Исследование состязательной устойчивости в реальном мире сверточных нейросетей

на примере систем детекции и распознавания лиц

Петюшко Александр

МГУ им. М.В.Ломоносова, к.ф.-м.н.
Video Intelligence and Fundamental Research Team Leader
Huawei, Intelligent Systems and Data Science Lab

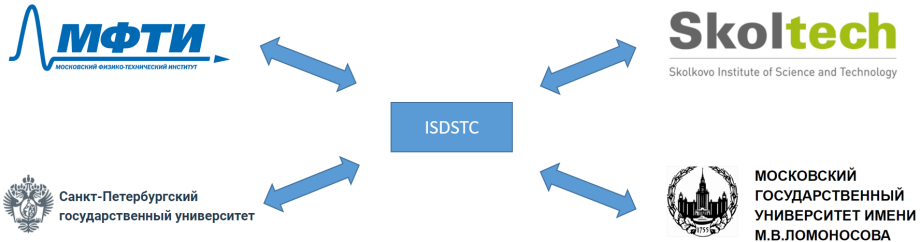
15 апреля, 2021

Plan

1. Лаборатория Интеллектуальных Систем и Науки о Данных
2. Потрясающие успехи CHC в компьютерном зрении
3. (Не) устойчивость CHC в компьютерном зрении
4. Методы для генерации состязательных примеров в цифровой области
5. ℓ_0-состязательные примеры
6. Методы для генерации состязательных примеров в реальном мире
7. Состязательные примеры для систем детекции лиц
8. Состязательные примеры для систем распознавания лиц
9. Защита от состязательных примеров в реальном мире
10. Black-box восстановление лица по вектору признаков

Научное сотрудничество: Лаборатория Интеллектуальных Систем и Науки о Данных

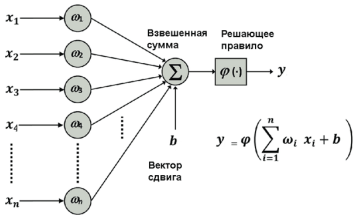
Российский исследовательский институт → Московский исследовательский институт → Лаборатория Интеллектуальных Систем и Науки о Данных

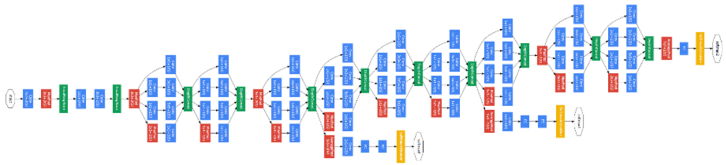


Развитие нейросетей

Начиная с 1943 года, когда впервые была предложена математическая формализация МакКалоком и Питтсом понятия “искусственного нейрона”, нейросети становились[1]:

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

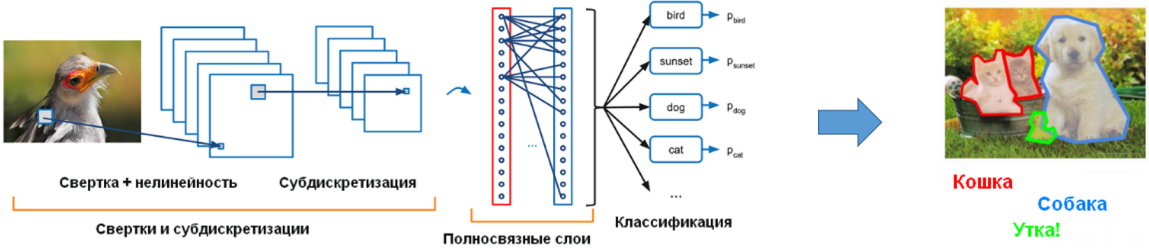




[1]Image credits: https://arxiv.org/abs/1409.4842

Сверточные нейросети[2]

- 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?



[2]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?

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?

CNN vs Human[3]

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

Человек или СНС?

ImageNet[4] (1000-классовая база данных изображений)

- Top-5 ошибка для человека[5]: 5.1%
- Top-5 ошибка для СНС[6]: 2.0%

Labeled Faces in the Wild[7] (база данных лиц)

- Ошибка верификации для человека[8]: 2.47%
- Ошибка верификации для СНС[9]: 0.17%

LFW

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

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

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

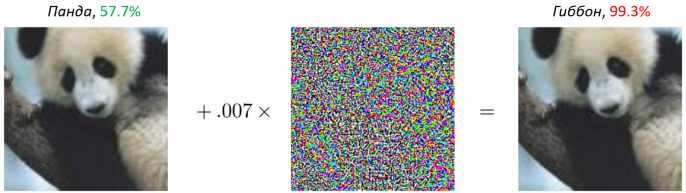
[7] http://vis-www.cs.umass.edu/lfw/

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

[9] 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
- Например, результат классификации с “панды” поменяется на “гиббона”[10]

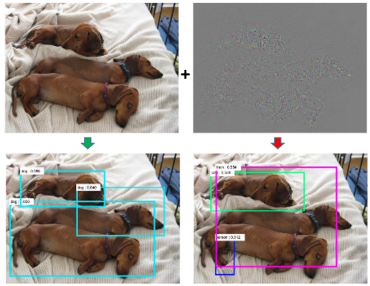


Такое возмущение называется **состязательным примером** (adversarial perturbation / example / attack)

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

Состязательные примеры в разных задачах

СНС для обнаружения и сегментации изображений[11]:



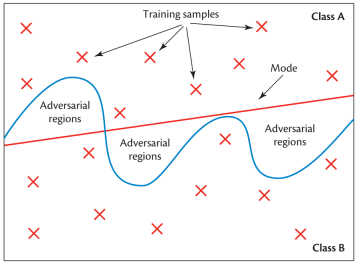
И даже НС для вопросно-ответных систем (QA, question answering systems)[12]:

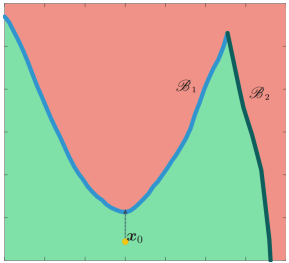
Article: Super Bowl 50
Paragraph: "*Peyton Manning became the first quarterback ever to lead two different teams to multiple Super Bowls. He is also the oldest quarterback ever to play in a Super Bowl at age 39. The past record was held by John Elway, who led the Broncos to victory in Super Bowl XXXIII 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

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

Одна из главных причин существования состязательных примеров

- 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[13,14]



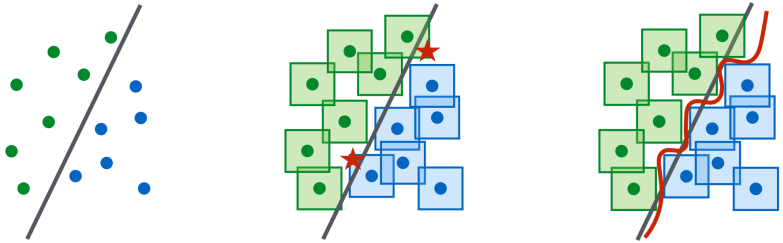


[13] Image credit: `https://secml.github.io/`

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

Простой метод защиты

- Поскольку можно изменить решение 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_∞)[15]



[15] 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
- 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
- Такой метод обучения называется состязательным (adversarial training)[16]

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 использовался в состязательном обучении

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

Состязательные примеры: 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 — правильный класс для x
- θ — параметры CHC-классификатора
- $L(\theta, x, y)$ — функция потерь
- $f(x)$ — выход классификатора (распознанный класс); при обучении мы добиваемся равенства $f(x) = y$
- $r \in B = [0,1]^{C \times M \times N}$ — аддитивная добавка ко входу x

Состязательный пример и устойчивость: формулировки

Цель состязательного примера

Поменять выход классификатора f на неправильный путем добавления минимального (по некоторой норме ℓ_p) возмущения r:

1. $||r||_p \to \min$
2. $f(x) = y$ (изначальный ответ СНС верный)
3. $f(x + r) \neq y$ (меняем выход с помощью возмущения r)
4. $x + r \in B$ (остаемся во множестве допустимых изображений)

Устойчивость классификатора

Найти такой класс возмущения $S(x, f) \subseteq B$, при котором классификатор не меняет свой выход:

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

Состязательное обучение: формулировка

В обозначениях выше обычное обучение можно сформулировать как

Обучение на примерах

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

В состязательном обучении[17] мы сначала генерируем (например, каким-нибудь методом) самый сложный пример из некоторой окрестности Δ входного примера (например, по ℓ_p-норме), а уже затем минимизируем по параметрам нейросети:

Состязательное обучение

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

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

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 примеров (1)

По цели

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

По осведомленности

- Открытые (white-box): генерирующий знает все о классификаторе (архитектуру и веса)
- Закрытые (black-box): генерирующий имеет частичную информацию о классификаторе (обычно только информацию о выходе)

По условию применения

- Цифровые (digital): для применения к фотографии
- Реальные (real-world): для применения к реальному объекту

Классификация состязательных примеров (2)

По универсальности

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

По переносимости

- Непереносимые (non-transferable): работают только для узкого класса классификаторов
- Переносимые (transferable): работают для широкого класса классификаторов (но при этом могут быть не универсальными)

- Наиболее сложный случай — направленный закрытый реальный универсальный переносимый состязательный пример
- В данной лекции будем рассматривать открытые состязательные примеры

Эффективность состязательных примеров

Введем простой критерий успешности (success) $S(A, Z)$ алгоритма A генерации состязательного примера $r_A(x)$ на множестве $Z \ni (x^i, y^i)$:

- В случае ненаправленного примера:

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

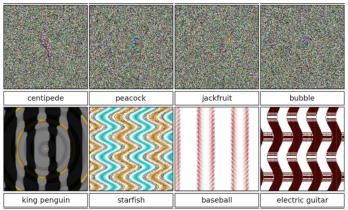
- В случае направленного на класс y_t:

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

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

Предтеча состязательных примеров

- 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”[18] (fooling images) и строились с помощью эволюционных алгоритмов



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

Метод генерации: L-BFGS-B

- Первый предложенный метод[19] использовал ℓ_2-норму для ограничения
- Рассматривались направленные примеры на класс $y_t \neq y$
- Функционал для минимизации с ограничением $x + r \in B$, $c = const$:

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

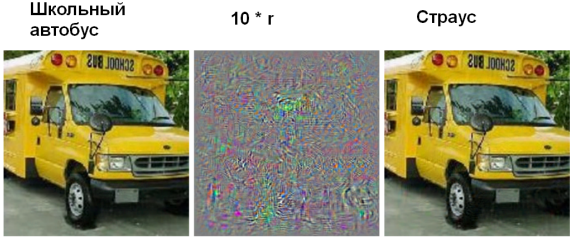
- Для оптимизации использовался метод L-BFGS-B[20] (**L**imited memory **B**royden–**F**letcher–**G**oldfarb–**S**hanno algorithm with **B**ox constraints) — квази-Ньютоновский метод минимизации с ограничением на память и на переменные
- В какой-то мере примеры были переносимы на другие архитектуры

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

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

Метод генерации: 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%

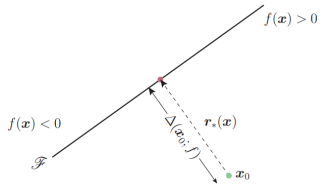
Метод генерации: CW

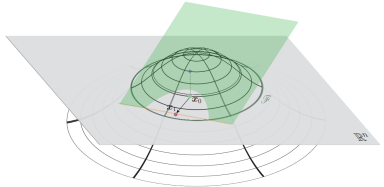
- Вспомним оптимизационную функцию: $c||r||_2 + L(\theta, x, y_t) \to \min_r$
- Поменяем функцию для направленных примеров: $f(x) = y_t \Leftrightarrow g(x) \leq 0$
  - Например, $g(x) = \max_{i \neq t} F(x)_i - F(x)_t$
  - где F(x) — выход SoftMax (вероятности), или логиты на входе SoftMax
- Перейдем к формулировке $||r||_p + c \cdot g(x + r) \to \min_r$, где $x + r \in B$
- Такой метод называется CW[21] (**C**arlini-**W**agner) и хорошо работает на разных метриках ℓ_p
- В данном случае не оптимизируем глядя на прокси-функцию (функцию потерь), а смотрим на реальный выход (вероятности или логиты)

[21] Carlini, Nicholas, and David Wagner. "Towards evaluating the robustness of neural networks." 2016

Метод генерации: 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: $\frac{|f(x_0)|}{||w||_2}$
  - $\Rightarrow$ Возмущение: $r = -\frac{f(x_0)}{||w||_2^2} w$
- 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 генерации: DeepFool

- Итеративный алгоритм DeepFool[22] 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

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

Метод генерации: 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[23] (**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. $\nabla_\theta L(\theta, x, y)$
- Теперь исследуется норма возмущения ℓ_∞

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

Метод генерации: I-FGSM (PGD)

- Часто линейная оценка окрестности функции достаточно грубая, и один шаг FGSM порой не приводит к хорошему примеру
- Для этого применяют итеративный метод I-FGSM[24] (**I**terative FGSM), который позволяет двигаться в сторону границы классификатора более точно
- Если Π_B — проекция на B, то в случае ненаправленных примеров

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

- 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)

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

Метод генерации: MI-FGSM

- **Замечание**: Методы генерации все больше похожи на шаги оптимизатора
- **Идея**: давайте использовать сглаживание градиента — MI-FGSM[25] (**M**omentum 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 $\|\boldsymbol{x}^* - \boldsymbol{x}\|_\infty \leq \epsilon$.

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

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

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

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

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

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

Сравнение FGSM-like методов

	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**

Как видно, MI-FGSM — наиболее успешная методика генерации.

Метод генерации: JSMA

- JSMA[26] (**J**acobian-based **S**aliency **M**ap **A**ttack) — один из первых ℓ_0-методов, для которого важно количество задействованных в методе пикселей, а не их значения
- **Идея**: менять те пиксели, которые дают максимальный вклад в производную по входу для нужного класса для направленного примера
- Можно делать это итеративно, постепенно добавляя пиксели в активную область 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 генерации: One pixel

- Однопиксельный состязательный пример[27] — предельный случай ℓ_0-метода генерации
- **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

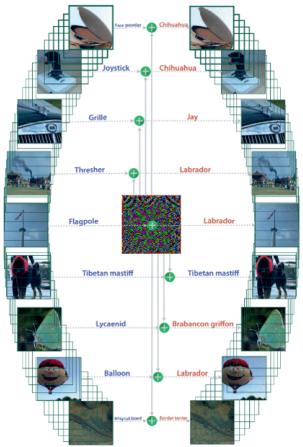
Target classes

[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

Универсальные состязательные примеры

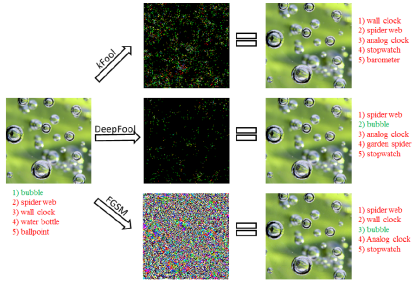
- До этого все примеры строились как функция от входа x
- Однако можно строить т.н. “универсальный” пример[29], который будет уже функцией от всего обучающего множества X
- При построении примера будем искать r, примерно одинаково далекий от всех классов из X
- Справа — универсальное возмущение для любого входа, которое меняет выход классификатора

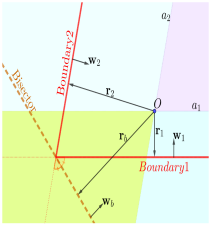


[29] Moosavi-Dezfooli, Seyed-Mohsen, et al. “Universal adversarial perturbations.” 2016

Состязательные примеры для мультиклассового случая

- Если нужно распознавать несколько классов (а обычно так и бывает), то просто “сдвинуть” правильный класс с первого места по вероятности в некоторых задачах недостаточно
- В таком случае помогают т.н. “top-k состязательные примеры”[30], где $k > 1$
- В данной работе используются красивые геометрические концепты (биссектрисы для top-k классов)

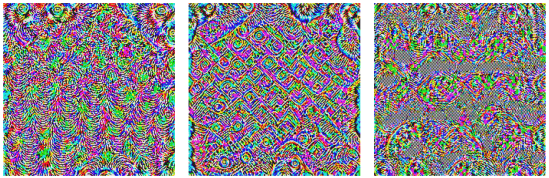




[30] Tursynbek N. et al. “Geometry-Inspired Top-k Adversarial Perturbations”. 2020

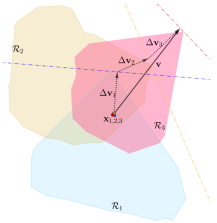
Универсальные состязательные примеры для мультиклассового случая

- Удалось построить универсальный пример, обладающий тем же свойством — “сдвигающий” правильный класс за пределы top-k предсказаний по вероятности



(a) ResNet-18 (b) VGG-16 (c) MobileNetV2

Figure 7. Result of kUAP ($k = 3$) to different deep neural networks for ILSVRC2012



Классы методов генерации

В целом, можем поделить методы генерации состязательных примеров на следующие классы:

- На основе ℓ_2-нормы (в т.ч. геометрические): наиболее удобные для классической оптимизации
- На основе ℓ_∞-нормы: соответствуют процессу восприятия человеческим глазом визуальной информации
- На основе ℓ_0-нормы: минимизируется область возмущения, но не ограничивается его сила

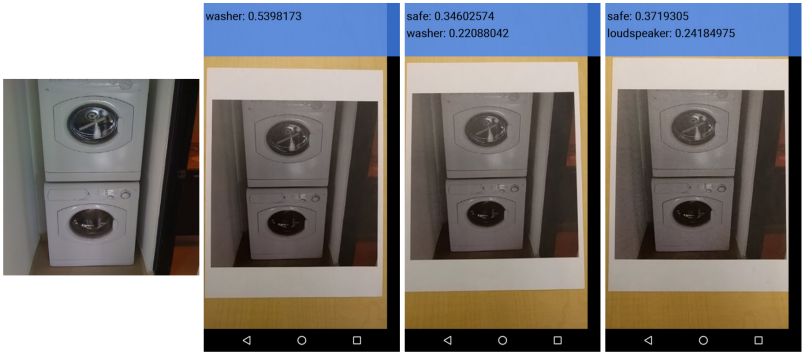
Тем не менее, для генерации физически реализуемых состязательных примеров этого мало.

Физические состязательные примеры

- Все состязательные примеры до этого работали в т.н. цифровой области (digital domain): изменяли картинку на уровне пикселей
- Если нет возможности изменить изображение непосредственно перед подачей в CHC, то такой метод бесполезен
- Поэтому состязательные примеры в реальном мире (real-world), или физические, наиболее универсальны
- Первый пример физических состязательных примеров[31] — генерация для изображения в цифровой области, затем печать на физическом носителе (бумага), затем снимок цифровой камерой и последующая обработка CHC
- Никакой специальной технологии для генерации таких физических примеров еще не было, просто была показана их возможность

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

Физические состязательные примеры



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

Физические состязательные примеры: EOT

- Не имеем доступа к фото (и его пикселям) $\Rightarrow$ единственная возможность — это изменить внешний вид самого объекта
- Подход **E**xpectation **O**ver **T**ransformation (EOT)[32] учитывает, что объект в реальном мире обычно претерпевает ряд преобразований таких как:
  - Масштабирование
  - Трансляции и повороты
  - Изменение яркости и/или контрастности, шум и т.п.
- Т.о. для объекта x нужно найти состязательный пример r с учетом преобразований $g \in T$:

EOT

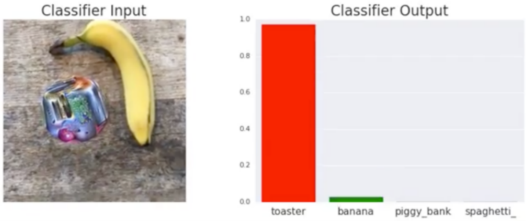
Найти $\arg\min_r \mathbb{E}_{g\sim T}[P(y|g(x+r))]$ для ненаправленного примера при условии:

1. $\mathbb{E}_{g\sim T}[d(g(x+r), g(x))] < \epsilon$, где $d(a,b)$ – некоторая функция расстояния (например, $d(a,b) = ||a-b||_p$)
2. $x + r \in B$

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

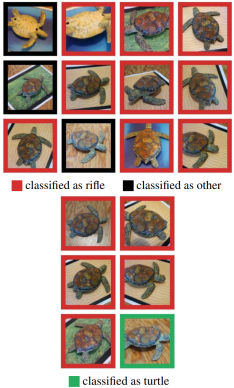
Физические состязательные примеры

Для ImageNet[33]:



Для дорожных знаков[34]:

3D-состязательные примеры:



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

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

Физические состязательные примеры: основные ингридиенты

- ℓ_0 оптимизация (маска) + EOT: обязательно
- **T**otal **V**ariation (TV) функция потерь — штраф за негладкость примера (в реальном мире мало больших попиксельных градиентов):

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

- **N**on-**P**rintability **S**core (NPS) — штраф за использование цветов, которые не поддерживаются (принтером). Если $G \subset [0,1]^3$ — поддерживаемая цветовая палитра (gamut), то штраф за использование цвета пикселя $q_0 \in [0,1]^3$:

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

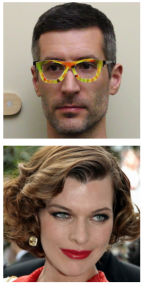
- Дополнительный маппинг цветов (например, принтер печатает не цвет c, а близкий $m(c)$, и это нужно учитывать)

Состязательные примеры для систем детекции и распознавания лиц

- Изначально т.н. **Camouflage Art**[35] использовался для обхода лидирующей системы детекции лиц Viola-Jones
- Это был вручную подобранный грим для обхода детектора Хаара
- Прорыв случился с работой Sharif et al.[36], где предложили использовать состязательные очки
- Использовано: ℓ_0-оптимизация + EOT + TV + NPS + цветовой маппинг
- Минусы: предобученные лица, системы распознавания лиц прошлого поколения

cvdazzle.com

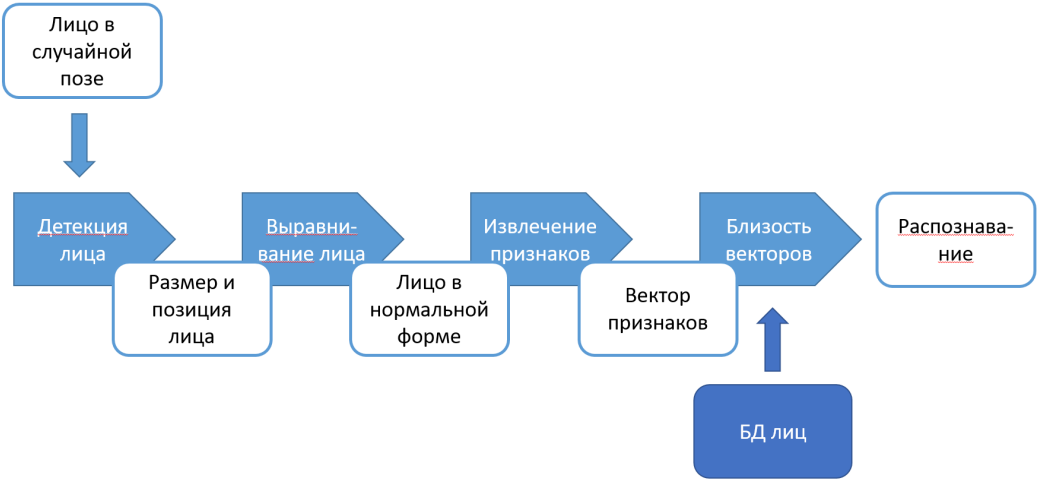
Состязательные очки



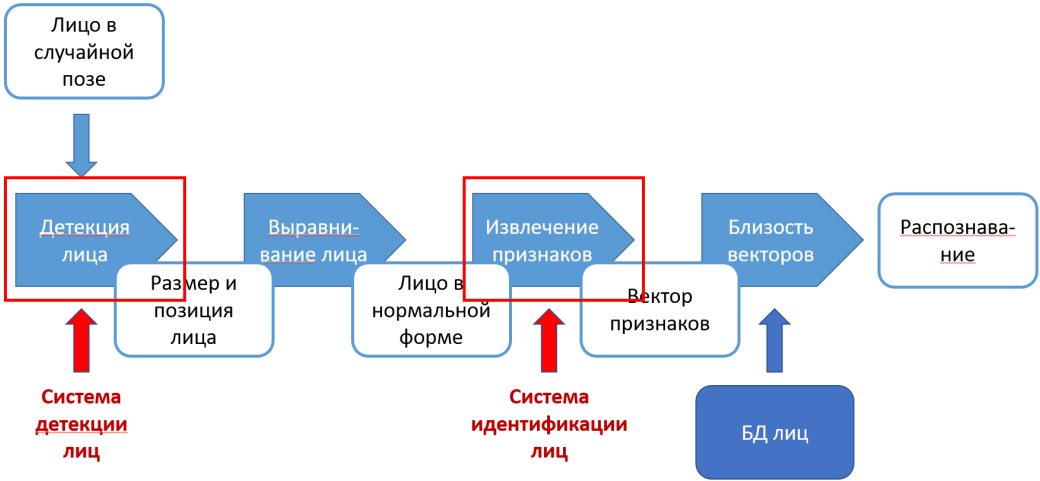
[35] Feng R. et al. "Facilitating fashion camouflage art." 2013

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

Общая схема детекции и распознавания лиц



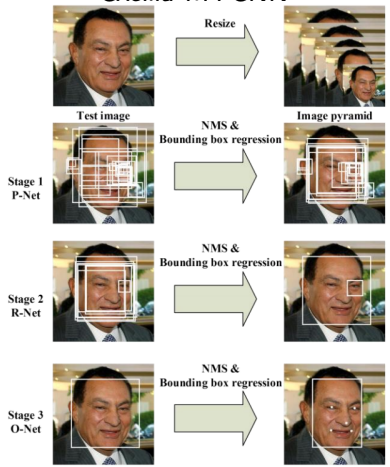
Общая схема детекции и распознавания лиц



Детектор лиц: MTCNN[37]

- В отличие от современных глубоких детекторов типа Faster RCNN или YOLO, MTCNN очень простой и неглубокий ⇒ меньше поле восприятия, сложнее поменять решение детектора
- В MTCNN каскадный подход: сначала грубое приближение (P-Net), а затем исправление (R-Net, O-Net)
- Решение: использовать для генерации состязательных примеров самый первый классификационный слой в P-Net (а не функции потерь для прямоугольников или ключевых точек)

Схема MTCNN

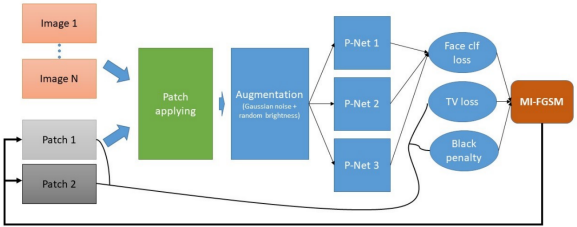


[37] Zhang K. et al. "Joint face detection and alignment using multitask cascaded convolutional networks." 2016

Состязательные примеры для детектора лиц MTCNN

- EOT: Гауссов шум, размер маски, яркость, набор разных фото лица
- TV: +, NPS: –
- Маппинг цветов: штрафуем за близость к черному цвету ($x_{i,j} = 1$) $\Rightarrow$ новая добавка в функцию потерь: $L_{BLK}(x) = \sum_{i,j} x_{i,j}$
- Оптимизатор: MI-FGSM

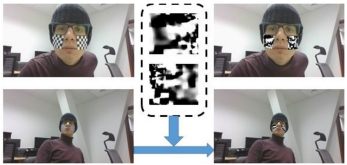
Схема генерациии состязательных примеров для MTCNN



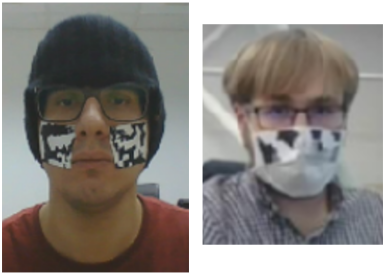
Состязательные примеры для детектора лиц MTCNN

- ℓ_0-оптимизация: 2 версии
  1. две раздельных маски на щеках
  2. цельная медицинская маска
- У MTCNN маленькое поле восприятия $\Rightarrow$ примеры не носят семантический характер
- Оценка параметров локальных аффинных проекций на основе заранее подготовленной специальной сетки для обучения

Проекция

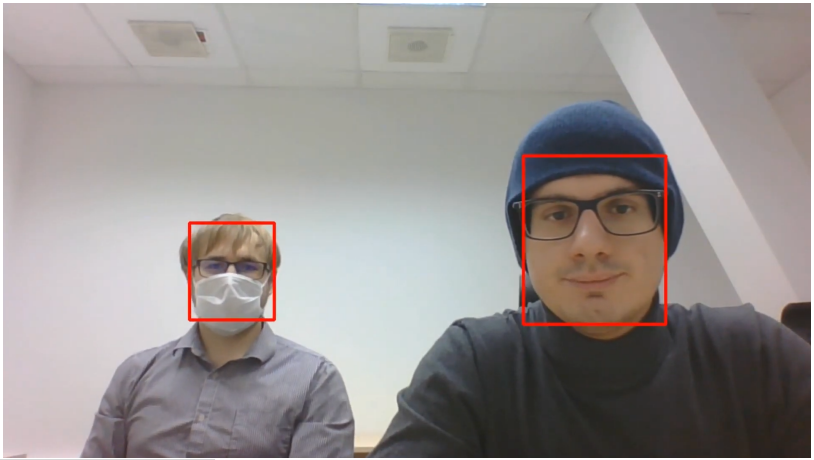


Примеры



Состязательные примеры для детектора лиц MTCNN: результат

Детали: статья[38] (IEEE-2019) и видео-демонстрация[39].

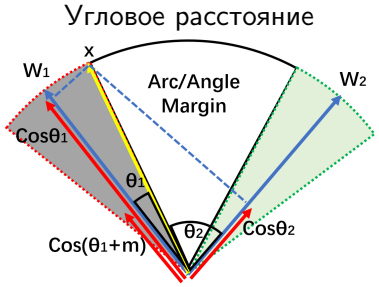


[38] Kaziakhmedov E. et al. "Real-world attack on MTCNN face detection system." 2019

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

Система распознавания лиц: ArcFace[40]

- Выбрана ведущая открытая система распознавания лиц признаков: ArcFace
- Главная идея ArcFace — использовать угловое расстояние между векторами признаков
- Огоромная обучающая база данных (MS1M) и глубокая СНС (ResNet-100)

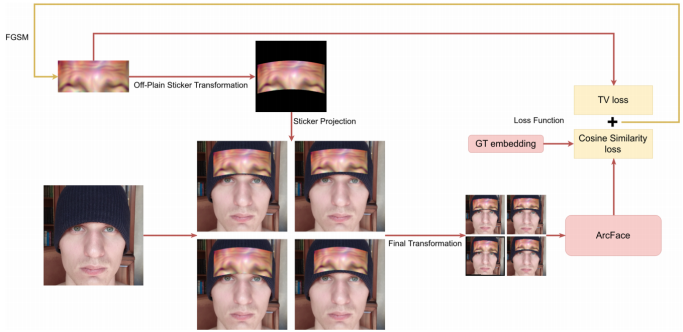


[40] Deng J. et al. "Arcface: Additive angular margin loss for deep face recognition." 2018

Состязательные примеры для распознавателя лиц ArcFace

- EOT: различные параметры проекций маски, одно изображение лица
- TV: +, NPS: –, цветовой маппинг: –
- Добавка к функции потерь $L_{sim}(x, x_{gt}) = cos(emb(x), emb(x_{gt}))$ для работы с любым лицом, где x_{gt} — фото лица, $emb(x)$ — вектор признаков x
- Оптимизатор: MI-FGSM

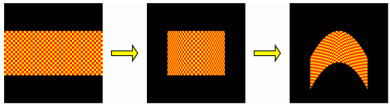
Схема генерациии состязательных примеров для ArcFace



Состязательные примеры для распознавателя лиц ArcFace

- ℓ_0-оптимизация: цветная связная область (патч) в области лба
- Глубокая СНС $\Rightarrow$ большое поле восприятия $\Rightarrow$ семантический характер примера
- Т.н. “off-plane” нелинейная проекция: $(x, y, 0) \to (x', y, z'), z' = a \cdot x'^2$

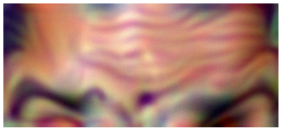
Off-plane проекция



- Дифференцируемый слой Spatial Transformer Layer[41]

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

Семантический характер примера



[41] Jaderberg M. et al. “Spatial transformer networks.” 2015

AdvHat — шапка-невидимка

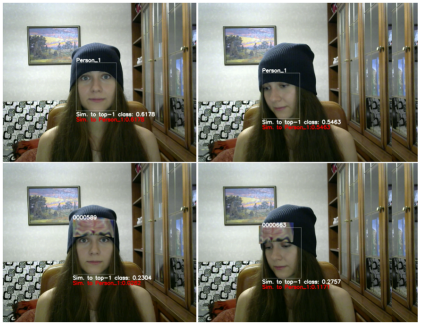
Благодаря улучшенной процедуре проекции и полного использования цветовой палитры, состязательный пример устойчив к разным поворотам и освещенности

Анфас
(AdvHat: –)
Близость (оригинал): **0.61**

|

|

Анфас
(AdvHat: +)
Близость (оригинал): **0.02**
Близость (другой): **0.23**



Профиль
(AdvHat: –)
Близость (оригинал): **0.54**

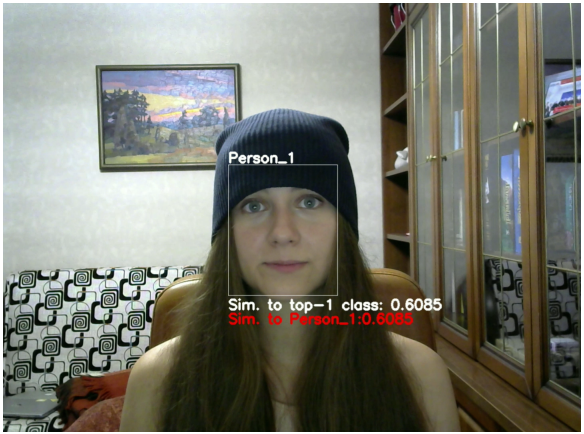
|

|

Профиль
(AdvHat: +)
Близость (оригинал): **0.11**
Близость (другой): **0.27**

Состязательные примеры для распознавателя лиц ArcFace: результат

Детали: статья[42] (ICPR-2020) и видео-демонстрация[43].



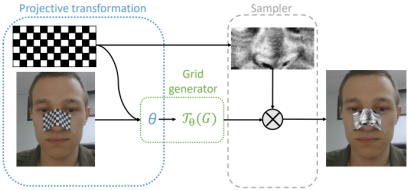
[42] Komkov S. et al. "Advhat: Real-world adversarial attack on arcface face id system." 2019

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

Состязательные примеры для распознавателя лиц ArcFace: черно-белые патчи[44] (IEEE-2019)

- Комбинация двех предыдущих подходов:
  - Цветовой маппинг (штраф за черный цвет)
  - Локальные аффинные проекции на основе подготовленной сетки
- Пример носит семантический характер

Проекция



Состязательные примеры

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

Защита от состязательных примеров для систем распознавания в реальном мире[46]

- Большинство состязательных примеров в реальном мире — патчи
  - Предложение: **A**dversarial **T**raining (AT)[45] в пиксельной области + прямоугольная аугментация
- Поэтапное AT:
  1. Позиция серого патча (max функции потерь) via:
     - Полный перебор
     - Max градиента по входу
  2. MI-FGSM внутри этого патча для поиска раскраски

- Обычная процедура обучения:

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

- Состязательное обучение (AT):

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

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

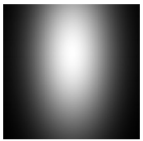
[46] Wu T. et al. "Defending Against Physically Realizable Attacks on Image Classification." 2019

Black-box восстановление лица по вектору признаков

- Black-box модель M: $M(x) = y$, где
  - x — входное фото лица
  - y — его векторное представление (эмбеддинг)
- **Задача**: восстановить x', сохраняя личность x
- **Ранее**: использование функции потерь MSE и метрик на признаках / GAN (NBNet[47])
- **Здесь**: метод оптимизации нулевого порядка для нахождения x': $M(x') \approx M(x)$
  - В качестве M используется: ArcFace
  - Тест независимым критиком: FaceNET[48]
  - Функция потерь (близость): $1 - cos(M(x'), M(x))$
  - Добавка: $(||M(x')||_2 - ||M(x)||_2)^2$

- **Главная сложность**: огромное пространство поиска
- **Решение**: использование априорной информации о лице — 2D-гауссианы

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



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

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

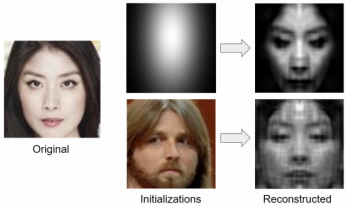
[48] Schroff F. et al. "Facenet: A unified embedding for face recognition and clustering." 2015.

Black-box восстановление лица по вектору признаков: трюки

- Даже использование априорной информации недостаточно
- **Трюк1**: Использовать вертикальную симметрию лица $\Rightarrow$ ищем только половину лица
- **Трюк2**: Для сохранения личности обычно цвет не нужен $\Rightarrow$ ищем только 1 цветовой канал вместо 3

Симметрия, без симметрии, в цвете

- **Инициализация**: Что взять за начальное приближение?
- **Обычно**: другое лицо (решение может быть смещено)
- **Можно**: оптимальный 2D-гауссиан (как раз нужна добавка на разность норм)



Black-box восстановление лица по вектору признаков: алгоритм и примеры

Алгоритм

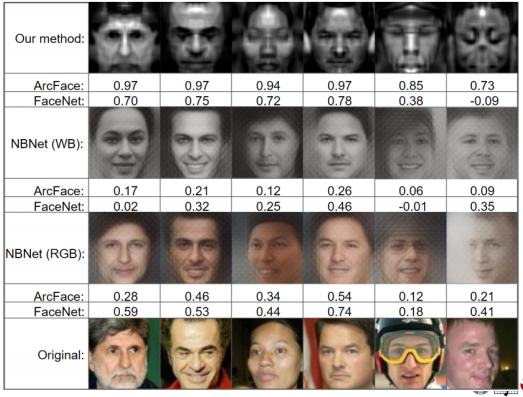
Algorithm 1 Face recovery algorithm

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
$X \leftarrow X + G_0$

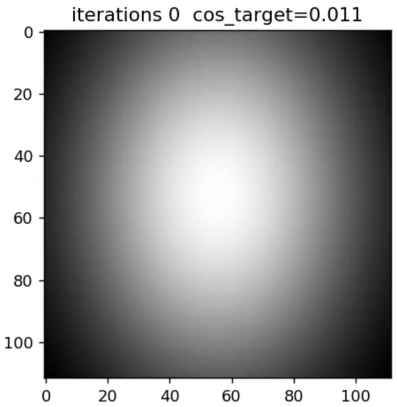
OUTPUT: reconstructed face X

Примеры



Black-box восстановление лица по вектору признаков: результат

Детали: статья[49] (ECCV-2020) и видео-демонстрация[50].



[49] Razzhigaev A. et al. "Black-Box Face Recovery from Identity Features." 2020

[50] https://www.youtube.com/watch?v=sOrTcqRTw2A

Заключительные выводы

- На данный момент СНС (в целом) работают гораздо лучше человека
- СНС неустойчивы (по входу)
- Перенести состязательный пример в реальный мир непросто
- Однако можно сломать даже супер навороченные системы распознавания лиц, имея лишь черно-белую бумажку с обычного принтера
- ℓ_0-оптимизация + EOT + TV обязательны
- Состязательное обучение может помочь в защите от состязательных примеров
- Изображения лиц могут быть восстановлены по векторам признаков даже в black-box манере

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

Вопросы к аудитории

1. В чем, на ваш взгляд, главная причина существования состязательных примеров (кроме приведенной ранее)?

Вопросы к аудитории

1. В чем, на ваш взгляд, главная причина существования состязательных примеров (кроме приведенной ранее)?
2. Приведите нетривиальные примеры борьбы с проклятием размерности в реальных задачах компьютерного зрения

Вопросы к аудитории

1. В чем, на ваш взгляд, главная причина существования состязательных примеров (кроме приведенной ранее)?
2. Приведите нетривиальные примеры борьбы с проклятием размерности в реальных задачах компьютерного зрения
3. Зачем нужны проекции для генерации реальных примеров, и зачем через них пропускать градиенты?

Вопросы к аудитории

1. В чем, на ваш взгляд, главная причина существования состязательных примеров (кроме приведенной ранее)?
2. Приведите нетривиальные примеры борьбы с проклятием размерности в реальных задачах компьютерного зрения
3. Зачем нужны проекции для генерации реальных примеров, и зачем через них пропускать градиенты?
4. Какие бы вы предложили варианты физических примеров для интересной практической задачи (кроме систем распознавания/детекции лиц) и, соответственно, защиты от них?

Вопросы к аудитории

1. В чем, на ваш взгляд, главная причина существования состязательных примеров (кроме приведенной ранее)?
2. Приведите нетривиальные примеры борьбы с проклятием размерности в реальных задачах компьютерного зрения
3. Зачем нужны проекции для генерации реальных примеров, и зачем через них пропускать градиенты?
4. Какие бы вы предложили варианты физических примеров для интересной практической задачи (кроме систем распознавания/детекции лиц) и, соответственно, защиты от них?
5. Какие вообще перспективы биометрии в важных приложениях? Чем можно заменить?

Спасибо за внимание!