ОЦЕНКА УСТОЙЧИВОСТИ НЕЙРОСЕТЕВЫХ МЕТОДОВ РАСПОЗНАВАНИЯ ОБЪЕКТОВ НА СНИМКАХ КОСМИЧЕСКИХ СИСТЕМ ДИСТАНЦИОННОГО ЗОНДИРОВАНИЯ ЗЕМЛИ

Ксендзук А. В.
Институт радиоэлектроники и информатики
Заведующий кафедрой радиоинформационных систем
РТУ МИРЭА
Москва, Россия
ks_alex@mail.ru

Мелкумян М. К.
ПАО «МАК «Вымпел»
Инженер-программист отдела моделирования и разработки СПО Центра перспективных проектов
Москва, Россия
melkumyanmarat9@gmail.com

***Аннотация* — В работе приведена оценка устойчивости нейросетевых методов распознавания объектов на снимках космических систем ДЗЗ. Обоснована актуальность использования нейросетевых методов, а также приведен список наиболее подходящих нейросетей. Исследованы и описаны методы запутывания нейронных сетей при помощи искажений, с целью последующей неверной классификации исследуемых объектов. Получены результаты к каждому из используемых методов.**

Ключевые слова* — *система контроля дистанционного зондирования земли, классификация объектов, метод запутывания, машинное обучение, распознавание образов, искусственная нейронная сеть, алгоритм определения эффективных искажающих элементов.

Введение

Иностранные государства обладают большим количеством космических средств оптической и радиолокационной разведки. В настоящее время на орбите находится более 480 спутников оптической разведки и более 90 спутников радиолокационной разведки и их количество постоянно растет, рисунок 1.

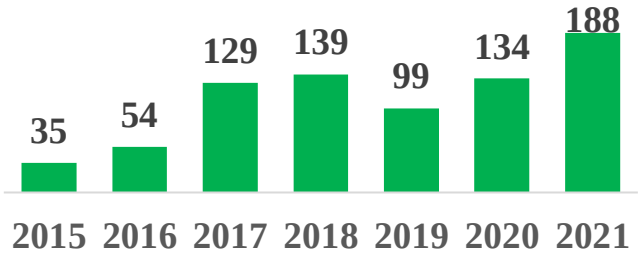


Рис.1. Число запусков спутников дистанционного зондирования (по годам)

Учитывая большие объемы информации, которые формируются этими группировками, распознавание объектов ведется с применением автоматических или полуавтоматических методов, основанных в том числе на применении нейросетей

Вероятно, что для этого применяются нейросети, которые хорошо себя зарекомендовали при обработке изображений – Meta Pseudo Label, EfficinetNet-B7 и другие. Существует ежегодно обновляемый рейтинг нейросетей, который позволяет определить те из них, которые обеспечивают наилучшее качество распознавания, таблица 1.

ТАБЛИЦА 1. НАИБОЛЕЕ ЭФФЕКТИВНЫЕ ДЛЯ ЗАДАЧ РАСПОЗНАВАНИЯ НЕЙРОСЕТИ 2023Г.

Нейронная сеть	Вероятность ложной классификации	Количество параметров
AlexNet	17 %	60 M
GoogLeNet	9.20 %	5 M
ResNet-152	4.49 %	60 M
EfficinetNet-B7	2.90 %	66 M
Meta Pseudo Label	1.20 %	480 M

Во многих практических задачах необходимо определить устойчивость классификации объектов нейросетью при наличии на исходных данных искажающих элементов. Под устойчивостью понимаем умение правильно относить изображения к нужным классам, при наличии искажающих элементов. Это необходимо как для оценки качества функционирования нейросети, так и для определения элементов, которые с наибольшей эффективностью исказят изображения так, что определенные объекты будут классифицированы неверно.

I. МЕТОДИКИ ЗАПУТЫВАНИЯ НЕЙРОСЕТЕЙ

Adversarial Machine Learning (AML) - «состязательное машинное обучение», подразумевая целенаправленное воздействие на нейронную сеть, которое способно вызвать ошибки в ее поведении.

Традиционной иллюстрацией AML-атаки(Adversarial attack) является пример [3], когда к исходному изображению панды, которое распознается с вероятностью 57,7 %, добавляется специальные шум, невидимый человеком, но замечаемый нейросетью. В результате нейронная сеть распознаёт изображение как гиббона с вероятностью 99,3 % (рис.2).

Вероятно, это одна из первых работ, где продемонстрировано, как исказить пиксели изображения, чтобы классификатор принял ошибочное решение. В основе метода лежит факт, что изображения обычно представлены в виде 8-битных значений.

Следует определить ошибочную классификацию входных данных уравнением:

$$\omega^T \hat{\hat{x}} = \omega^T \mathrm{x} + \omega^T \eta \quad (1)$$

Где:

$\hat{\hat{x}}$ — входные данные, нужные для введения нейросети в заблуждение,

ω^T — выходные данные классификатора по неизменённому изображению,

η — особенный вектор, добавленный к исходным входным данным таким образом, чтобы вся сеть приняла ошибочное решение о классификации.

Механизм определения η следующий:

$$\mathrm{sign}(\nabla_X J(\theta, \mathrm{x}, \mathrm{y})) \quad (2)$$

Где:

sign() — знаковая функция (sign function). Она отвечает за знак значения. Для положительного значения функция равна 1, для отрицательного –1.

∇x — градиенты ,

J — функция стоимости (cost function), используемая для обучения нейросети,

θ — параметры модели,

x — входные данные,

y — целевые выходные данные, то есть «ошибочный» класс.

Поскольку вся сеть является дифференцируемой, значения градиента можно легко найти с помощью метода обратного распространения ошибки.

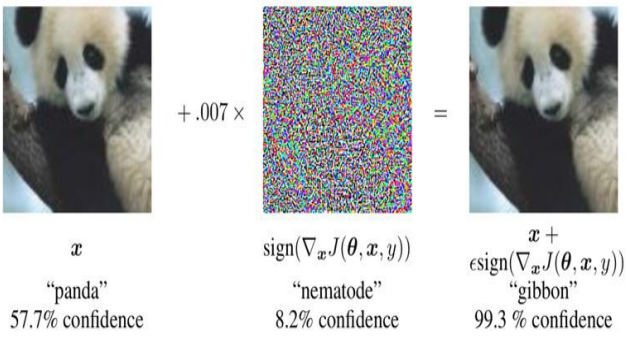


Рис.2. Принцип AML-атаки на ML-системы распознавания изображений

Следовательно, изменив входные данные и выяснив с помощью анализа, какое направление нужно изменить (применив информацию о градиентах), можно легко заставить сеть неправильно классифицировать изображение.

Одной из главных причин появления Adversarial attack ялвяется то, что методы МО были созданы для стационарных и безопасных сред, где обучающая и тестовая выборки сгенерированы из одного и того же статистического распределения. На практике является возможным манипулирование входными данными, чтобы использовать уязвимости ML-алгоритмов и поставить под угрозу безопасность всей системы машинного обучения. Выделяют 2 вида AML-атак.

1. Уклонение - используются для неадекватного поведения уже готового продукта со встроенной в него ML-моделью.

2. Отравление - получение доступа к данным и процессу обучения ML-модели, чтобы ее «отравить» (задать неправильное обучение) для последующей неправильной работы

Основополагающими факторами, определяющие вид атаки на контролируемые ML-алгоритмы являются нижеперечисленные:

- ***влияние на классификатор*** - атака направлена на внедрение уязвимостей на этапе классификации путем манипулирования обучающими данными или поиск и последующее использование уязвимостей;

- ***нарушение безопасности***, - когда искаженные образцы ошибочно классифицированы как легитимные цель состоит в том, чтобы увеличить неправильную классификацию легитимных образцов;

- ***особенность атаки*** – целевая или нецелевая. При целевой атаке используются конкретные образцы для разрешения конкретного вторжение.

Похожая по направлению задача определения искажающих элементов решалась в основном для препятствия распознаванию лиц [1,2]. В результате были созданы методы, которые с высокой эффективностью препятствуют распознаванию объектов, реализованные в том числе в виде мобильных приложений(Camera Adversaria), рисунок 3.

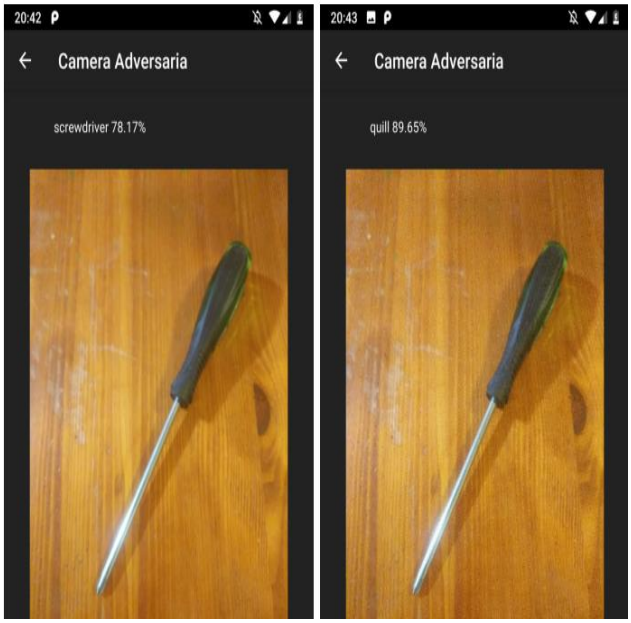


Рис. 3. Слева: фотография в галерее до искажения. Верная классификация - отвертка. Справа: та же фотография после искажения, результат классификации - ручка.

Как показывает анализ работ, наиболее простым и эффективным методом является добавлении шума Перлина, представляющим собой процедурную текстуру [3], рисунок 4.

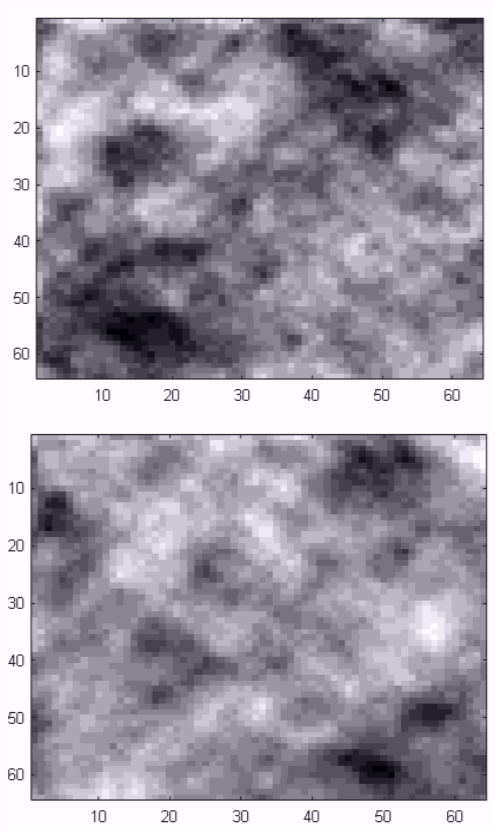


Рис. 4. Примеры шума Перлина, сформированные в Octave.

Такой подход позволяет использовать новый способ маскировки объектов – наносить не только стандартные маскировочные узоры, но дополнительно искажать их шумом Перлина так, чтобы дополнительно снизить вероятность правильной классификации нейросетями.

Существуют методы снижения правильной классификации нейросетью при использовании «атак на нейросети»[4]. Одна из распространенных вариаций – «One pixel attack» (атака одного пикселя). Цель данной атаки заставить алгоритм (нейросеть) выдать некорректный ответ. Используя метод многомерной математической оптимизации, а именно - дифференциальную эволюцию, находится особенный пиксель, способный изменить изображение так, чтобы нейросеть стала неправильно классифицировать это изображение (несмотря на то, что ранее алгоритм классифицировал этот же объект верно и с высокой точностью).

Рис. 5. Сверху: оригинальная картинка, снизу: изменение одного пикселя.

После загрузки моделей нужно оценить тестовые изображения для каждой модели, чтобы убедиться, что мы атакуем только изображения, чья классификация была корректной.

Поиск пикселя, который способен заставить алгоритм выдать неверный результат формулируется как задача оптимизации: при «нецелевой атаки» нужно минимизировать доверие к нужному классу, а при «целевой атаке» - максимально увеличить доверие к целевому классу.

Результаты показали, что применяющиеся сверточные архитектуры нейросетей уязвимы перед специально обученным алгоритмом One pixel attack. [5, 6].

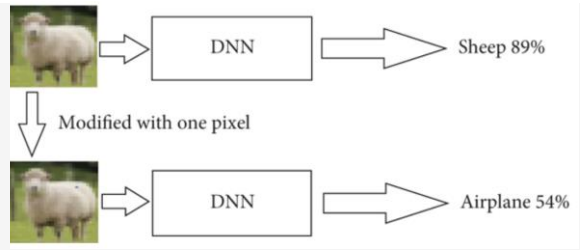


Рис. 6. Результат примера атаки одного пикселя.

На примере используется дифференциальная эволюция (Differential Evolution). При использовании такого метода используются образцы, на основе которых генерируются «дочерние» образцы, а из них потом оставляют лишь те, что получились лучше «родительских». Далее осуществляется новая итерация формирования «дочерних» образцов.

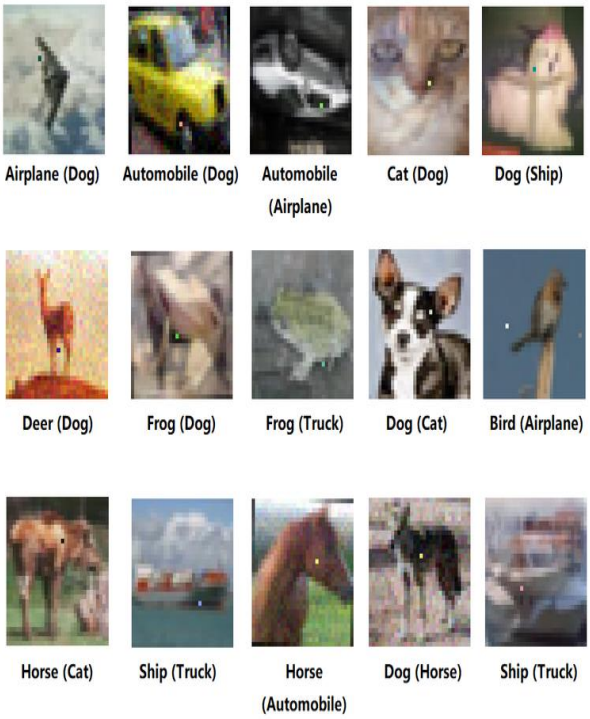


Рис. 7. Результат однопиксельной атаки. В картинках изменено всего по одному пикселю, и в результате нейросеть классифицировала их неправильно. В скобках указана ошибочная категория после атаки.

Еще один метод, используемый для запутывания нейросетей - **вредоносная заплатка** — новая и популярная методика генерирования искажений. В прошлых двух методах искажающие данные добавлялись к исходным входным данным.

При использовании заплатки подбирают данные, которые подходят для всех изображений. Под «заплаткой» в данной задаче следует понимать следующее: изображение меньшего размера, которое накладывается поверх входных, чтобы обмануть классификатор.

Оптимизация работает в соответствии с этим уравнением:

$$\hat{p} = \arg\frac{max}{p} E_{x} \sim x, t \sim T, l \sim L[\log Pr(\hat{Y}| A(p, x, l, t))] \quad (3)$$

где:

$\hat{P}$ — подобранная заплатка,
$\hat{Y}$ — целевой класс,
A(p, x, l, t) - функция применения заплатки, которое является случайной и определяет куда и как накладывать заплату на входное изображение. p — заплатка. x — входное изображение. l — место, куда накладываются заплатки. t — преобразование заплатки (например, масштабирование и вращение).

В результате заплатка оказывалась эффективной на всех изображениях, используемого датасета – результатом была неправильная классификация. Это главное отличие данного метода от двух предыдущих. В них нейросеть обучалась на одном изображении, а этот метод позволяет выбрать заплатку, которая используется на большой выборке картинок. Заплатка может быть оптимизирована с помощью метода обратного распространения ошибки.

Рисунок 8 - Пример заплатки.

Далее будет рассмотрен **алгоритм выбора эффективных искажающих элементов.**

В данной работе предлагается прозрачный алгоритм выбора эффективных искажающих элементов, основанный на переборе возможных элементов, их положения, ориентации, сжатия, рисунок 9.

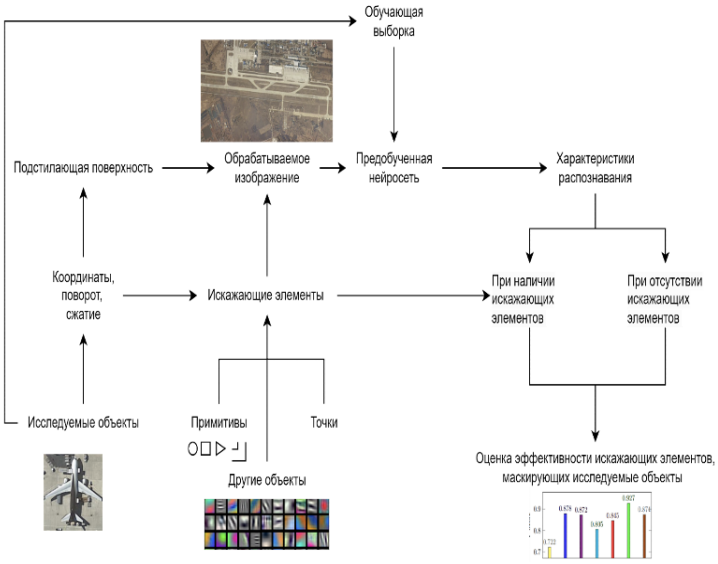


Рисунок 9 – Предлагаемый алгоритм определения эффективных искажающих элементов.

Алгоритм работает следующим образом:

Для выбранного исследуемого объекта (например, самолета на оптическом изображении) формируется обрабатываемое нейросетью изображение, которое представляет собой изображение подстилающей поверхности, на которую нанесены объекты с различной ориентацией, размером, поворотом и прочими оптическими преобразованиями.

Далее на обрабатываемое изображение в окрестности исследуемых объектов наносятся искажающие элементы. Такими элементами могут выступать случайные совокупности пикселей (как предельные частные случаи - один пиксель или шум на подстилающей поверхности и /или объекте). Также могут использоваться примитивы (кресты, прямоугольники, окружности и прочие элементы). Искажающими элементами могут быть элементы других объектов.

Как показали предварительные численные исследования, для ускорения расчетов в качестве «прочих элементов» целесообразно использовать элементы, на которых основано распознавание прочих объектов. При этом в качестве таких объектов целесообразно выбирать те, которые наиболее «подобны» исследуемому. Так, например, в работе [7] для самолета (airplane) наиболее похожим объектом является резервуар для хранения (storage tank). Выбор элементов, которые характеризуют резервуар для хранения, в окрестности самолета будет более эффективным, рисунок 4.

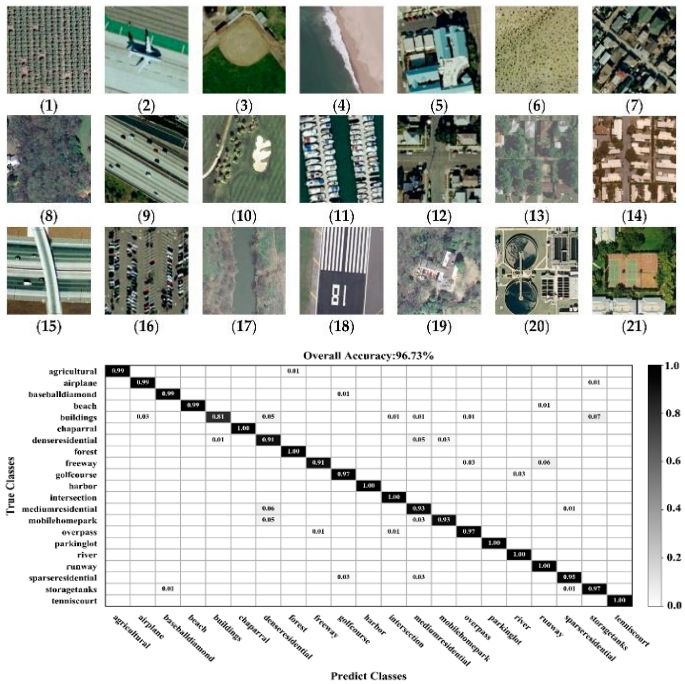


Рисунок 10 – Выбора прочих искажающих элементов для исследуемого объекта.

Результирующее изображение с исследуемыми объектами и искажающими элементами подается на обученную нейросеть. Лучше всего использовать сети, используемые при обработке данных дистанционного зондирования, [8]. Такие нейросети обучаются на наборах данных, например, на EuroSAT dataset, UCMerced-LandUse, NWPU-RESISC45 и другие. Набор данных EuroSAT основан на спутниковых снимках Sentinel-2, охватывающих 13 спектральных диапазонов состоящих из 10 классов с 27000 размеченных и географически привязанных выборок. Предлагаются два набора данных: - rgb: содержит только оптические полосы частот R, G, B, закодированные как изображение JPEG. - all: содержит все 13 полос исходного диапазона значений (float32).

Для исследуемых объектов определяются характеристики качества классификации при наличии и отсутствии конкретных искажающих элементов.

В качестве такой метрики могут использоваться Accuracy (точность) – доля правильно предсказанных классов, которая хорошо отражает качество работы обученных моделей для равнораспределенных классов.

$$Accuracy = (TN+TP) / (TN+FP+FN+TP) \quad (4)$$

Также может использоваться матрица ошибок (confusion matrix).

Для оценки нами эффективности рассчитывались метрики Precision, Recall, F1. Их расчет представлен в (5) – (7) соответственно:

$$Precision = TP / TP + FP, \quad (5)$$

$$Recall = TP / TP + FN, \quad (6)$$

$$F1 = 2 * ((Precision * Recall) / (Precision + Recall)) \quad (7)$$

где: TP – число истинно-положительных распознаваний; TN – число истинно-отрицательных распознаваний; FP – число ложно-положительных распознаваний; FN – число ложно-отрицательных распознаваний.

II. Результат использования методов запутывания нейросети:

Нейросеть обучалась на двух типах изображений – самолетах и машинах. Целью программы является максимально снизить правильную классификацию объектов при использовании методов запутывания нейросети.

Ниже представлены примеры применения и соответствующие результаты классификации изображения для каждого из описанных в работе методов запутывания нейросети:

```
1/1 [==============================] - 0s 57ms/step
Predicted Class (0 - Cars , 1 - Airplanes):  0.99521893
```

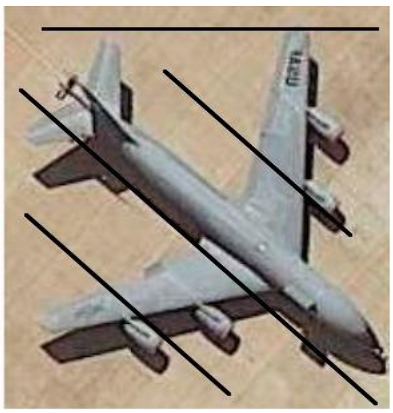
Рисунок 11 – Оригинальное изображение и результат классификации.

На рисунке 11 изображено оригинальное изображение объекта – самолет, нейросеть распознала самолет с вероятностью 99%.

```
1/1 [==============================] - 0s 64ms/step
Predicted Class (0 - Cars , 1 - Airplanes):  0.79968023
```

Рисунок 12 – Изображение с использованием пиксельной атаки (добавление 4 пикселей) и результат классификации.

На рисунке 12 применен метод пиксельной атаки, в нашем случае было использовано 4 пикселя. Нейросеть распознала самолет с вероятностью 79%.



```
1/1 [==============================] - 0s 59ms/step
Predicted Class (0 - Cars , 1 - Airplanes):  0.927808
```

Рисунок 13 – Изображение с использованием примитивов (прямых линий) и результат классификации.

На рисунке 13 применен метод с использованием примитивов, с целью запутывания классификатора, в нашем случае это прямые черные линии. Нейросеть распознала самолет с вероятностью 92%.

```
1/1 [==============================] - 0s 79ms/step
Predicted Class (0 - Cars , 1 - Airplanes):  0.06773631
```

Рисунок 14 – Изображение с использованием эффективных искажающих элементов, основанное на переборе возможных элементов (элементы машины), и результат классификации

На рисунке 14 использован метод эффективных искажающих элементов, основанный на переборе возможных элементов похожих (по мнению нейросети) на исследуемый объект. Нейросеть распознала самолет с вероятностью 6%, изображение было отнесено к классу «машины» с вероятностью 94%.

III. ЗАКЛЮЧЕНИЕ

Учитывая большое количество спутниковых систем дистанционного зондирования, для классификации объектов на изображениях используются нейронные сети. Качество классификации определённых объектов в таких сетях может быть значительно ухудшено при наличии искажающих элементов определённой формы, размера и ориентации. Линейный алгоритм поиска таких искажающих элементов путем полного перебора всех возможных комбинаций характеризуется излишне высокими требованиями к вычислительным возможностям аппаратуры. Следовательно, целесообразно ускорить вычисление за счет применения в качестве искажающих элементов - те, на которых основано распознавание объектов, которые «по мнению нйросети» наиболее «подобны» исследуемому (например, к самолету наиболее близки резервуары). Однако совершить качественный скачок в формировании эффективных искажающих признаков можно только при использовании генеративно-состязательной сети.

Список литературы

[1] Browne K., Swift B., Nurmikko-Fuller T. Camera adversaria //Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems. – 2020. – С. 1-9.

[2] Shan S. et al. Fawkes: Protecting privacy against unauthorized deep learning models //29th USENIX security symposium (USENIX Security 20). – 2020. – С. 1589-1604.

[3] Co K. T. et al. Procedural noise adversarial examples for black-box attacks on deep neural networks. – 2019.

[4] One pixel attack. Или как обмануть нейронную сеть // Habr URL: https://habr.com/en/articles/498114 (дата обращения: 10.10.2023).

[5] Su J., Vargas D. V., Sakurai K. One pixel attack for fooling deep neural networks //IEEE Transactions on Evolutionary Computation. – 2019. – Т. 23. – №. 5. – С. 828-841.

[6] Yu D. et al. An efficient and lightweight convolutional neural network for remote sensing image scene classification //Sensors. – 2020. – Т. 20. – №. 7. – С. 1999.

[7] Adegun A. A., Viriri S., Tapamo J. R. Review of deep learning methods for remote sensing satellite images classification: experimental survey and comparative analysis //Journal of Big Data. – 2023. – Т. 10. – №. 1. – С. 93.