Состязательная атака на нейронную сеть YOLO

Н.В. Тетерев [1], В.Е. Трифонов [1], А.Б. Левина [1]

[1]Санкт-Петербругский государственный электротехнический университет «ЛЭТИ»,
Россия, 197022, г. Санкт-Петербург

Аннотация. В работе приведено системное исследование формализованных, для нейросетевых алгоритмов компьютерного зрения семейства YOLO, условий генерации универсальных состязательных примеров – изображений, ложно классифицируемых алгоритмами компьютерного зрения. Была выявлена и изучена закономерность успешной генерации универсального состязательного примера с использованием предложенной математической модели алгоритма. Результаты демонстрируют зависимость эффективности атаки от набора данных, на котором обучены нейронные сети, с применением быстрой знаково-градиентной атаки (FGSM). Экспериментальная часть исследования охватывает модели YOLO версий 8-11, обученных на стандартном наборе данных COCO. Работа демонстрирует возможность создания состязательного примера, успешно воздействующего на несколько нейронных сетей одновременно.

Ключевые слова: Состязательная атака, состязательный пример, YOLO, нейронные сети, информационная безопасность.

Abstract. This paper presents a systematic study of formalized conditions for generating universal adversarial examples – images that are misclassified by computer vision algorithms – for YOLO family neural networks. The research identifies and examines patterns in successful generation of universal adversarial examples using the proposed mathematical model of the algorithm. The results demonstrate the dependence of attack's effectiveness on the training dataset when applying the Fast Gradient Sign Method (FGSM). The experimental study covers YOLO versions 8-11 trained on the standard COCO dataset. The work demonstrates the feasibility of creating adversarial examples that simultaneously affect multiple neural networks.

Keywords: Adversarial attack, adversarial example, YOLO, neural network, information security.

Введение

Современные высокопроизводительные вычислительные устройства способны эффективно обрабатывать большие объёмы данных и обучать сложные нейронные сети, что сделало глубокое обучение ключевым инструментом в различных прикладных областях. В частности, рассматриваемая технология позволила значительно повысить точность решения таких задач, как классификация изображений, распознавание речи, машинный перевод и других направлений, требующих высокой точности обработки информации. Например, глубокие нейронные сети успешно применяются в анализе молекулярных структур при разработке новых фармацевтических препаратов, где точность прогнозирования критически важна.

Благодаря способности обучаться на огромных массивах данных, нейронные сети демонстрируют исключительную эффективность в решении широкого круга задач, что сделало их основой для многих современных приложений и сервисов и привело к их активному использованию в критически важных с точки зрения безопасности областях, таких как автономные транспортные средства, системы кибербезопасности, беспилотные летательные аппараты и робототехнические комплексы.

В настоящее время алгоритмы компьютерного зрения в значительной степени опираются на методы глубокого обучения. По оценкам «Market Research Future», в 2017 году объём мирового рынка компьютерного зрения составил 9,2 млрд. долларов, а к 2023 году он превысил 48,3 млрд. долларов, причём после 2020 года наблюдается устойчивый рост [1], что проиллюстрировано на рисунке 1.

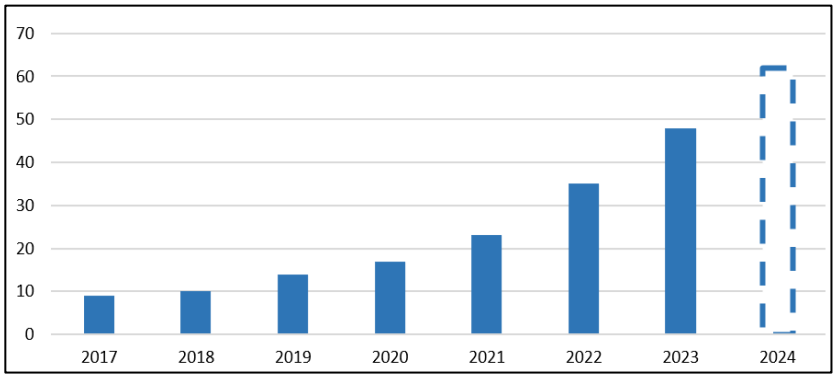

Рисунок 1 – Рост рынка компьютерного зрения в мире, млрд долларов

Однако подобная интеграция нейронных сетей в реальные системы вызывает серьёзные опасения относительно безопасности и целостности. В связи с этим растёт число исследований, посвящённых угрозам, влияющим на работоспособность нейронных сетей [1, 2]. Атаки, рассматриваемые в данной статье, используются злоумышленниками, жертвами которых могут стать любые организации, использующие в своей работе нейросетевые алгоритмы компьютерного зрения [3], множество различных атак на которые активно исследуются в научных работах [4], и основная часть которых была обнаружена в период с 2016 по 2020 год. Данные атаки делятся на два основных класса [5]:

- «Белый ящик» – злоумышленник обладает полной информацией о модели, что позволяет досконально изучить существующие уязвимости исследуемого объекта;
- «Чёрный ящик» – атакующий обладает ограниченными знаниями, например, общедоступной информацией о цели исследования, его сети и параметрах, либо не обладает ими вовсе.

Атаки «чёрного ящика», в отличие от «белого ящика», максимально приближены к реальным атакам и являются наиболее сложными, поскольку злоумышленнику необходимо полагаться исключительно на внешние наблюдения и запросы, чтобы создать эффективные примеры враждебного поведения, не имея доступа к внутренним параметрам системы.

Идея проведения атак на нейросетевые алгоритмы компьютерного зрения заключается в генерации состязательного примера. Состязательный пример – это изображения, содержащие малозаметные для человеческого восприятия искажения, которые приводят к некорректной работе модели – классифицирующая нейронная сеть распознаёт объект на изображении иначе, чем его видит человеческий глаз. Такие изображения получаются наложением сгенерированного шума на исходное изображение – они необходимы для проведения состязательной атаки. В контексте данной статьи под шумом понимаются данные, которые относятся к целевому классу распознавания практически со стопроцентной вероятностью. Схематично создание состязательного примера показано на рисунке 2.

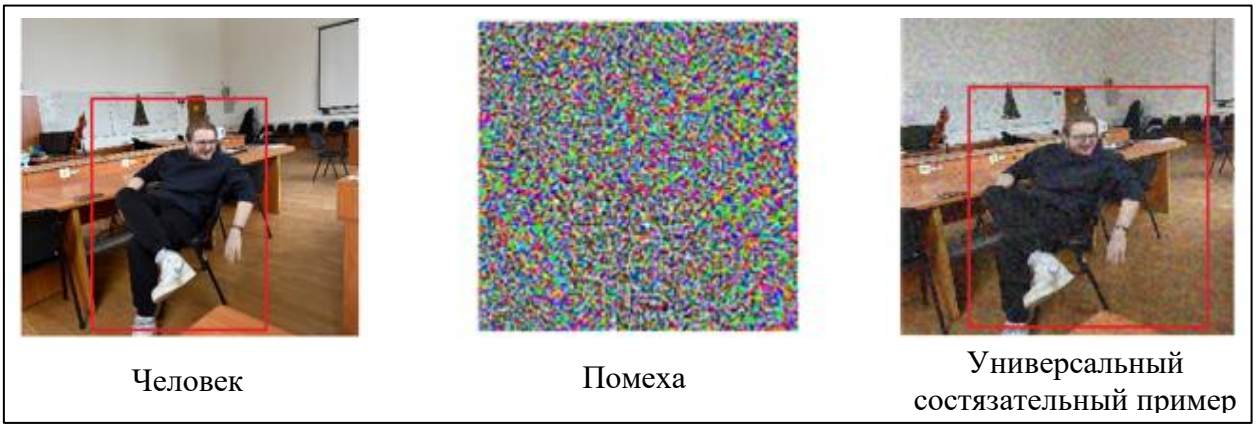

Рисунок 2 – Создание состязательного примера

В классических сценариях состязательный пример создаётся для конкретного изображения и нейросетевого алгоритма компьютерного зрения. Такие примеры, как правило, неэффективны при попытке атаковать системы обнаружения, использующих два и более нейросетевых алгоритма одновременно. Данный фактор сужает область применения созданного состязательного примера и делает его неэффективным в атаках на системы обнаружения, использующих две и более нейронные сети.

В данной работе описывается создание универсального состязательного примера для различных нейросетевых алгоритмов компьютерного зрения семейства YOLO, представленных на рисунке 3.

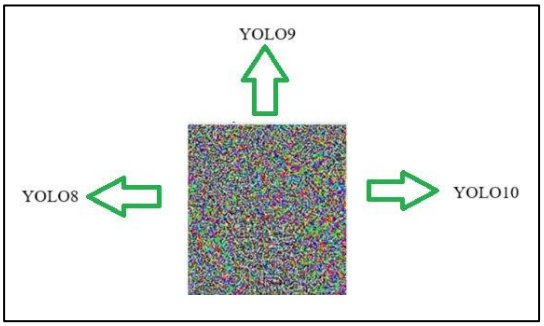

Рисунок 3 – Нейронные сети, для которых подходит универсальный состязательный пример

Нейронные сети отличаются друг от друга не только по архитектуре, но и по способу обучения. Наибольшее распространение получил метод «обратного распространения

ошибки», основанный на вычислении градиента функции потерь [7]. При обучении нейронных сетей этот градиент используется для минимизации ошибки предсказания модели. Описанный метод нашёл применение в процессе обучения нейронных сетей в данной работе.

1. Математическая модель алгоритма

Одним из наиболее известных классов атак на компьютерное зрение является атака на градиент функции потерь. Поиск градиента при обучении нейронных сетей проиллюстрирован на рисунке 4.

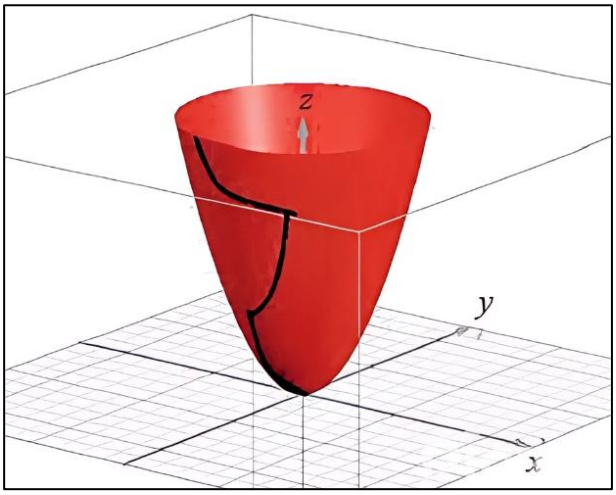

Рисунок 4 – Графическая иллюстрация метода обучения градиентного спуска

На каждой итерации поиска минимума функции ошибки вычисляется градиент и сравнивается с предыдущим значением. Если предыдущее значение больше, то направление спуска остаётся прежним, а если меньше, то направление меняется.

Основаня идея градиентных атак заключается в уязвимости метода обратного распространения ошибки – алгоритма, используемого для обучения нейронной сети. Одной из наиболее распространенных градиентных атак является Fast Sing Gradient Method (FGSM).

FGSM [7] представляет собой эффективный метод генерации состязательных примеров, использующий уязвимость нейронных сетей к целенаправленным малым возмущениям входных данных. Суть метода заключается в том, что даже минимальные, специально подобранные изменения изображения (порядка ϵ) способны кардинально изменить результат классификации, оставаясь при этом визуально незаметными для человека. FGSM описывается следующим образом:

$$\eta = \epsilon \cdot sign\big(\nabla_x J(\theta, x, y,)\big),$$

где J – функция потерь;

θ – параметр модели;

x – входные данные для модели;

y – цель;

ϵ – коэффициент, определяющий размер «возмущения»;

sing – знак каждого элемента вектора, определяется знаком входного градиента в элементе.

В ходе атаки берётся минимальный градиент функции потерь, и на его основе вычисляется состязательный пример. В отличие от итеративных методов, требующих многократного пересчёта, FGSM выполняет генерацию за один проход, что обеспечивает выигрыш в скорости. Ключевые особенности метода:

1) Относится к классу атак «белого ящика» – предполагает полную доступность архитектуры и параметрам целевой модели нейронной сети;
2) Использует обратное направление градиента (антиградиент) для нахождения оптимальных возмущений;
3) Оптимизирует возмущения в направлении максимального роста функции потерь.

Градиентные атаки также включают в себя:

- CW (Carlini-Wagner) [9] – формулирует проблему генерации примеров как задачу оптимизации с ограничениями. Основаня цель – найти наименьшее возмущение входных данных, которое, при добавлении к исходным данным x, приводит к их ошибочной классификации целевой моделью f. Математически это выражается следующей целевой функцией:

$$J(x') = \alpha \cdot dist(x, x') + \beta \cdot loss(f(x'), y_t),$$

где α, β – положительные константы,

x – вектор входных данных;

x' – вектор возмущённых данных;

$\mathrm{d}ist \in \{L_2, L_\infty\} -$ измерение возмущений, где L_2, L_∞ – кратные нормы;

$loss(f(x'), y_t) -$ потери из-за неправильной классификации целевой модели f в результате добавления возмущённых входных данных относительно целевого класса y_t.

- ZOO (Zero-Order Optimization) [10] – атака на нейросетевые модели в условиях ограниченной информации, когда информация о градиентах недоступна, например, в сценариях оптимизации «чёрного ящика», где доступны только оценки функций (выходы) без прямого доступа к градиентам.
- SimBA (Simple Black-Box Attack) [11] – относится к классу атак «черного ящика» и реализует эффективный подход к генерации состязательных примеров без доступа к внутренним параметрам модели. На каждой итерации во входные данные вносятся локальные возмущения, после чего анализируется реакция модели. Такой подход позволяет исследовать пространство входных данных и находить оптимальные направления для атаки без знания архитектуры нейронной сети, её градиентов или параметров обучения. Процесс можно описать следующим образом:

$$\left(P_x, P_y\right)_i = \bigcup_{\{(a,b)\in\left(P_X, P_Y\right)_{i-1}\}} \bigcup_{\substack{\{x\in[a-d,a+d],\\ y\in[b-d,b+d]\}}} (x, y),$$

где $P_x -$ значение входного слоя;

$P_y -$ значение объективной функции для выходного слоя;

$d -$ параметр, определяющий область возмущения (половина стороны квадрата);

$a, b -$ координаты значения пикселей, подвергаемых модификации.

Предложенный алгоритм создаёт универсальный состязательный пример и начинает работать аналогично атаке FGSM. На первом этапе вычисляются векторы знаков градиентов функции потерь:

$$\bar{n} = sign\big(\nabla_x J(\theta, x, y)\big),$$

где J – функция потерь, используемая для повышения точности распознавания при обучении нейронной сети;

θ – параметр модели;

x – вектор входных активаций для модели;

y – результирующий вектор нейронной сети в случае ошибочной классификации объекта, называемого целью;

sign – знак каждого элемента вектора определяется знаком градиента входа в элемент;

$\bar{n} = \{n_i\}, n_i \in \{-1,0,1\}, i \in \mathbb{N}$ – градиент функции потерь для каждой нейронной сети.

Полученный вектор умножается на параметр, определяющий величину возмущений, и используется для создания состязательного примера, подходящего для конкретной атакуемой нейронной сети. В контексте описанного алгоритма, следующим шагом является адаптация этого вектора для создания универсального состязательного примера. Минимизация изменений в рассматриваемом векторе приведёт к ошибочной классификации для нескольких алгоримов компьютерного зрения.

Для фиксации параметров, отвечающих за успешное создание состязательного примера, вычисляется расстрояние Хэмминга [12] относительно вектора знаков градиента, полученного при рассчёте для первой нейронной сети $\overline{n_1}$, и аналогичного вектора, вычесленного для следующей нейронной сети $\overline{n_2}$.

$$h = d(\overline{n_1}, \overline{n_2}) = \sum_i \delta\big(n_{i,1}, n_{i,2}\big),$$

где функция δ, в контексте решаемой задачи, вычисляется по формуле:

$$\delta\big(n_{i,1}, n_{i,2}\big) = \begin{cases} 1, n_{i,1} = n_{i,2}; \\ 0, n_{i,1} \neq n_{i,2}. \end{cases}$$

На следующем шаге необходимо найти вектор из полученных векторов $\overline{n'}$, который содержит максимальное количество элементов из $\overline{n_1}$ и $\overline{n_2}$, расположенных в соответствующих местах. На первой итерации предлагается использовать вектор $\overline{n'} = \overline{n_1}$ и функцию η:

$$\eta(\overline{n_1}, \overline{n_2}) = \begin{cases} n_{i,1}, n_{i,1} = n_{i,2}; \\ n_{i,1}, n_{i,1} \neq n_{i,2}. \end{cases}$$

В последующих итерациях для создания универсального состязательного примера для двух нейронных сетей необходимо изменять вектор таким образом, чтобы расстояние Хэмминга между предыдущими значениями вектора знаков функции потерь и аналогичным вектором, созданным для новой атакуемой нейронной сети h'_{i+1}, отличалось на минимальную величину или оставалось постоянным. Таким образом, можно сделать вывод, что в контексте решаемой задачи требуется минимизировать изменение расстояния Хэмминга относительно предыдущей итерации для каждой нейронной сети:

$$h_i - h'_{i+1} \to min.$$

Чтобы адаптировать вектор знаков градиентов функции потерь, необходимо изменить те элементы, которые не входят в $\overline{n_1}$ и $\overline{n_2}$:

$$\eta(\overline{n_1}, \overline{n_2}): \eta_i \in \overline{n_1} \cup \eta_i \in \overline{n_2}.$$

На основе полученного вектора строится состязательный пример для второй нейронной сети и выполняется следующая итерация алгоритма, где a – вектор знаков градиентов функции потерь второй нейронной сети.

2. Практические результаты

В рамках исследования, нейронные модели YOLO, классифицирующие объекты на изображении, были протестированы на устойчивость к созданию универсальных состязательных примеров для FGSM-атак. Архитектурные различия между версиями нейронных сетей YOLO существенно ограничивают возможность создания единого состязательного примера для всех моделей одновременно. Однако при обучении на общем наборе данных (COCO) наблюдается значительная близость минимальных значений градиента функции потерь относительно друг друга, как показано на рисунке 5.

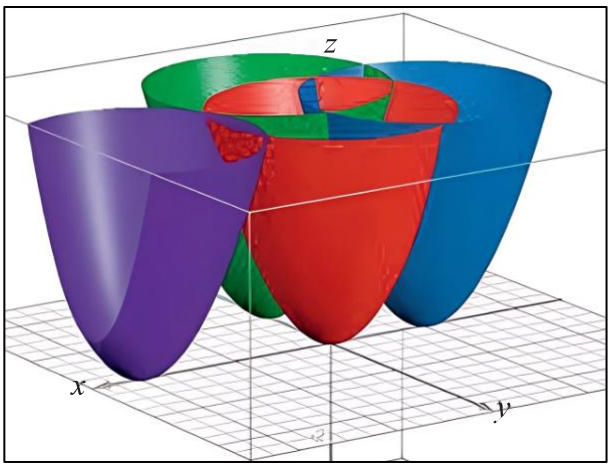

Рисунок 5 – Расположение минимумов градиента для различных сетей: YOLO 11 (фиолетовый), YOLO 10 (красный), YOLO 9 (синий), YOLO 8 (зелёный)

Представленная трёхмерная визуализация демонстрирует распределение градиентов функции потерь для исследуемых моделей. Для YOLO 8-10 обнаружена значительная близость глобальных минимумов и наблюдается область пересечения градиентов, потенциально пригодная для генерации универсального состязательного примера, который будет применим к различным нейросетевым алгоритмам компьютерного зрения. В то же время, минимум градиента функции потерь YOLO 11 существенно удалён от других моделей, что исключает, при использовании предложенного решения, возможность генерации универсального состязательного примера, который подойдёт для всех четырёх нейронных сетей [13].

Описанная последовательность действий реализована в текущем исследовании для моделей нейронный сетей YOLO, которые имеют схожую архитектуру (YOLO 8-11). Модели обучались на одном и том же наборе данных из Common Objects in Context (COCO) – наборе данных из 330 000 изображений, 200 000 из которых содержат аннотации для задач обнаружения, сегментации и создания меток объектов. Для успешного создания

универсального состязательного примера вычисляется расстояние Хэмминга между векторами знаков градиента функции потерь, которое представлено в таблице 1.

Таблица 1 – Отношение расстояния Хэмминга между моделями к количеству нейронов во входном слое нейронной сети

$\frac{h}{\lvert YOLO \rvert}$	YOLO 8	YOLO 9	YOLO 10	YOLO 11
YOLO 8	0			
YOLO 9	0,43	0		
YOLO 10	0,31	0,22	0	
YOLO 11	0,56	0,63	0,79	0

Экспериментальные результаты продемонстрировали возможность идентификации общей точки пересечения градиентов для трёх из четырёх исследуемых моделей YOLO (8-10), что позволяет создать универсальный состязательный пример, работающий против нескольких алгоритмов компьютерного зрения. Как показано в результатах, процент ложной классификации варьируется в зависимости от коэффициента возмущения ϵ, сохраняя при этом визуальную незаметность изменений для человеческого восприятия. Анализ выявил, что обнаруженная точка общего минимума градиента находится в непосредственной близости от градиента модели YOLO 9, тогда как для YOLO 11 она оказалась неприменимой из-за существенных архитектурных изменений, внесённых разработчиками ultralytics и повлекших смещение положения точки глобального минимума градиента функции потерь [14].

Выявленная закономерность подтверждается данными таблицы 2, где значения коэффициента расстояния Хэмминга между векторами градиентов функции потерь демонстрируют значительную близость для моделей YOLO 8-10 и заметное расхождение для YOLO 11, что визуально соответствует пространственному распределению локальных минимумов, представленному на рисунке 4.

Таблица 2 – Отношение вектора знаков градиента функции потерь для универсального состязательного примера к количеству нейронов входного слоя

Название модели	
YOLO 8	0,82
YOLO 9	0,76
YOLO 10	0, 68
YOLO 11	0,14

В результате работы алгоритма создания универсального состязательного примера, рассмотренного в данной работе, было сгенерировано изображение, которое ложно классифицируется нейронными сетями YOLO 8-10 [15], что подтверждает уязвимость этих архитектур к состязательной атаке. Кроме того, даже при высоком уровне шума изображения, модель YOLO 11 с низкой вероятностью указывает на объект ложного обнаружения как на один из вариантов распознавания, что показано в таблице 3 [14].

Таблица 3 – Коэффициент ложного распознавания неблагоприятного примера, %

Название модели	ϵ		
	0,07	0,10	0,15
YOLO 8	38	46	59
YOLO 9	63	72	78
YOLO 10	0	24	45
YOLO 11	0	0	2

Полученные результаты согласуются с теоретическими выводами о структурных различиях в архитектуре YOLO 11, делающих его менее восприимчивым к универсальным состязательным примерам, эффективным против предыдущих версий нейронной сети.

Заключение

Предложенный в исследовании метод генерации универсальных состязательных примеров обладает значительным практическим потенциалом для систем, в которых изображения обрабатываются несколькими нейросетевыми алгоритмами компьютерного зрения. В подобных условиях традиционные состязательные примеры, создаваемые для конкретной модели, теряют свою эффективность при применении к другим нейронным сетям. Особенно актуален данный подход в сценариях многоэтапной проверки контента, когда изображение проходит через последовательный анализ различными нейронными сетями, либо при регулярных обновлениях архитектур на интернет-ресурсах, направленных, например, на автоматическое выявление запрещённой информации на изображении – для обнаружения подобной информации нередко применяется сразу несколько алгоритмов компьютерного зрения.

Разработанный алгоритм позволяет создавать состязательные примеры, способные обходить несколько моделей семейства YOLO, оставаясь незаметным для стандартных механизмов обнаружения. Экспериментальная часть работы показала возможность применения атаки FSGM для генерации универсального состязательного примера, нарушающего работу моделей YOLO 8-10. Проведённый анализ также выявил принципиальные ограничения метода, зависящие от расстояний Хэмминга относительно знаков градиентов функции потерь – архитектурные изменения в YOLO 11 привели к смещению минимума функции потерь, что существенно снизило результативность состязательной атаки.

Ключевым достижением исследования стало установление зависимости между структурой обучающего набора данных, конфигурацией градиентных пространств и возможностью генерации универсального состязательного примера с помощью атаки FSGM. Полученные результаты раскрывают особенности нового вектора атак на системы компьютерного зрения и формируют основу для разработки устойчивых архитектур и стратегий обучения нейронных сетей, способных противостоять подобным воздействиям.

Благодарности

Работа выполнена в рамках государственного задания Министерства науки и высшего образования Российской Федерации № 075-00003-24-01 от 08.02.2024 (проект FSEE-2024-0003).

Список литературы

1. Анализ уязвимости нейросетевых моделей YOLO к атаке Fast Sign Gradient Method / Тетерев Н.В., Трифонов В.Е., Левина А.Б. // Научно-технический вестник информационных технологий, механики и оптики. Т. 24, №6. с. 1066-1070. doi: 10.17586/2226-1494-2024-24-6-1066-1070;
2. Adversarial attacks and defences: A survey / Chakraborty A., Alam M., Dey V., Chattopadhyay A., Mukhopadhyay D. // arXiv. 2018, arXiv:1810.00069v1. URL: https://doi.org/10.48550/arXiv.1810.00069;
3. Threat of adversarial attacks on deep learning in computer vision: A survey / Akhtar N., Mian A. // IEEE Access. 2018, Vol. 6, pp. 14410–14430. URL: https://doi.org/10.1109/access.2018.2807385;
4. Explaining and harnessing adversarial examples / Goodfellow I., Shlens J., Szegedy C. // Proc. of the 3rd International Conference on Learning Representations, ICLR, 2015;
5. An efficient adversarial attack for tree ensembles / Zhang C., Zhang H., Hsieh C.-J. // Advances in Neural Information Processing Systems, 2020, vol. 33;
6. It is all about data: A survey on the effects of data on adversarial robustness / Xiong P., Tegegn M., Sarin J.S., Pal S., Rubin J. // ACM Computing Surveys, 2024, Vol. 56, No. 7, pp. 1–41. URL: https://doi.org/10.1145/3627817;
7. Regularization effect of fast gradient sign method and its generalization / Zuo C. // arXiv, 2018, arXiv:1810.11711. URL: https://doi.org/10.48550/arXiv.1810.11711;
8. Understanding neural networks through deep visualization / Yosinski J., Clune J., Nguyen A., Fuchs T., Lipson H. // arXiv, 2015, arXiv:1506.06579v1. URL: https://doi.org/10.48550/arXiv.1506.06579;
9. Towards evaluating the robustness of neural networks / Carlini N., Wagner D. // Proc. of the IEEE Symposium on Security and Privacy (SP), 2017, pp. 39–57. URL: https://doi.org/10.1109/sp.2017.49;
10. Zeroth-order optimization for composite problems with functional constraints / Li Z., Chen P.-Y., Liu S., Lu S., Xu Y. // Proceedings of the AAAI Conference on Artificial Intelligence, 2022, vol. 36, no. 7, pp. 7453–7461. URL: https://doi.org/10.1609/aaai.v36i7.20709;
11. Simple black-box adversarial attacks / Guo C., Gardner J., You Y., Wilson A., Weinberger K. // Proceedings of Machine Learning Research, 2019, vol. 97, pp. 2484–2493;
12. Быстрая реализация расстояния Хэмминга на VLIW-архитектурах на примере платформы Эльбрус / Лимонова Е.Е., Рженев Н.Л., Усков А.В., Нейман-заде М.И. // «Труды Института системного анализа Российской академии наук», 2018;
13. Building Towards «Invisible Cloak»: Robust Physical Adversarial Attack on YOLO Object Detector / Yang D. Y., Xiong J., Li X., Yan X., Raiti J., Wang, Y. // IEEE Annual Ubiquitous Computing, Electronics & Mobile Communication Conference, 2018 pp. 368-374;
14. Physical adversarial examples for object detectors / Song D., Eykholt K., Evtimov I., Fernandes E., Li B., Rahmati A. // USENIX workshop on offensive technologies, 2018;
15. Adversarial attack detection via fuzzy predictions / Li Y., Angelov P., & Suri N. // IEEE Transactions on Fuzzy Systems, 2024.