ML for Security.
Лабораторная работа 3.
Методы цифровой форензики

Суть задачи лабораторной работы

Провести исследование по проверке подлинности изображений за счёт выявления следов JPEG-сжатия.

Используемый метод и программные средства

В лабораторной работе нужно использовать метод, описанный в работе https://arxiv.org/abs/2012.00468. Он предназначен для обнаружения двойного JPEG-сжатия и оценивания показателя качества QF_1 первого сжатия (второе и так известно, если изображение сохранено как jpeg, или может быть легко оценено по блочному спектру ДКП, если оно декодировано в tiff/PNG).

Для выполнения работы необходимо использовать файлы, предоставленные авторами статьи:

https://github.com/andreacos/BoostingCNN-Jpeg-Primary-Quantization-Matrix-Estimation

Чтобы не беспокоить уважаемых итальянских учёных, я рекомендую не форкать их репозиторий, а скачать его себе (Code -> Download ZIP). Ознакомьтесь с readme этого репозитория и в общих чертах посмотрите статью, чтобы понимать основную суть метода и проведённого авторами исследования. Мы будем работать с уже обученными нейросетевыми моделями.

По меньшей мере одна из используемых итальянцами библиотек не имеет сборки для Windows. Те, кто не привыкли работать с Linux, могут воспользоваться Google Colab. Для подготовки подходящего окружения необходимо выполнить команды:

```
!sudo apt-get install libmagickwand-dev
pip install -r "/content/requirements.txt"
pip install 'h5py==2.10.0' --force-reinstall
```

Порядок выполнения лабораторной работы

Основа нашей работы – это файлы usage.py (для вариантов A) и localization.py (для вариантов B). На основе одного из этих файлов необходимо создать один ноутбук и выполнить задание.

Обученные модели из репозитория возвращают целочисленные оценки 15 первых ДКП-коэффициентов в зигзагообразной развёртке. Они связаны с показателем качества QF_1, в варианте A нужно оценить его по этим оценкам. Сеть принимает на вход патчи размером 64x64.

Ваш pull-request в мой репозиторий с заданием должен содержать только ноутбук. Все изображения, создаваемые вами и используемые в экспериментах, должны лежать в папке /images/, которая может содержать ещё подпапки. Но /images/ вы не отправляете в pull-request. Также не нужно включать в него файлы из репозитория итальянцев. Таким образом, я буду при первичной проверке смотреть код и результаты по ноутбуку, а при показе вы должны быть готовы запустить его по моей просьбе.

Варианты задания

Есть два существенно отличающихся друг от друга варианта: обозначены цифрами A и B. В рамках каждого из них есть общая часть, определяющая общую схему исследования, и есть различия, определяемые подвариантами.

A. Анализ однородных изображений, не подвергавшихся локальным искажениям.
В варианте А работаем с изображениями размера 64x64.

A0 (базовое задание). Список пунктов:

- Написать функцию оценивания QF_1 по вектору из 15 оценок ДКП-коэффициентов (ближайший вариант по MSE или ближайший вариант по MSE со взвешиванием по матрице Q_{50})
- Программно сгенерировать 5 принципиально различных ситуаций: однократное сжатие, $QF_1 < QF_2$, $QF_1 \ll QF_2$, $QF_1 > QF_2$, $QF_1 \approx QF_2$.
- Сравнить ошибки оценивания QF_1 в различных ситуациях. Можно для каждой ситуации использовать одно изображение, но лучше 5-10, и усреднить результаты. Агрегировать результаты в какую-нибудь таблицу для удобства.
- При необходимости ответить на поверхностные вопросы, касающиеся самой задачи, авторского метода.

A1. A0 + Сравнить ошибку при оценивании QF_1 по одному патчу и при оценивании по $p = 10$ патчам, выбранным из одного изображения без пересечения.

A2. A0 + Построить график зависимости ошибки при оценивании QF_1 в зависимости от числа патчей p. Дойти до такого значения p, при котором график стабилизируется.

A3. A0 + Сравнить два варианта оценивания QF_1 (MSE и weighted MSE).

B. Исследование изображений, представляющих собой пример атаки image splicing (атака заключается в копировании фрагмента из одного изображения в другое).
В варианте В работаем с изображениями размера примерно 200x200 (при stride = 1). Больший размер будет слишком долго обрабатываться.

B0 (базовый вариант). Список заданий:

- Сгенерировать бинарную маску локальных изменений
- Сгенерировать составные изображения из по-разному сжатых JPEG-изображений (одно произвольное сочетание $QF_{1,1}$, $QF_{1.2}$ и QF_2).
- Автоматически сформировать маску изменений по одному произвольному ДКП-коэффициенту. При этом нельзя использовать информацию об истинных значениях $QF_{1,1}$, $QF_{1.2}$ и QF_2. Сделать это можно разными способами: качественными или не очень. Если нужно, могу дать совет, как это сделать получше.
- Выбрать меру оценивания качества построения маски изменений и произвести оценивание.

B1. B0 + использовать для построения маски несколько коэффициентов (можно хоть все 15). Сравнить с B0.

B2. B0 + реализовать функцию оценивания QF_1 и построить маску изменений по QF_1. Сравнить с B0.

B3. B0 + оценить влияние stride на качество оценивания маски.

B4. B0 + исследовать эмпирически, какая разница должна быть между $QF_{1,1}$, $QF_{1.2}$, чтобы маска определялась с приемлемым качеством

B5. B0 + встроить одновременно два разных ложных фрагмента с разными QF_1 и определить итоговую маску

Порядок выбора вариантов

- Лабораторную работу можно выполнять по одному или вдвоём.
- Если в общем списке (см мой файл) вы стоите на нечётной позиции, то делаете вариант A. Если на чётной позиции, то вариант B. Если делают задание двое, они могут «привязаться» к позиции в списке любого из пары.
- Студенты группы 6231 реализуют функцию на основе MSE, студенты группы 6233 – на основе Weighted MSE.
- Достаточно сделать A0 или B0. Но очень рекомендуется выбрать себе один из более продвинутых вариантов и записать его в таблицу. Должно быть выбрано не более 4-х реализаций каждого варианта (A1-A3, B1-B5). Реализаций A0 или B0 может быть сколько угодно.