делать двухэтапным:
  - Сначала ищем наилучшую позицию для прямоугольника (среднего серого цвета),
    - Либо полным перебором (скользящим окном) по всем возможным позациям,
    - Либо на основе позиций максимального значения градиента по входу,

[42] Wu, Tong, Liang Tong, and Yevgeniy Vorobeychik. "Defending Against Physically Realizable Attacks on Image Classification." 2019

Вариант защиты от состязательных атак в реальном мире[42]

- Большинство состязательных атак в реальном мире основано на том, что к объекту добавляется специальная (обычно – прямоугольная) картинка, которая и ломает распознавание.
  - А давайте будем обучать в цифровой области (на обычных картинках), используя состязательное обучение и добавляя специальную прямоугольную аугментацию!
- Состязательное обучение предлагается делать двухэтапным:
  - Сначала ищем наилучшую позицию для прямоугольника (среднего серого цвета),
    - Либо полным перебором (скользящим окном) по всем возможным позациям,
    - Либо на основе позиций максимального значения градиента по входу,
  - А затем – запускаем состязательную атаку (здесь PGD) внутри этого прямоугольника.

[42] Wu, Tong, Liang Tong, and Yevgeniy Vorobeychik. "Defending Against Physically Realizable Attacks on Image Classification." 2019

Заключительные выводы

- На данный момент СНС (в целом) работают гораздо лучше человека

[43] Image credit: http://reddit.com

Заключительные выводы

- На данный момент CHC (в целом) работают гораздо лучше человека
- CHC легко “обмануть”, используя их неустойчивость по входу

[43] Image credit: `http://reddit.com`

Заключительные выводы

- На данный момент CHC (в целом) работают гораздо лучше человека
- CHC легко “обмануть”, используя их неустойчивость по входу
- Наиболее распространенный прием атаки — производная по входу

[43] Image credit: `http://reddit.com`

Заключительные выводы

- На данный момент CHC (в целом) работают гораздо лучше человека
- CHC легко “обмануть”, используя их неустойчивость по входу
- Наиболее распространенный прием атаки — производная по входу
- Перенести атаку в реальный мир непросто

[43] Image credit: `http://reddit.com`

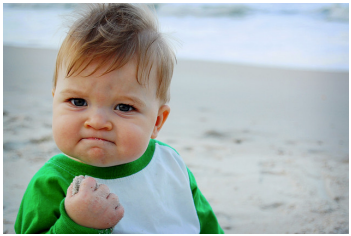
Заключительные выводы

- На данный момент CHC (в целом) работают гораздо лучше человека
- CHC легко “обмануть”, используя их неустойчивость по входу
- Наиболее распространенный прием атаки — производная по входу
- Перенести атаку в реальный мир непросто
- Однако можно сломать даже супер навороченные системы распознавания лиц, имея лишь обычный принтер

[43] Image credit: `http://reddit.com`

Заключительные выводы

- На данный момент CHC (в целом) работают гораздо лучше человека
- CHC легко “обмануть”, используя их неустойчивость по входу
- Наиболее распространенный прием атаки — производная по входу
- Перенести атаку в реальный мир непросто
- Однако можно сломать даже супер навороченные системы распознавания лиц, имея лишь обычный принтер
- Для человечества пока еще не все потеряно[43]!



[43] Image credit: http://reddit.com

Спасибо за внимание!