МОСКОВСКИЙ ФИЗИКО-ТЕХНИЧЕСКИЙ ИНСТИТУТ
(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)

На правах рукописи

Гринчук Олег Валерьевич

МЕТОДЫ ОПРЕДЕЛЕНИЯ ПОДЛИННОСТИ ИЗОБРАЖЕНИЙ ЛИЦ

05.13.17 — Теоретические основы информатики

Диссертация на соискание ученой степени
кандидата технических наук

Научный руководитель:
д.ф.-м.н. В.И. Цурков

Москва — 2020

`

Содержание

Введение 5

Глава 1 12

Задача определения подлинности изображения лица 12

1.1. Постановка задачи определения живости 12

1.2. Условия применения алгоритмов определения живости 16

1.2.1. Сценарии применения 16

1.2.2. Кооперативность поведения пользователя 17

1.2.3. Модальности 18

1.3. Устойчивость алгоритмов определения живости 19

1.4. Классификация методов взлома 20

Физические артефакты 20

Доступность биометрического шаблона 21

Квалификация взломщика 21

Классификация атак 22

1.5. Процедура построения алгоритма определения живости 26

1.6. Необходимый размер обучающей выборки 33

1.7. Выбор представления данных 35

1.8. Существующие методы определения живости по изображению лица 38

1.8.1. Текстурно-частотный анализ 38

1.8.2. Анализ изменения фокуса камеры 39

1.8.3. Анализ движения глаз 39

1.8.4. Анализ оптического потока 40

1.8.5. Анализ моргания 40

1.8.6. Анализ кодирования компонент лиц 40

1.8.7. Анализ трехмерной структуры лица 41

1.8.8. Анализ признаков фона 41

Глава 2 42

Кооперативные методы определения живости 42

2.1. Атомарный метод определения живости 44

1. Атом по улыбке 47

2. Атом по открытому рту. 47

3. Атом по поднятым бровям. 47

`

4-7. Атом по повороту головы. 48
8. Атом по морганию. 49
2.2. Определение живости по оптическому потоку 52
2.3. Практическая реализация оценки живости по оптическому потоку 57
2.3.1. Сбор данных 59
2.3.2. Обучение модели 60
2.4. Заключение 64
Глава 3 64
Некооперативные методы определения живости для СКУД 64
3.1. Сбор обучающей выборки 67
3.2. Живость по одному изображению 69
3.2.1. Обучение модели 71
3.2.2. Эксперименты 71
3.3. Живость по границам изображения 73
3.3.1. Обучение модели 75
3.3.2. Эксперименты 76
3.4. Живость по динамике трека. 78
3.4.1. Описание алгоритма 80
3.4.2. Эксперименты 82
3.5. Заключение 83
Глава 4 86
Методы определения живости по мультимодальным данным 86
4.1. Живость по мультимодальным данным 86
4.2. Описание выборки 87
4.3. Предлагаемый метод 89
4.4. Архитектура модели 90
4.5. Эксперименты 91
4.6. Влияние мультимодальности 94
4.7. Заключение 95
Глава 5 96
Методы определения живости по видеопоследовательности 96
5.1. Описание выборки 98
5.2. Предлагаемый метод 98

`

5.3. Архитектура модели 100
5.4. Эксперименты 101
5.5. Заключение 103
Заключение 105
Список литературы 107

`

Введение

Актуальность темы. С развитием технологий глубокого обучения и увеличением вычислительных мощностей системы биометрической идентификации получили широкое практическое применение. Один из основных разделов данного направления – биометрия по лицу. Аутентификация по изображению лица постепенно вытесняет физические ключи обеспечения доступа на закрытые объекты, используется вместо пароля для удаленного подтверждения операций в финансовых учреждениях и заменяет пин-код или отпечатки пальца для авторизации в мобильных телефонах. По оценкам аналитического агентства Global Industry, объем рынка систем распознавания лиц составит $7.2 млрд. в 2020 году, а к 2027 году увеличится более чем в 3 раза – до $22.7 млрд [62]. С широким внедрением технологии распознавания лиц возникла задача защиты систем от попыток взлома и подлога чужого биометрического шаблона. Модели распознавания лиц проверяют только схожесть образца с биометрическим шаблоном и не могут оценить, зарегистрированный пользователь проходит контроль доступа или злоумышленник пытается обмануть систему. Самые распространенные попытки подлога осуществляются с помощью физических артефактов с изображением нужного человека, таких как бумажные распечатки фотографий, записи с экрана мобильного устройства или силиконовые маски, повторяющие трехмерную геометрию лица жертвы. Чтобы обеспечить надежную работу систем идентификации, требуются комплементарные подходы, проверяющие подлинность пользователя на изображении. Методы, решающие данную задачу, называются методами определения **живости.**

Современные системы биометрии по изображению лица уже превосходят способности человека в распознавании им других людей [18]. Значительная часть такого успеха обусловлена наличием больших размеченных выборок [19, 20], которые довольно легко собрать из Интернета, так как фотографии лиц являются наиболее рас-

`

пространенным видом изображений. В отличие от задачи распознавания лиц, изображения попыток подлога для обучения систем определения живости могут быть собраны только вручную, так как таких изображений в свободном доступе нет. Такие данные собираются исследовательскими группами в лабораториях с приглашенными участниками [21,22] и, следовательно, сильно ограничены в количестве и разнообразии, что нивелирует преимущества моделей глубокого обучения, которые хорошо работают на больших по объему выборках.

С другой стороны, для определения живости можно использовать не только обычные камеры, но и специальные сенсоры, предоставляющие дополнительные модальности для анализа. Эти модальности добавляют дополнительную информацию и могут повысить точность определения живости. Например, инфракрасная (ИК) камера нечувствительна к экранам электронных устройств и автоматически защищает от возможных подлогов такого рода. Камера глубины позволяет получить трехмерное изображение объекта, делая обнаружение любых плоских (отличающихся от формы лица) подделок проще.

Высокий спрос индустрии на решения определения живости стимулировал активное развитие исследований в данной области. В последние годы в открытом доступе появляется все больше соответственных обучающих выборок. Но, на текущий момент, существующие решения не удовлетворяют требованиям промышленных систем по скорости и точности работы, оставляя широкое поле для исследований. Кроме этого, алгоритмы, обученные на конкретных выборках, показывают слабые результаты на изображениях из других доменов, что делает их непригодными на практике.

Данная работа систематизирует текущее состояние задачи определения живости, а также описывает разработанные автором практически применимые методы.

`

Цели и задачи диссертационной работы.

В работе были поставлены следующие **цели**:

1. Исследовать кооперативные методы определения живости для стационарных и мобильных устройств с высоким уровнем защиты от всех видов взлома.
2. Разработать практически применимые методы определения живости для систем контроля и управления доступом (СКУД) с защитой от самых распространенных попыток взлома.
3. Предложить алгоритм определения живости для мультимодальных данных.
4. Предложить некооперативный метод определения живости для стационарных и мобильных устройств с защитой от неизвестных попыток взлома.

Для достижения поставленных целей были решены следующие **задачи**:

1. Разработка кооперативного метода определения живости для стационарных и мобильных устройств и анализ его работоспособности для различных типов атак.
2. Разработка удобного в применении кооперативного метода защиты от наиболее распространенных видов взлома.
3. Сбор обучающей и тестовой выборки, построение алгоритмов генерации новых данных для увеличения вариативности обучающей выборки.
4. Создание практически применимого алгоритма определения живости, эксплуатирующего особенности сценария.
5. Исследование полезных признаков модальностей глубины и ИК, разработка метода определения живости для мультимодальных данных.
6. Разработка устойчивого к неизвестным видам атак алгоритма определения живости по видеопоследовательности для стационарных и мобильных устройств.

`

Научная новизна.

1. Предложен новый кооперативный метод определения живости для мобильных и стационарных устройств, защищающий от известных видов атак.
2. Предложен новый точный и высокопроизводительный метод определения живости для систем контроля и управления доступом.
3. Предложена новая архитектура нейронной сети для решения задачи определения живости по мультимодальным данным, основанная на тесной связи промежуточных признаков разных модальностей.
4. Предложена концепция искусственных модальностей разработки методов защиты против неизвестных типов атак, на основе которой предложен быстрый и масштабируемый метод определения живости по видеопоследовательности.

Методы исследования. Для большей части предложенных алгоритмов широко применяется аппарат глубокого обучения нейронных сетей [76]. Для предобработки и подготовки данных к алгоритмам определения подлинности использовались модели детектирования лица и его ключевых точек [74, 75] и извлечения оптического потока [56]. Для сбора данных и демонстрации результатов применялись методы разработки клиенто-серверных мобильных приложений.

Основные положения, выносимые на защиту.

1. Кооперативный метод определения живости для работы в кооперативных сценариях для мобильных и стационарных устройств.
2. Метод определения живости по движению головы для мобильного сценария.
3. Метод определения живости для систем контроля и управления доступом, включающий в себя три независимых алгоритма: по одному изображению, по картам границ и по динамическим временным признакам лица.
4. Алгоритм определения живости по мультимодальным данным.

`

5. Алгоритм определения живости по видеопоследовательности для мобильных и стационарных устройств.

Теоретическая и практическая значимость. Предложенные в работе методы успешно используются в России и за рубежом различными компаниями. Все разрабатываемые методы ориентированы в первую очередь на практическое применение, где, помимо качества, учитывается скорость работы на широком диапазоне устройств. Часть предложенных алгоритмов показали лучший в мире результат на двух самых больших выборках по определению живости изображений лиц.

Степень достоверности и апробация работы. Достоверность результатов подтверждена экспериментальной проверкой, в том числе сторонними организациями; публикациями результатов исследования в рецензируемых научных изданиях и конференциях по машинному обучению. Результаты работы докладывались и обсуждались на следующих научных конференциях.

1. “Recognizing Multi-Modal Face Spoofing with Face Recognition Networks”, Международная конференция “Computer Vision and Pattern Recognition Workshops”. – Long Beach, CA. - 2019.
2. “Creating Artificial Modalities to Solve RGB Liveness”, Международная конференция “Computer Vision and Pattern Recognition Workshops”. – Virtual. – 2020.

Публикации по теме диссертации. Основные результаты по теме диссертации изложены в 6 печатных работах, которые изданы в журналах, рекомендованных ВАК.

1. *O. Grinchuk, V.I. Tsurkov.* Training a Multimodal Neural Network to Determine the Authenticity of Images. // Journal of Computer and Systems Sciences International. - 2020. – V. 59 (4). – P. 575-582.

`

2. *A. Parkin, O. Grinchuk*. Recognizing Multi-Modal Face Spoofing With Face Recognition Networks. // Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition Workshops. – 2019.
3. *O. Grinchuk, V.I. Tsurkov, L.P. Wang*. Neural Network Training System for Marker Encoding. // Journal of Computer and Systems Sciences International. – 2019. – V. 58 (3). – P.434-440.
4. *A. Gladkov, O. Grinchuk, Y. Pigareva, I. Mukhina, V. Kazantsev, A. Pimashkin.* Theta rhythm-like bidirectional cycling dynamics of living neuronal networks in vitro. // PloS one. – 2018. – V. 13 (2).
5. *O. Grinchuk, V.I. Tsurkov*. Cyclic Generative Neural Networks for Improved Face Recognition in Nonstardard Domains. // Journal of Computer and Systems Sciences International. – 2018. – V. 57 (4). – P.620-625.
6. *O. Grinchuk, V. Lebedev, V. Lempitsky*. Learnable Visual Markers. // Advances in Neural Information Processing Systems – 2016. – P. 4143-4151.

Личный вклад. Разработка алгоритмов и проведение экспериментов, описанных в главах 4 и 5, производилось совместно с Паркиным А.Н. [72], вклад автора был решающим. Вклад автора во все прочие положения, выносимые на защиту, также является решающим.

Обоснование специальности. Данная диссертация по своей тематике и направленности полученных результатов соответствует следующим пунктам паспорта специальности ВАК 05.13.17 “Теоретические основы информатики”:

5. Разработка и исследование моделей и алгоритмов анализа данных, обнаружения закономерностей в данных и их извлечениях, разработка и исследование методов и алгоритмов анализа текста, устной речи и изображений.

`

7. Разработка методов распознавания образов, фильтрации, распознавания и синтеза изображений, решающих правил. Моделирование формирования эмпирического знания.

Структура и объем работы. Диссертация состоит из оглавления, введения, пяти разделов, заключения и списка литературы. Основной текст занимает 112 страниц, включая 32 иллюстрации и 14 таблиц. Библиография включает 71 наименование.

Краткое содержание работы по главам. В первой главе вводятся основные понятия и определения, а также систематизируется решаемая задача по сценариям применения, поведению пользователя и типам фальсификаций. Рассматриваются существующие методы решения задачи определения подлинности изображений лиц.

Во второй главе предлагается кооперативный алгоритм определения живости и его атомарные составляющие.

В третьей главе рассматриваются некооперативные методы для систем управления и контроля доступом, предлагаются новые алгоритмы с учетом динамической структуры трека.

В четвертой главе предлагаются методы определения живости для мультимодальных данных.

В пятой главе предлагается некооперативный алгоритм определения живости по видеопоследовательности, противодействующий неизвестным типам атак.

`

Глава 1

Задача определения подлинности изображения лица

1.1. Постановка задачи определения живости

Задача определения подлинности изображения лица является составной частью процесса прохождения биометрической идентификации. Сначала сырые данные, захватываемые с камеры, передаются на модуль предобработки изображений, включающий в себя обнаружение лиц. Последовательность изображений, принадлежащих одному человеку затем передается в модуль распознавания лиц и модуль определения живости лица. Модуль распознавания лиц сравнивает поступившее изображение или набор изображений с биометрическим шаблоном, хранящемся в базе авторизированных для доступа пользователей. Модуль определения живости проверяет поступившие данные на наличие фальсификаций. Если оба модуля вынесли положительный вердикт, то человеку открывается доступ в систему, если хотя бы одна из частей системы ответила отрицательно, доступ в систему блокируется. (Рис 1.1.)

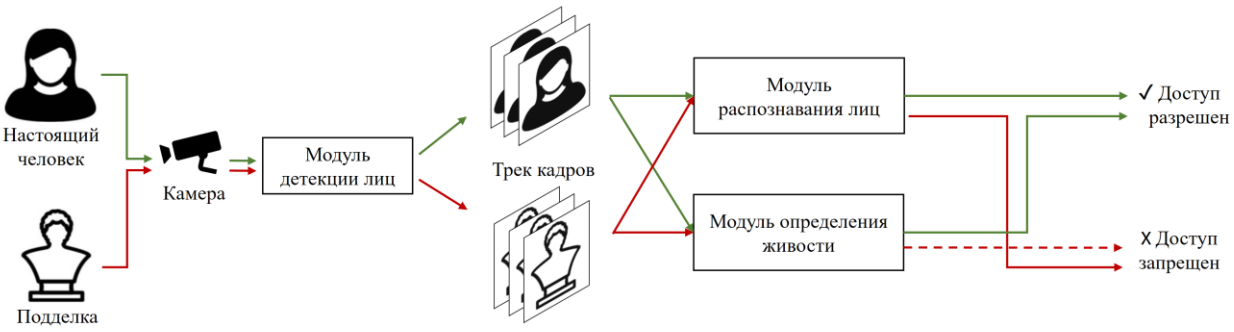

Рис 1.1. Блок-схема процесса биометрической аутентификации.

Будем рассматривать задачу определения подлинности лица по изображению или последовательности изображений изолированно от внешних факторов, предполагая,

`

что поток с записывающего устройства приходит сразу на проверку живости. Задача защиты данных в момент процесса передачи является прерогативой общей системы информационной безопасности и не относится к текущей проблематике.

Определение 1. *Изображением X* будем считать матрицу целых чисел размера $W \times H \times C$ в диапазоне от 0 до 255, где W – ширина, H – высота, C – количество каналов. Например, для обычных цветных изображений: $C = 3$, для черно-белых: $C = 1$.

Определение 2. Пусть $\mathbb{X}$ – множество изображений, на которых присутствует центрированное лицо человека. *Треком* длины T_N назовем последовательность изображений $T = \{X_j\},\ j = 1, \dots, T_N$ таких, что $X_j \in \mathbb{X}$.

Определение 3. Будем рассматривать *живость трека* T как бинарную переменную $l \in \mathbb{L}$, равную 1, если на записи живой человек, и 0, если перед камерой демонстрируется его "копия" – распечатанная фотография, экран электронного устройства с записью лица этого человека, трехмерная силиконовая маска и т.д.

Определение 4. *Моделью алгоритмов определения живости* назовем параметрическое семейство отображений $\{\mathbf{f}\ (T, \mathbf{w})\}$, где $\mathbf{f}\ (T, \mathbf{w})$ – в общем случае не дифференцируемая по параметрам $\mathbf{w} \in \mathbb{W}$ функция из множества треков в множество меток живости:

$$\mathbf{f}: \mathbb{X}^{\mathrm{N}} \times \mathbb{W} \to \mathbb{L}.$$

Пусть дана обучающая выборка

$$\boldsymbol{\mathcal{D}} = \{(T_i,\ l_i)\}, i = 1, \dots, m, \tag{1.1}$$

где $T_i = \{X_j\}, j = 1, \dots, N_{T_i},\ \ l_i \in \{0,1\}$ – множество треков и соответствующих им меток живости.

`

Алгоритм определения живости – отображение из множества треков в множество меток живости, чьи параметры оптимизированы по заданной обучающей выборке.

Метод определения подлинности изображений лиц – получение алгоритма определения живости для произвольной обучающей выборки.

Определение 5. *Оценкой живости* для трека T_i по некоторому алгоритму будем считать предсказание $s_i = \mathbf{f}(T_i, \mathbf{w}) \in [0,1]$, т.е. вероятность принадлежности трека классу 1.

Оценку живости всегда можно перевести в живость путем округления по некоторому выбранному на обучающей выборке порогу t

$$a_i = \begin{cases} 1, \text{если } s_i \geq t \\ 0, \text{если } s_i < t \end{cases} \tag{1.2}$$

Задача поиска оптимального алгоритма сводится к минимизации эмпирического риска по обучающей выборке $\boldsymbol{\mathcal{D}}$:

$$\frac{1}{m}\sum_1^m \mathcal{L}\left(\mathbf{f}(T_i,\ \mathbf{w}),\ l_i\right) \to \min, \tag{1.3}$$

где $\mathcal{L}$ – некоторая функция потерь. В большинстве случаев для поиска алгоритмов определения живости используется бинарная кросс-энтропия [54]

$$\mathcal{L}_i = -l_i \log s_i - (1 - l_i)\log(1 - s_i). \tag{1.4}$$

Минимизация эмпирического риска на обучающей выборке не гарантирует, что найденный алгоритм будет хорошо работать на других объектах. Хороший алгоритм должен обладать обобщающей способностью, которую можно измерить либо методом кросс-валидации на обучающей выборке, либо на зафиксированной тестовой выборке $\boldsymbol{\mathcal{D}_{test}}$.

`

Для разработки процедуры выбора оптимального алгоритма в случае задачи определения живости рассмотрим особенности получения обучающих и тестовых выборок.

Согласно вероятностной постановке задачи, рассматривается существование неизвестного распределения на множестве $\mathbb{X}^{\mathrm{N}} \times \mathbb{L}$, с плотностью $p(t, l)$, из которого случайно и независимо выбираются m объектов – полученная выборка называется простой. Для простых выборок хорошо работают методы кросс-валидации для получения оптимальных алгоритмов. Но в практических случаях, доступные данные для задачи определения живости распределены не равномерно. Рассмотрим такое распределение подробнее.

Множество возможных треков $\mathbb{X}^N$ прямо зависит от базового множества изображений лиц $\mathbb{X}$. Вероятностное распределение пар (T, l) представляется в виде

$$p(T, l) = p(T)p(T|l). \tag{1.5}$$

Априорное распределение треков изображений лиц $p(T)$ зависит от сценария использования, места съемки, освещения, оптических характеристик камеры и т.д. Условия могут быть внешними и внутренними.

Внутренними назовем условия, которые могут меняться в процессе эксплуатации. Примеры внутренних условий источника данных: модель видеокамеры, задний план изображения, освещенность. Ожидается, что качество алгоритма не будет зависеть от внутренних условий.

Внешними назовем условия, на которые можно опираться при разработке конкретного алгоритма, т.е. есть априорное знание что данные условия будут постоянны во время эксплуатации алгоритма. К внешним условиям относятся: сценарий применения S, кооперативность поведения пользователя I и модальность данных M.

`

1.2. Условия применения алгоритмов определения живости

1.2.1. Сценарии применения

Сценарий S – специфическая среда применения метода определения живости, отличающаяся типами записывающих устройств, их положением относительно человека, проходящего проверку и конечной целью проверки. Рассмотрим три основных типа сценариев:

$S_{\text{СКУД}}$: Система контроля и управления доступом (СКУД)– отвечает за контроль входа и выхода человека в помещение/зону с помощью системы биометрической идентификации по изображению лица. Представлена в виде турникетов с камерами, которые транслируют видеопоток на сервер, который находит на видеозаписи лица и сравнивает их с биометрическими шаблонами в базе разрешенных посетителей. В СКУД необходимо определение живости как метод защиты от несанкционированного доступа, чтобы злоумышленники не могли пройти внутрь по фотографии сотрудника. Отличительная особенность СКУД – человек виден на камере издалека, в течение следующих нескольких секунд он приближается, в конечном итоге проходя мимо неподвижной камеры.

$S_{\text{ПК}}$: Стационарные устройства – Персональный компьютер, банкомат, платежный терминал и другие неподвижные устройства, в которые требуется авторизация. Пользователь в начале и в процессе авторизации находится в непосредственной близости от камеры и смотрит в нее.

$S_{\text{моб}}$: Мобильные устройства – Мобильный телефон, планшет, и другие подвижные устройства, которые пользователь может держать в руке. Авторизация может потребоваться как для доступа в само устройство, так и для подтверждения важных операций внутри приложений, например, банковских.

`

Приведенные выше сценарии отличаются подвижностью камеры, диапазоном возможных размеров лиц и углов поворота головы, а также потенциальными уязвимостями к различным видам фальсификаций, что делает рациональным разработку специфических методов определения живости под каждый конкретный сценарий.

1.2.2. Кооперативность поведения пользователя

Ввиду того, что размеры выборок для обучения задач определения подлинности на порядки меньше необходимых для эффективного решения общей задачи, в практическом применении до сих пор используются алгоритмы, интерактивно итерирующие с пользователем, что существенно увеличивает точность решения. Система может попросить человека совершить некоторое действие, которое поможет в определении живости, например, улыбнуться. Дополнительный алгоритм проверки наличия улыбки, запущенный на треке, по логическому “или” сможет отклонить любые статические фальсификационные артефакты.

Методы, интерактивно взаимодействующие с пользователями, назовем *кооперативными*; методы, не требующие от пользователя никаких действий – *некооперативными*. Будем считать характеристикой метода определения живости набор параметров I, которые описывают запрашиваемые алгоритмами действия. *Степенью кооперативности* будем считать удобство пользователя и время выполнения действия. Для некооперативных методов степень кооперативности равна 0.

Несмотря на хороший уровень защиты, основным недостатком кооперативных методов является повышенное время работы и неудобство любых интерактивных действий для пользователя, поэтому сейчас основные исследования ведутся в области разработки некооперативных алгоритмов.

`

1.2.3. Модальности

Помимо стандартных камер, снимающих видео в видимом диапазоне в формате RGB (red, green, blue – аддитивная цветовая модель, описывающая способ кодирования цвета), существуют специальные устройства, записывающие поток в других видах и диапазонах, таких как: глубина, ближний инфракрасный диапазон, тепловой диапазон. Такие камеры стоят дороже и подходят не для всех сценариев, но при этом ввиду особых свойств получаемых изображений позволяют извлечь больше полезной информации для решения задачи определения живости. Рассмотрим подробнее основные виды модальностей, встречающиеся в промышленных системах:

M_{RGB} – модальность обычных изображений в формате RGB. Самая распространенная модальность практически в любом сценарии ввиду низкой стоимости устройств и уже существующей инфраструктуры соответствующих видеокамер.

$M_{\text{глубина}}$ – модальность изображений с камер глубины, показывающих объемную картину. Карта глубины – одноканальное изображение со значениями расстояния до соответствующих пикселей в миллиметрах. Алгоритмы определения живости, обученные на данных такого типа обычно устойчивы к плоским артефактам, таким как бумажные листки или экраны устройств, так как те легко различимы на трехмерном изображении. Недостаток $M_{\text{глубина}}$ – высокая стоимость камеры, работа в определенном диапазоне расстояния и плохое качество изображения при солнечном свете. Но в сценариях, позволяющих использовать камеры глубины, методы определения живости на их основе показывают самую высокую надежность.

$M_{\text{ИК}}$ – модальность изображений в инфракрасном (ИК) спектре. Камеры, снимающие в таком диапазоне обычно доступнее камер глубины, а также могут работать в ночном режиме, что послужило их широкому распространению. Кроме этого, ИК-диапазон обладает полезным свойством для определения подлинности лица – он не отображает содержание экранов электронных устройств, показанных на камеру, автомати-

`

чески отсекая большое семейство потенциальных фальсификаций. Еще одно преимущество ИК-изображений – глаза реальных людей на них выглядят специфически, поглощая инфракрасное излучение, что также является сильным признаком для решения целевой задачи.

1.3. Устойчивость алгоритмов определения живости

Распределение изображений $p(T)$ можно записать как

$$p(T) = p(T \mid \boldsymbol{\theta}, S, I, M), \tag{1.6}$$

где $\boldsymbol{\theta}$ – набор внутренних условий. При зафиксированных внешних условиях (1.6) можно рассматривать как совокупность кластеров внутренних условий

$$p(T) = p(T \mid \boldsymbol{\theta}, S, I, M) = p(T|\boldsymbol{\theta_1}, S, I, M) + p(T|\boldsymbol{\theta_2}, S, I, M). \tag{1.7}$$

Будем считать, что условия $\boldsymbol{\theta_1}$ генерируют обучающую выборку, а $\boldsymbol{\theta_2}$ – неизвестную контрольную выборку, т.е.

$$\boldsymbol{\mathcal{D}} \sim p(T|\boldsymbol{\theta_1}, S, I, M)p(T|l), \tag{1.8}$$

$$\boldsymbol{\mathcal{D}_{test}} \sim p(T|\boldsymbol{\theta_2}, S, I, M)p(T|l).$$

При этом обе выборки по отдельности являются простыми. В таком случае задача определения живости сводится к обучению алгоритма на известной выборке и контроле качества на тестовой выборке. При обучении, выборка разбивается на обучающую и валидационную.

Определение 6. Пусть значение функции потерь на обучающей выборке равно $\mathcal{L}_{train}$, на валидационной – $\mathcal{L}_{\mathrm{val}}$. *Устойчивостью обучения* назовем

$$\Delta = \frac{\mathcal{L}_{val} - \mathcal{L}_{train}}{\mathcal{L}_{val}} \tag{1.9}$$

`

Если $\Delta \to 1$, то алгоритм сильно переобучается и плохо работает на валидации, хотя та выбрана из того же распределения, что и обучающая выборка. В таком случае, ожидать хорошей работы на контрольной выборке маловероятно. Если $\Delta \sim 0$, то можно смотреть на качество классификации на контрольной выборке.

Внутренние и внешние условия определяют распределение $p(T)$. Но, данные для задачи определения живости также зависят от класса объекта, т.е. от $p(T|l)$. Рассмотрим случай $l = 0$, т.е. когда перед камерой демонстрируются различные фальсификации.

1.4. Классификация методов взлома

Атакой PA (Presentation Attack) на систему биометрической идентификации назовем демонстрацию перед камерой материального артефакта, содержащего образ человека, биометрический шаблон которого находится в базе разрешенных пользователей, с целью осуществить несанкционированный доступ в систему. Само распознавание лиц не способно отличать атаки от реальных попыток, для этого используются методы определения живости.

Атаки отличаются видами артефактов, степенью сложности получения изображения пользователя, а также уровнем возможностей взломщика. Рассмотрим классификацию атак по этим типам.

Физические артефакты

1. Бумажные. Изображение лица пользователя распечатывается на обычной или фотобумаге. Возможно вырезание по контуру лица, а также прорези для глаз/рта.
2. Электронные. Изображение лица пользователя выводится на экран телефона/планшета. Возможна демонстрация не только статической картинки, но и видеозаписи.

`

3. Трехмерные. На основе изображения лица делается силиконовая маска (другие материалы также возможны), которую злоумышленник надевает на себя. Возможны маски с подвижными глазами, ртом.

Доступность биометрического шаблона

1. Уровень А. Статическое изображение лица пользователя обычного качества. Часто можно получить из социальных сетей или скрытой фотографией.
2. Уровень В. Видеозапись лица пользователя, изображение лица высокого качества. Такие шаблоны получить сложнее, в открытых источниках их почти нет.
3. Уровень С. Видеозапись лица пользователя с конкретными действиями (улыбка, поворот головы налево, и т.д.), изображение лица в другой модальности (ИК). Такие данные получить без целенаправленной слежки за пользователем невозможно.

Квалификация взломщика

1. Уровень А. Обычный человек, не знакомый с методами защиты биометрических систем.
2. Уровень В. Человек средней подготовленности, имеющий представление об основных методах определения живости, кооперативности и скрытых проверках.
3. Уровень С. Эксперт или группа экспертов в области информационной безопасности и методов определения живости.

В зависимости от рода используемых артефактов, доступности биометрического шаблона и экспертизы злоумышленника рассматривается классификация атак по степени сложности [63].

`

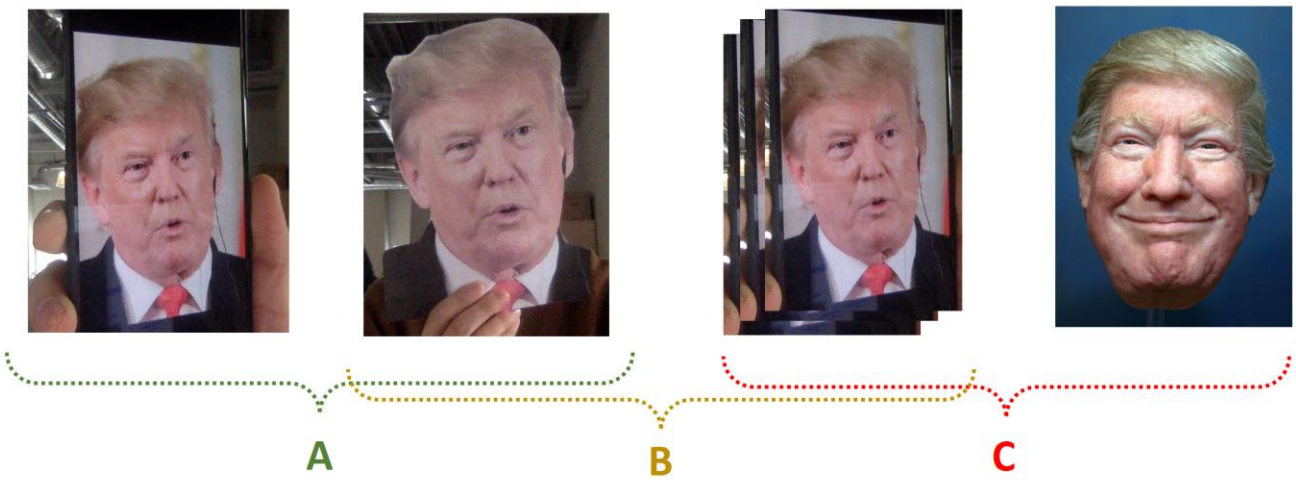

Рис 1.2. Примеры атак разных уровней.

Классификация атак

1. Уровень А. Распечатанная или показанная на экране фотография с небольшой доработкой. Самый распространенный вид атак, т.к. не требует специализированного оборудования, высокой экспертизы и сложностей в получении биометрического шаблона. Примеры атак уровня А: А4 распечатка лица; вырезанная по форме лица фотография в натуральную величину; фотография лица на экране мобильного либо стационарного устройства.
   Минимальная экспертиза взломщика: уровень А.
   Доступность биометрического шаблона: уровень А.

2. Уровень В. Видеозапись пользователя или бумажная маска на основе фото высокого разрешения. Без специальной подготовки такой уровень атак неосуществим, т.к. требуется повышенная экспертиза и есть сложности в получении шаблона. Примеры атак уровня В: видеозапись пользователя, демонстрируемая на экране мобильного/стационарного устройства; бумажная маска с грубой 3D структурой, вырезанная из бумаги; последовательность распечатанных фотографий с определенным выражением лица (фото без улыбки, фото с улыбкой).

`

Минимальная экспертиза взломщика: уровень B.

Доступность биометрического шаблона: уровень B.

3. Уровень C. Трехмерные маски, сложные видеозаписи, данные из других модальностей. Основное отличие от уровня B – во времени на подготовку и стоимости артефакта взлома. Кроме этого требуется высокая экспертиза взломщика в создании реалистичных артефактов и трехмерных структур. Примеры атак уровня C: силиконовая/керамическая маска, копирующая лицо пользователя; Распечатанное изображение в ИК диапазоне; видео, повторяющее процедуру конкретного кооперативного метода liveness.

   Минимальная экспертиза взломщика: уровень C

   Доступность биометрического шаблона: уровень B

Таблица 1.1. Описание наиболее распространенных видов атак.

Код	Описание	Уровень
P1	Распечатанное изображение (например, на A4) без модификаций. Края бумаги видны в кадре.	A
P2	Распечатанное изображение (например, на A4) без модификаций. Края бумаги не видны в кадре.	A
P3	Распечатанное изображение лица, вырезанное по контуру. Без изменений на изображении лица, как в P4.	A
P4	Распечатанное изображение лица, вырезанное по контуру + вырезанные глаза или рот для выполнения активных выражений, как моргнуть или улыбнуться.	A

`

P5	Вырезанный бумажный участок лица, для накладывания на реальное лицо, например, область вокруг глаз + щеки. Перед тестированием подобной атаки против liveness необходимо убедиться в том, что этой информации достаточно для распознавания лиц с ожидаемым ответом.	B
P ext	Несколько артефактов из P* одного человека – нейтральное лицо, улыбающиеся лицо и т.д.	A
P ir	Артефакт из P*, распечатанный в ИК-диапазоне	C
D1	Статическое изображение лица (как из социальных сетей), выведенное на экран телефона. Границы телефона находятся в кадре.	A
D2	Статическое изображение лица, выведенное на экран планшета / ноутбука / монитора. Границы экрана не видны в кадре.	A
D1/D2 ext	Несколько кадров из D1/D2 с фиксированными кооперативными выражениями лица	A
D3	Случайное видео с атакуемым (что можно добыть в соц. сетях) с движениями губ, либо с морганием, либо с сменой выражения лица.	B
D4	Видео с атакуемым, повторяющее близкие движения к тому, что требуется в системе.	B/C
D5	Видеозвонок с попыткой удаленного доступа. Сговор взломщика и легитимного пользователя.	C
M1	Объемная бумажная маска атакуемого.	B
M2	Точная силиконовая маска атакуемого без возможности выполнения активного действия (вырезанные глаза, рот и так далее).	C
M3	Точная силиконовая маска атакуемого с возможностью выполнения активного действия.	C

`

M4	Точная керамическая маска без возможности выполнения активного действия.	C

Большинство видов атак относятся к категориям A и B (табл. 1.1.). Уровень C требует высокой экспертизы и стоимости оборудования, что часто делает невыгодным попытку взлома такого уровня сложности. Поэтому на практике, заказчики ориентируются на алгоритмы, защищающие от первых двух категорий, считая это достаточным. Чем выше уровень атак, против которых нужна защита, тем больше такие методы неудобны для конечного пользователя и дороже для заказчика.

С учетом различных видов атак, распределение данных для задачи определения живости можно выразить как

$$p(T,l) = p(T)p(T|l) = p(T)p(T|l=1) + p(T)p(T|l=0)$$
$$p(T|l=0) = p(T|l=0, PA_1) + p(T|l=0, PA_2) + \cdots$$

или, упрощая и перенося класс $l = 1$ в множество атак

$$p(T,l) = p(T)p(T|\{\boldsymbol{PA}\}) \tag{1.10}$$

Тогда (1.8) перепишется как

$$\boldsymbol{\mathcal{D}} \sim p(T|\boldsymbol{\theta_1}, S, I, M)p(T|\{\boldsymbol{PA}\}_\mathbf{1}), \tag{1.11}$$

$$\boldsymbol{\mathcal{D}}_\mathbf{test} \sim p(T|\boldsymbol{\theta_2}, S, I, M)p(T|\{\boldsymbol{PA}\}_\mathbf{2}).$$

В общем случае набор атак, представленных в обучающей выборке, не эквивалентен набору, который может встретиться в тесте. Рассмотрим процесс построения алгоритма определения живости при заданных условиях и видах атак.

`

1.5. Процедура построения алгоритма определения живости

Пусть дана обучающая выборка $\boldsymbol{\mathcal{D}}$ в условиях *S, I, M*. Требуется построить алгоритм определения живости по заданной выборке. Предлагается процедура получения оптимального алгоритма, состоящая из четырех этапов. Вначале, если возможно, генерируются дополнительные данные, которые разнообразят обучающую выборку. После чего выборка фиксируется и разбивается на подвыборки для кросс-валидации. Далее, исходя из бюджета на время работы итогового алгоритма и максимальный размер памяти выбирается семейство моделей и обучаются соответствующие алгоритмы. Гиперпараметры модели, параметры обучения и функции аугментации данных выбираются из хорошо зарекомендовавшего для данных моделей множества параметров исходя из результатов кросс-валидации. Наконец, обученный алгоритм тестируется на контрольной выборке.

1.5.1. Генерация синтетических данных

При фиксированной модели, чем больше разнообразных данных доступно для обучения, тем выше точность итогового алгоритма [64]. В зависимости от внешних условий *S, I, M,* можно предложить метод генерации данных для реальных и поддельных изображений лиц.

Пример. Для кооперативного алгоритма определения живости по улыбке можно составить новые примеры для класса $l = 0$ путем дублирования одного кадра реального человека. Получится новое семейство поддельных треков, которое разнообразит обучающую выборку.

1.5.2. Разбиение обучающей выборки

В случае простой выборки одним из оптимальных способов избежать переобучения является кросс-валидация [65]. Рассмотрим особенности данного метода для данных определения живости. Предполагается, что заданная обучающая выборка $\boldsymbol{\mathcal{D}}$ –

`

простая, то есть примеры выбраны независимо и случайно при внутренних условиях $\boldsymbol{\theta_1}$ и наборе атак $\{\boldsymbol{PA}\}_\mathbf{1}$.

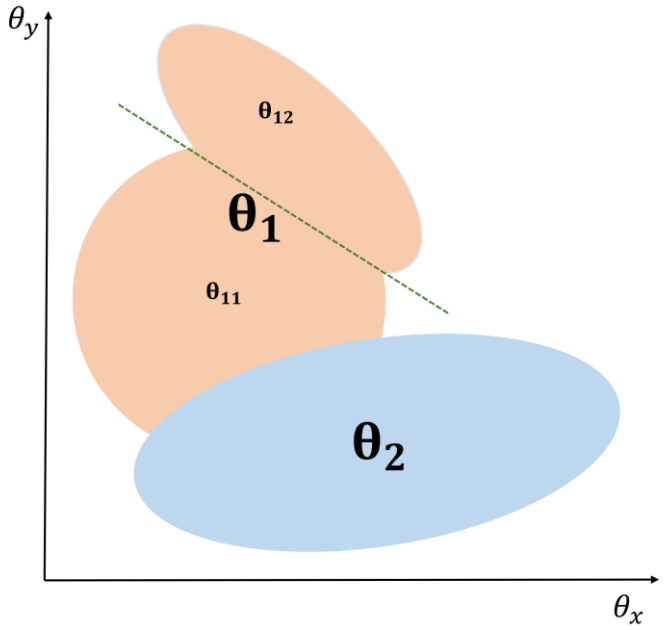

Рис 1.3. Разбиение обучающей выборки по подгруппам внутренних условий.

Однако в действительности выборка неоднородна – условия и атаки разделяются на подгруппы. При наивной генерации подвыборок для кросс-валидации примеры из одинаковых подгрупп окажутся и в обучении, и в валидации, делая валидацию неинформативной. Поэтому предлагается разделять выборку на обучающую и валидационную части *по подгруппам* (рис. 1.3.). Например, целесообразным разбиение изображений лиц на группы по конкретным людям – **id** (от англ. identity), чтобы избежать переобучения на черты лица. Для простоты обозначений, будем рассматривать id как одну из размерностей внутренних условий.

Разбиение заданной выборки на обучающую и валидационную по подгруппам эмулирует существующее разделение данных на заданную и контрольную выборки. В предлагаемом методе (1.11) будет иметь вид

$$\boldsymbol{\mathcal{D}} = \boldsymbol{\mathcal{D}}_{\mathbf{train}} \cup \boldsymbol{\mathcal{D}}_{\mathbf{val}} \sim \tag{1.12}$$

`

$$p(T|\boldsymbol{\theta_{11}}, S, I, M)p(T|\{\boldsymbol{PA}\}_{\boldsymbol{11}}) + p(x|\boldsymbol{\theta_{12}}, S, I, M)p(T|\{\boldsymbol{PA}\}_{\boldsymbol{12}})$$

Количество частей задается в зависимости от выбранного метода кросс-валидации. Можно разбивать стандартно на 5, либо, в случае ограниченности вычислительных ресурсов, можно разбить на две части, выделив на валидационную часть около 15-20% всех доступных данных.

Кросс-валидация по условиям, атакам и id сделает алгоритм более устойчивым к неизвестным данным, которые могут присутствовать в контрольной выборке, но уменьшит доступное для обучения число изображений, так как предлагаемые параметры разбиения не всегда могут линейно разделить выборку.

1.5.3. Выбор модели и оптимизация гиперпараметров

В данной работе в качестве семейств моделей $\mathbf{F} = \{\mathbf{f}\ (T, \mathbf{w})\}$ преимущественно рассматриваются сверточные нейронные сети, хорошо работающие для задач компьютерного зрения.

Нейронная сеть – дифференцируемая по параметрам $\mathbf{w} \in \mathbb{W}$ функция $\mathbf{f}\ (X, \mathbf{w})$, такая что

$$\mathbf{f}(X_i, \mathbf{w}) = s_i,$$

где s_i – оценка живости. Последним слоем нейронной сети обычно является сигмоидная функция

$$\sigma(x) = \frac{1}{1+e^{-x}} \tag{1.12}$$

переводящая s_i в диапазон $[0, 1]$.

Для обучения алгоритмов оценки живости используются хорошо зарекомендовавшие себя архитектуры : mobilenet [49], resnet [37, 77], efficientnet [66] и их модификации. Кроме этого, предлагается архитектура SimpleNet, построенная по общему принципу формирования сверточных сетей.

`

Сначала определяются модели $\hat{\mathbf{F}}$, удовлетворяющие заданному бюджету времени работы T. Пусть $\tau_{\mathbf{f}}$ – время работы прямого прохода нейронной сети для одного трека $\mathbf{f}\ (T_i, \mathbf{w})$. Тогда множество рассматриваемых моделей:

$$\hat{\mathbf{F}} = \{\mathbf{f}(T, \mathbf{w})\ \forall\ \mathbf{f} \in \mathbf{F}\colon \tau_{\mathbf{f}} \leq \mathrm{T}\} \tag{1.13}$$

Рассмотрим процесс обучения нейронной сети для треков и соответствующих меток живости в случае, когда длина трека равна 1. Обучающая выборка будет иметь вид

$$\boldsymbol{\mathcal{D}}_{\mathbf{train}} = \{X_i, l_i\}, i = 1, \dots, k,$$

а выбранная валидационная часть:

$$\boldsymbol{\mathcal{D}}_{\mathbf{val}} = \{X_i, l_i\}, j = k + 1, \dots, m.$$

Каждая из рассматриваемых архитектур обучается с несколькими базовыми наборами гиперпараметров $\boldsymbol{\gamma}$, после чего выбирается лучшая по кросс-валидации архитектура:

$$\mathbf{f}^* = \arg\min_{\mathbf{f}\in\hat{\boldsymbol{F}},\boldsymbol{\gamma}} \mathcal{L}_{\mathrm{val}}(l\ , \mathbf{f}(X, \mathbf{w} \mid \boldsymbol{\gamma}\ )) \tag{1.14}$$

К гиперпараметрам обучения $\boldsymbol{\gamma}$ будем относить:

- параметры оптимизатора – начальный коэффициент обучения и функция его изменения относительно номера эпохи.
- структуру нейронной сети – количество сверток в каналах, количество слоев, функции активации.
- набор аугментаций исходных данных и степень их случайности. Обычно используются повороты, сдвиги, размытие, цветовая коррекция, зеркальные отображения.

`

Параметры нейронной сети обучаются методом обратного распространения ошибки [67] с функцией потерь – бинарная кросс-энтропия $\mathcal{L}$ (1.4). На рис 1.2. показана схема обучения нейронной сети. На каждом шаге обучения данные разбиваются на батчи (несколько изображений, объединенных в один четырехмерный тензор), подаются на вход сети, считается функция потерь, градиент и обновляются веса модели. В конце обучения сохраняется слепок параметров $\mathbf{w}$, показавший лучший результат на валидационной выборке.

Вход: $\boldsymbol{\gamma}$: размер батча b, количество эпох $nepoch$; оптимизатор Opt и его гиперпара метры; функции предобработки изображений *Augment.*

Выход: обученные параметры $\mathbf{w}$

1. **для** $e = 1, \dots, epoch$
2. перемешать индексы $1, \dots, m \to p_1, \dots, p_m$
3. **для** $i = 1, \dots, [\frac{m}{b}]$
4. $\boldsymbol{X} = [\, X_{p_{ib}}, X_{p_{ib+1}}, \dots, X_{p_{ib+b-1}}]$, $\boldsymbol{l} = [\, l_{p_{ib}}, l_{p_{ib+1}}, \dots, l_{p_{ib+b-1}}]$,
5. $\boldsymbol{X} = Augment(\boldsymbol{X})$
6. $\boldsymbol{s} = f(\boldsymbol{X}, \mathbf{w}),\ loss = \mathcal{L}(\boldsymbol{s}, \boldsymbol{l})$
7. $\nabla\mathbf{w} = \frac{\partial loss}{\partial \mathbf{w}}$
8. $\mathbf{w} = Opt\ (\mathbf{w}, \nabla\mathbf{w})$
9. посчитать *loss* для $\boldsymbol{\mathcal{D}}_{\mathbf{val}}$, сохранить $\mathbf{w}$.

Рис 1.4. Псевдокод обучения нейронной сети.

`

Поочередно меняя гиперпараметры от допустимого минимального до максимального, обучается 20-30 моделей $\mathbf{f}^*$ и потом из них выбирается лучшая по точности на валидационной выборке:

$$\mathbf{w}^* = \arg\min_{\mathbf{w},\boldsymbol{\gamma}} \mathcal{L}_{val}(l\,, \mathbf{f}^*(X, \mathbf{w} \mid \boldsymbol{\gamma})) \tag{1.15}$$

Полученный таким образом алгоритм $\mathbf{f}^*(X, \mathbf{w}^*)$ назовем *оптимальным*.

1.5.4. Проверка результатов на тестовой выборке

Тестовая выборка никак не участвует в обучении и выборе оптимальной модели, но результаты на ней позволяют оценить устойчивость полученного алгоритма при других внешних условиях. Помимо значения функции потерь, считаются и другие показатели.

Основными мерами качества для задачи определения подлинности являются показатели TPR (True Positive Rate) в определенных точках FPR (False Positive Rate), а также ACER (Average Classification Error Rate). Рассмотрим эти метрики подробнее.

Пусть задана тестовая выборка

$$\boldsymbol{\mathcal{D}}_{\mathbf{test}} = \{(T_i, l_i)\}, i = 1, \ldots, m,$$

и модель определения живости $\mathbf{f}(T_i, \mathbf{w}^*)$, возвращающая в качестве результата вероятность $s_i \in [0,1]$ – уверенность модели в том, что трек T_i принадлежит живому человеку. Пусть зафиксирован некоторый порог $t \in [0,1]$, по которому предсказания модели переводятся в бинарные значения:

$$a_i = \begin{cases} 1, \text{если } p_i \geq t \\ 0, \text{если } p_i < t \end{cases} \tag{1.16}$$

Тогда

$$\begin{aligned} TP &= \textstyle\sum_1^m [a_i = l_i = 1] \\ TN &= \textstyle\sum_1^m [a_i = l_i = 0] \\ FP &= \textstyle\sum_1^m [a_i \neq l_i = 0] \\ FN &= \textstyle\sum_1^m [a_i \neq l_i = 1] \end{aligned} \tag{1.17}$$

`

где TP, TN, FP, FN – истинно-положительное, истинно-отрицательное, ложно-положительное и ложно-отрицательное число предсказаний в выборке относительно модели $\mathbf{f}$.

FPR – False Positive Rate – доля ложно-положительных предсказаний относительно общего числа предсказаний отрицательного класса. TPR – True Positive Rate – доля истинно-положительных предсказаний относительно общего числа предсказаний положительного класса, т.е.

$$FPR = \frac{FP}{FP+TN}$$
$$TPR = \frac{TP}{TP+FN} \quad (1.18)$$

Так как меры выше зависят от порога t, для оценки качества алгоритма будем смотреть на несколько фиксированных точек FPR, например 0.001, 0.01 и 0.1. На практике, если, зафиксирован порог в точке $FPR = 0.01$, при котором $TPR = 0.97$, то, из всех попыток взлома системы, 1% окажутся успешными, при этом из всех попыток пройти реальному пользователю откажут в 3% случаев.

Кроме FPR и TPR, часто рассматривается ACER – Average Classification Error Rate, состоящий из APCER – Attack Presentation Classification Error Rate и BPCER – Bonafide Presentation Classification Error Rate. Они почти ничем не отличаются от предыдущих мер, т.к. APCER = FPR и BPCER = 1-TPR и

$$\text{ACER} = \frac{\text{APCER}+\text{BPCER}}{2} \quad (1.19)$$

но более удобны для восприятия в контексте задачи определения живости, поэтому получили более широкое применение чем традиционные меры.

Значения функции потерь $\mathcal{L}$, TPR в точке FPR или ACER на контрольной выборке $\boldsymbol{\mathcal{D}}_{\mathbf{test}}$ будем называть *мерой качества* Q алгоритма определения живости $\mathbf{f}$.

`

1.6. Необходимый размер обучающей выборки

Качество алгоритмов, основанных на нейронных сетях, напрямую зависит от количества доступных данных. Зависимость качества алгоритма $\mathbf{f}$ от размера обучающей выборки подчиняется экспоненциальному закону [68, 69, 70]:

$$Q(\mathbf{f}, \boldsymbol{\mathcal{D}}_{\mathbf{test}}) = a|\boldsymbol{\mathcal{D}}|^{-b} + c, \tag{1.20}$$

где Q – некоторая мера качества, $|\boldsymbol{\mathcal{D}}|$ – размер обучающей выборки. Не теряя общности, будем считать, что минимальное значение меры качества равно 0, и оно соответствует идеальной классификации.

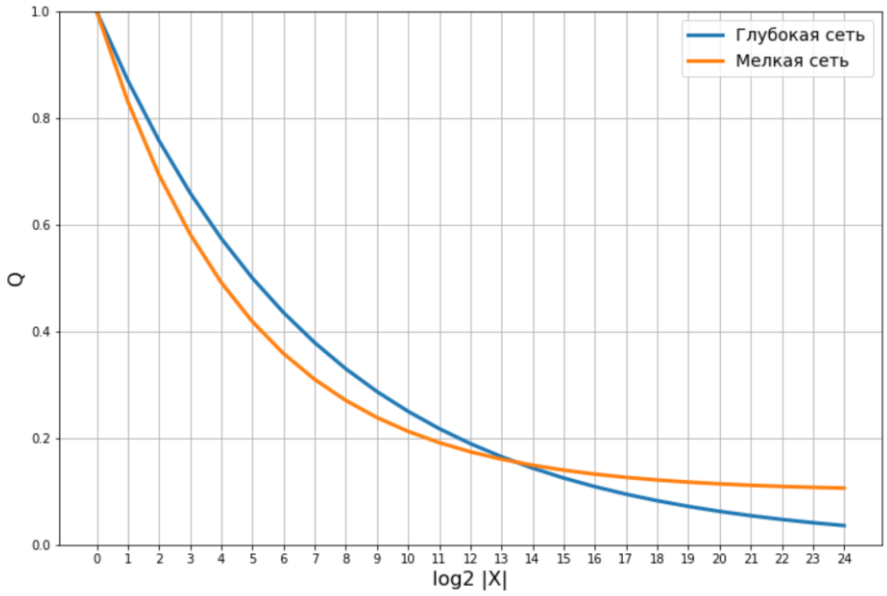

Рис 1.5. Пример зависимости меры качества от размера обучающей выборки для разных нейронных сетей.

`

На рис. 1.5. показан пример поведения меры качества в зависимости от размера обучающей выборки при разных значениях параметров кривой. Синяя кривая – пример более глубокой нейронной сети. При недостаточном количестве данных ее качество хуже, чем у легкой сети и параметр устойчивости обучения Δ выше, чем у менее глубокой сети.

Но при этом при наличии большого числа примеров для обучения итоговое качество глубокой модели становится лучше. Однако, для задачи определения живости большие выборки не всегда доступны ввиду сложности сбора данных, поэтому в ряде случаев предпочтение отдается более легким моделям, которые к тому же удовлетворяют временным ограничениям для практического применения.

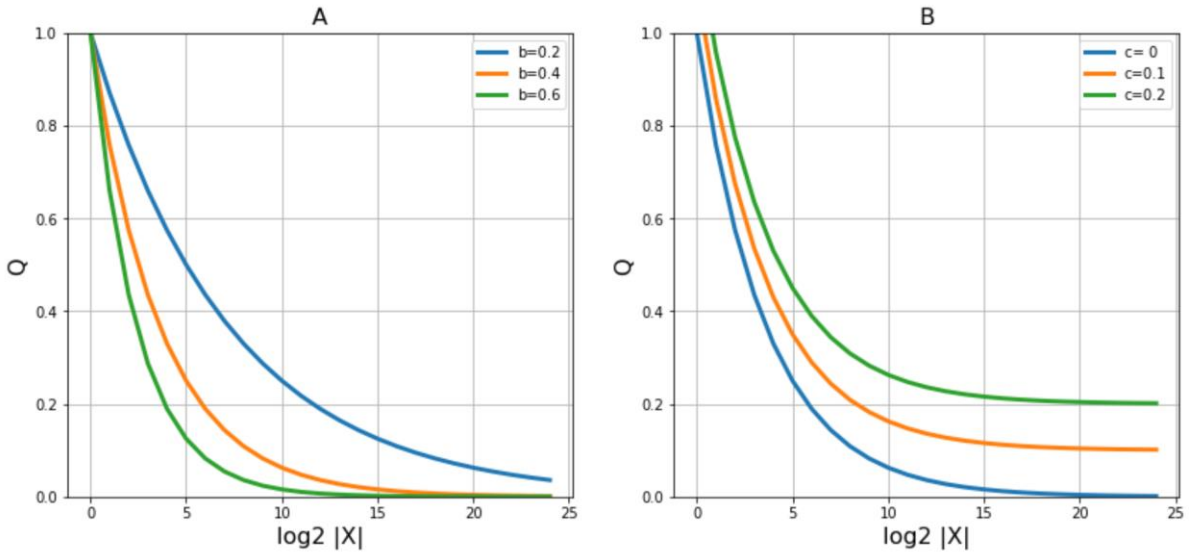

Рис 1.6. Зависимость меры качества от размера обучающей выборки для разных значений параметров b (слева) и c (справа).

Определение 7. Параметр c из (1.20) назовем *пределом потенциала* алгоритма определения живости $\mathbf{f}$ на выборке $\boldsymbol{\mathcal{D}}$ по контрольной выборке $\boldsymbol{\mathcal{D}}_{\mathbf{test}}$.

`

Предел потенциала показывает максимальное качество, которое можно получить на неограниченной по размеру обучающей выборке. В идеальном случае $c = 0$ при $|\boldsymbol{\mathcal{D}}| \to \infty$, но если задано ограничение сверху на количество параметров нейронной сети либо обучающая выборка покрывает не все множество внутренних условий $\boldsymbol{\theta}$, достичь 0 не всегда возможно (рис. 1.6.В).

Определение 8. Параметр b из (1.20) назовем *степенью эффективности* алгоритма определения живости $\mathbf{f}$ на выборке $\boldsymbol{\mathcal{D}}$ по контрольной выборке $\boldsymbol{\mathcal{D}}_{\mathbf{test}}$.

Степень эффективности чаще всего принимает значения из диапазона $[0, 1]$. Чем выше степень эффективности, тем меньше данных нужно алгоритму, чтобы достичь хорошего значения качества (рис 1.6.А).

В реальных условиях точно построить график зависимости качества от размера выборки не представляется возможным ввиду ограниченности размера выборки, но его можно экстраполировать по известным точкам, подобрав параметры a, b, c методом наименьших квадратов для точек $|\boldsymbol{\mathcal{D}}|, \frac{|\boldsymbol{\mathcal{D}}|}{2}, \frac{|\boldsymbol{\mathcal{D}}|}{4}, \ldots$

Рассмотрим случай, когда семейство моделей алгоритмов зафиксировано и есть процедура получения оптимального алгоритма по заданной выборке. В таком случае, улучшения качества можно добиться за счет изменения представления входных данных.

1.7. Выбор представления данных

Обычное цветное изображение лица X выбирается из множества изображений лиц $\mathbb{X}$, при этом

$$X \in \mathbb{X} \subset \mathbb{Z}_{[0,255]}^{3WH}, \tag{1.21}$$

`

где $\mathbb{Z}_{[0,255]}^{3\mathrm{WH}}$ – пространство матриц размера $3 \times W \times H$, состоящих из интенсивностей пикселей с значениями от 0 до 255. Аналогично, для трека длины N

$$T \in \mathbb{X}^{\mathrm{N}} \subset \mathbb{Z}_{[0,255]}^{3\mathrm{NWH}}, \tag{1.21}$$

$\mathbb{X}^{\mathrm{N}}$ – богатое пространство, объекты которого содержат множество мелких деталей, в том числе черты лица человека и элементы заднего плана. Поэтому, при обучении алгоритма определения живости на выборке небольшого размера либо собранной при очень ограниченных внутренних условиях $\boldsymbol{\theta}$, возможно переобучение на не имеющие отношения к живости признаки.

Пример 1. Треки реальных людей собраны из Интернета, треки атак собраны в лаборатории с узорчатой стеной на заднем плане. Самый простой разделяющий признак в данном случае – узор стены и нейронная сеть переобучится на него.

Пример 2. Реальные и поддельные данные собраны в одной и той же лаборатории. Но для атак использовались артефакты с биометрическими шаблонами знаменитостей из выборки лиц Celeba[A], которые в большинстве случаев содержат улыбку и необычные прически. Нейронная сеть может переобучиться на эти признаки и плохо работать для других доменов.

Уменьшить масштаб проблемы можно с помощью представления исходных данных в другом пространстве, которое минимизирует наличие не влияющих на живость деталей изображений и акцентирует внимание на полезных признаках.

Рассмотрим некоторую функцию $\phi: \mathbb{X}^N \rightarrow \mathbb{M}$, где $\mathbb{M}$ – пространство изображений размера $C \times W \times H$, т.е.

$$\phi(T) \in \mathbb{M} \subset \mathbb{R}^{CWH} \tag{1.22}$$

Пусть заданы обучающая и контрольная выборки обычных цветных изображений $\boldsymbol{\mathcal{D}}$ и $\boldsymbol{\mathcal{D}_{test}}$, а также процедура выбора оптимального алгоритма $\mathbf{f}^*(\boldsymbol{\mathcal{D}})$. Переведем выборки в пространство $\mathbb{M}$, т.е.

`

$$\widetilde{\boldsymbol{\mathcal{D}}} = \{(\phi(T_i), l_i)\ \forall\ (T_i, l_i) \in\ \boldsymbol{\mathcal{D}}\} \tag{1.23}$$

$$\widetilde{\boldsymbol{\mathcal{D}}}_{\boldsymbol{test}} = \{(\phi(T_i), l_i)\ \forall\ \ (T_i, l_i) \in \boldsymbol{\mathcal{D}}_{\mathbf{test}}\}$$

Построим оптимальные алгоритмы $\mathbf{f}^*(\boldsymbol{\mathcal{D}})$ и $\mathbf{f}^*(\widetilde{\boldsymbol{\mathcal{D}}})$ и посчитаем степени эффективности алгоритмов $\tilde{b}$ и b по контрольным выборкам $\boldsymbol{\mathcal{D}}_{\mathbf{test}}$ и $\widetilde{\boldsymbol{\mathcal{D}}}_{\boldsymbol{test}}$ соответственно.

Определение 9. Пространство $\mathbb{M}$ назовем *искусственной модальностью,* а функцию ϕ – функцией *преобразования модальности*, если $\tilde{b} > b$.

Если размер выборки небольшой, то алгоритм, построенный на изображениях искусственной модальности, будет работать лучше на контрольной выборке, чем алгоритм, обученный на оригинальных изображениях (рис. 1.7.).

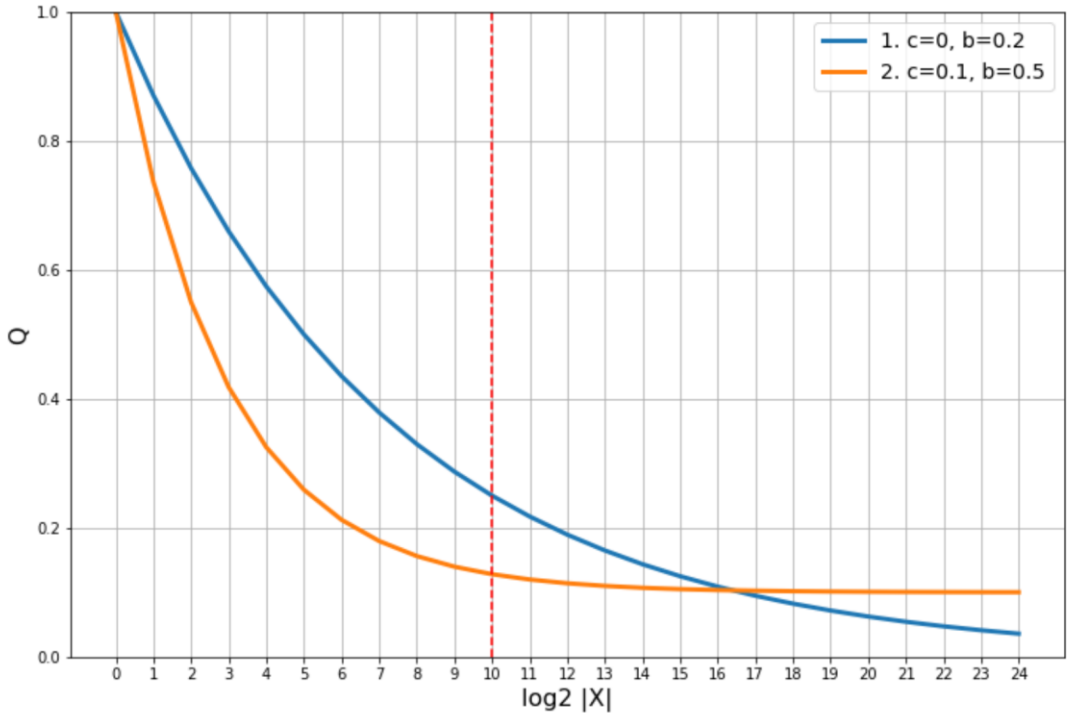

Рис 1.7. Пример алгоритмов, обученных на обычных изображениях (1) и изображениях искусственной модальности (2).

`

Для больших и разнообразных выборок предел потенциала алгоритмов на обычных изображениях выше, чем на изображениях из искусственной модальности, но в прикладных задачах собрать выборку такого размера чаще всего не представляется возможным. Рассмотрим примеры построения алгоритмов определения живости, которые используют искусственные модальности в том или ином виде.

1.8. Существующие методы определения живости по изображению лица

Одна из первых статей по данной теме [1] была опубликована в 2002 году. Авторы статьи пишут, что “определение живости основано на распознавании физиологической информации как признака подлинности биометрического шаблона”. В то же время для распознавания лиц достаточно ключевых черт, индивидуальных для пользователя. Первые методы определения подлинности изображения лица были опубликованы еще до эпохи глубокого обучения и основывались либо на текстурном анализе изображений, либо на кооперативности и обнаружении заданных действий объекта.

1.8.1. Текстурно-частотный анализ

Данный метод рассмотрен в [2]. Основная цель метода – различать живые лица и двумерные распечатанные маски на основе формы и детализации. Предложено рассмотреть низко- и высокочастотные спектры изображения и их отличия для подлинных и поддельных изображений, которые заключались в двух пунктах. Во-первых, различия распределения освещенности трехмерных форм, по которым сделаны фотографии, отображаются в низкочастотной информации. Во-вторых, детализация реальных лиц содержится в высокочастотной части. Для получения нужной информации изображения преобразованы с помощью двумерного дискретного преобразования Фурье. Наконец, одномерный вектор признаков аполучен комбинированием средних значений энергии для всех концентрических колец частотного пространства Фурье.

`

Для текстурного извлечения признаков применен популярный в то время метод Local Binary Pattern (LBP). Объединенное решение базировалось на методе опорных векторов (SVM) для классификации подлинности. Похожие решения с применением текстурного анализа опубликованы в [3, 4]. Основная идея – выделить различия текстур в признаковом пространстве с помощью LBP и классифицировать с помощью SVM.

1.8.2. Анализ изменения фокуса камеры

Метод оценки подлинности по изменению фокуса камеры впервые предложен в [5]. Ключевая идея – использовать различия в значениях пикселей двух последовательных изображений лица при изменении программно-регулируемого фокуса камеры. Предполагая, что за короткий срок съемки движение лица незначительно, авторы попытались находят в фокальном расстоянии для реальных и поддельных изображений. Для подлинных лиц некоторые регионы при сфокусированной съемке четкие в то время как другие расплывчаты из-за трехмерной структуры лица. Для распечатанных изображений уровень размытия примерно одинаков для всех областей. Основным ограничением метода является его зависимость от расстояния объектов до камеры, так как для разных расстояний уровень расфокусирования разный. Для оценки уровня фокусировки авторы используют сумму модифицированных лапласиан.

1.8.3. Анализ движения глаз

Метод описан в [6] как часть полномасштабной системы биометрического распознавания. Рассматривается стандартное отклонение значений пикселей в области глаз для серии последовательных кадров и утверждалось, что для реальных изображений вариативность выше, чем для статических подделок. Для увеличения устойчивости метода применяются различные алгоритмы нормализации изображений и удаления эффектов освещенности.

`

1.8.4. Анализ оптического потока

Первое использование алгоритмов оптического потока для задач определения подлинности описано в [7], где проводится анализ различий оптического потока для двумерных и трехмерных плоскостей. На основании разницы в значениях компонент потока для поддельных и реальных изображений лиц подобран порог классификации. Дальнейшее развитие оптического потока для определения живости показано в работах [8] и [9]. Идея использования оптического потока для задачи определения подлинности взята за основу для некоторых алгоритмов, представленных в данной диссертационной работе.

1.8.5. Анализ моргания

Алгоритмы, основанные на выявлении морганий, стали основоположниками семейства кооперативных методов определения живости. Впервые такой метод представлен в [10]. Применяется Conditional Random Fields (CRF) для моделирования естественных паттернов моргания у реальных людей и полученное распределение используется для определения поддельных видеозаписей, где моргание генерируется случайно из статических изображений с открытыми и закрытыми глазами. Метод усовершенствован в работах [11, 18, 27].

1.8.6. Анализ кодирования компонент лиц

Технология кодирования частей лица разработана в [12]. Предложенный метод, состоит из четырех шагов: (1) выделение компонент лица; (2) кодирование низкоуровневых признаков для всех компонент; (3) синтез высокоуровневого представления лица из дескрипторов компонент с весами по критерию Фишера; (4) объединение гистограмм всех компонент в один вектор и последующая классификация. Метод основан на трех основных различиях реальных и поддельных изображений: подделки более размыты, так как фотографировался уже воспроизведенный артефакт; для реаль-

`

ных лиц изображение лица зависит от параметра гамма коррекции камеры; для подделок распределение теней на изображении неестественно. Итоговое решение включает в себя классификацию с помощью SVM дескрипторов частей лица, полученных алгоритмами извлечения признаков LBP и HOG.

1.8.7. Анализ трехмерной структуры лица

В [13] предложен метод, основанный на анализе трехмерных признаков лица, полученных из двумерной фотографии. Показано, что восстановленные признаки формы лица для изображений реальных людей отличаются от поддельных. В работах [14, 15, 16] рассматривается метод восстановления трехмерной формы по набору последовательных кадров, снятых с разного ракурса, что частично эквивалентно стереокамере, и последующее построение классификатора для определения живости. Недостаток данного семейства методов в высокой вычислительной сложности построения коэффициентов трехмерного объекта.

1.8.8. Анализ признаков фона

Еще одно семейство решений [17, 19, 20] нацелено на извлечение полезных признаков из анализа фона вокруг лица. Оценивается движение пикселей фона и его стабильность, степень “аффинности” заднего плана (распечатки часто бывают согнутыми, из-за чего геометрия сцены искажается) и другие моменты, которые могут свидетельствовать о фальсификации. Данные методы можно использовать как дополнение к основным, нацеленным на изображение лица.

`

Глава 2

Кооперативные методы определения живости

Первые работы по решению задачи определения живости [8,9] относятся к кооперативным методам и направлены на защиту от атак уровня A, то есть от неподвижных фальсификаций. Идея таких методов – попросить пользователя сделать какое-то машинно-определяемое действие, которое невозможно повторить статическим артефактом. Это действие выявляется сторонним алгоритмом, после чего метод определения живости представляет собой простое логическое “или” – состоялось действие или нет. В таком случае точность определения живости ограничена сверху точностью определения этого действия.

В донейросетевую эру компьютерного зрения алгоритмы справлялись с выявлением только простых действий, поэтому ранние работы фокусировались на определении моргания или улыбки как признака подлинности человека.

В последствии, с появлением нейронных сетей, увеличением количества и объема выборок по атрибутам лиц и возросшему интересу к тематике, количество доступных действий для проверки подлинности увеличилось [10]. Помимо улыбки и моргания, стали использовать поворот головы в определенном направлении [11], а также направление взгляда человека [12].

`

Ключевым недостатком таких кооперативных методов является простота взлома с помощью видео. Даже не обладая полным видео, где объект выполняет требуемое действие, можно отредактировать запись, добавив закрытые глаза или улыбку.

Другое направление работ по кооперативным методам оценки живости нацелено на анализ текстур лица и окружения при воздействии светом определенного диапазона. Последовательно зажигая подсветку разных цветов и узоров на экране устройства, алгоритмы считывают распределение света на изображении лица пользователя, строя на этом метод определения живости [15]. Такие методы уже требуют данных реальных и поддельных лиц для обучения модели различия цветовых отражений, но при этом способны противодействовать видеозаписям – атакам уровня B. Но на практике, алгоритмы по отражению света не нашли широкого применения, т.к. помимо ограниченности сценариев применения (невозможно использовать при дневном или ярком освещении, а также на большом расстоянии от экрана устройства), оказались крайне раздражающими для конечных пользователей систем.

В данном разделе предлагается обобщение простых кооперативных методов в комплексную модель, способную справиться с атаками уровня B и некоторыми атаками уровня C, а также предлагается метод определения живости по оптическому потоку.

Рассмотрим задачу определения живости в мобильном сценарии $S_{\text{моб}}$ для изображений из видимого диапазона M_{RGB}, распределение (1.10) в таком случае будет иметь вид

$$\boldsymbol{\mathcal{D}} \sim p(T|\boldsymbol{\theta}, S_{\text{моб}}, I, M_{RGB})p(T|\{\boldsymbol{PA}\}) \tag{2.1}$$

Требуется построить алгоритм $\mathbf{f}(T, \mathbf{w})$, такой что

$$\frac{1}{m}\sum_{1}^{m} \mathcal{L}\left(\mathbf{f}(T_i, \mathbf{w}),\ l_i\right) \to \min$$

`

Рассмотрим случай, когда $m \to 0$, т.е. когда в наличии один или несколько примеров на каждый класс. В таком случае большинство методов машинного обучения неприменимы. Однако, доступные примеры неслучайны и зависят от кооперативности алгоритма I, т.е. структура алгоритма *управляет* пространством данных. В таком случае, можно построить алгоритм оценки живости исходя из априорного представления о поведении реального и поддельного классов при заданном I.

2.1. Атомарный метод определения живости

Пусть дан трек $T = \{X_i\},\ i = 1, \dots, N$, на котором непрерывно присутствует лицо одного человека и его метка живости l.

Назовем *алгоритмом атрибута* некоторый алгоритм $\mathcal{K}$ компьютерного зрения, который по заданному кадру X_i возвращает некоторое действительное число, вектор или метку класса k_i:

$$\mathcal{K}(X_i) = k_i \tag{2.2}$$

Назовем *атомом* $\mathcal{A}$ алгоритм определения живости , который по последовательности $\{k_i\}$ и фиксированным гиперпараметрам $\boldsymbol{\gamma}$ определяет бинарный ответ живости l, при этом

$$\mathcal{A}(\{\mathcal{K}(X_i)\}, \boldsymbol{\gamma}) = l, \tag{2.3}$$

`

Построим конкретное семейство атомов, которое оценивает бинарное действие пользователя. Пусть $\boldsymbol{\psi}$: $\mathbb{R}^N \rightarrow \mathbb{R}$ – функция агрегации последовательности действительных чисел, а $\boldsymbol{\gamma}$ – набор гиперпараметров. Введем функции агрегации и гиперпараметры, которые будут использоваться для построения атомов.

Функции агрегации для последовательности $\{k_i\}, i = 1, \dots, n$:

1. $\boldsymbol{\psi}_{\mathbf{max}}(\{k_i\}) = \max(\{k_i\})$
2. $\boldsymbol{\psi}_{\mathbf{avg}}(\{k_i\}) = \frac{1}{n}\sum_i^n k_i$
3. $\boldsymbol{\psi}^{>}_{\mathbf{max/avg}}(\{k_i\}, x) = [\boldsymbol{\psi}_{\mathbf{max/avg}}(\{k_i\}) > x]$
4. $\boldsymbol{\psi}^{\leq}_{\mathbf{max/avg}}(\{k_i\}, x) = [\boldsymbol{\psi}_{\mathbf{max/avg}}(\{k_i\}) \leq x]$

где [] – оператор, равный 1, если условие в скобках выполняется и 0 в противном случае.

Для оценки бинарного действия разделим исходную последовательность на две части:

$$K_1 = \{k_i\}, i = 1, \dots, \lceil dN \rceil, \qquad (2.4)$$
$$K_2 = \{k_i\}, i = \lceil dN \rceil + 1, \dots, N,$$

где $\lceil\ \rceil$ – целая часть числа, d – гиперпараметр, показывающий, какую долю трека отнести к первой части. В частности, если $d = \frac{1}{N}$, то первая часть будет состоять из одного кадра. Формулу (2.4) сокращенно запишем как

$$K_1, K_2 = \boldsymbol{\psi}_{\mathbf{split}}(\{k_i\}, d) \qquad (2.5)$$

Пример. Рассмотрим пример атома – кооперативного метода оценки живости по улыбке. Система просит пользователя смотреть в камеру с нейтральным выражением лица примерно одной секунду, после чего улыбаться в течение еще одной секунды. Для такого алгоритма нужен метод определения улыбки и небольшая если-то надстройка, агрегирующая все кадры. Рассмотрим такой метод с алгоритмической точки зрения.

`

Пусть $\mathcal{K}$ – алгоритм, определяющий наличие улыбки на лице пользователя, т.е. возвращающий число 1, если улыбка присутствует и 0, если отсутствует. Метод атрибута переводит трек в бинарную последовательность длины N. Разобьем трек на две равные части по $N/2$ кадров. Если в первой половине большинство нулей, а во второй – большинство единиц (например, 75% от общего количества), то будем считать, что пользователь прошел проверку на живость.

Пример псевдокода для такого атома $\mathcal{A}$ показан на рис 2.2. В данном случае, гиперпараметрами алгоритма являются разбиение трека на 2 части поровну и 75% как критерий большинства для определения статуса в каждой из половин. При внедрении, эти параметры можно вынести в отдельный конфигурационный файл и настраивать в процессе работы.

Вход: $\{X_i\}$, $\mathcal{K}$
Выход: l
1: **для** $i = 1, \dots, N$
2: $\quad k_i = \mathcal{K}(X_i)$
3: $s_1 = \sum_0^{\frac{N}{2}-1} k_i$
4: $s_2 = \sum_{\frac{N}{2}}^{N} k_i$
5: **если** $\left(s_1 \leq 0.25 * \frac{N}{2}\right)$ **и** $\left(s_2 > 0.75 * \frac{N}{2}\right)$, **то:** $l = 1$
6: **иначе:** $l = 0$

Рис 2.1. Псевдокод алгоритма атомарного алгоритма по улыбке.

`

С учетом введенных функций агрегации, алгоритм атома можно описать проще. В данной работе предлагается восемь атомов, рассмотрим их подробнее.

1. Атом по улыбке. $\mathcal{K}$ – алгоритм определения наличия улыбки. Пользователя просят смотреть в камеру с нейтральным выражением лица, потом улыбаться. $\mathcal{K}$ возвращает бинарный ответ, где 1 – улыбка есть, 0 – улыбки нет.

Гиперпараметры: соотношение длины подсессий d, жесткость наличия атрибута в подсессии t (в примере выше – $t = 75\%$).

Алгоритм:

$$K_1, K_2 = \boldsymbol{\psi}_{\mathbf{split}}(\{k_i\}, d)$$

$$\mathcal{A}(\{k_i\}) = \boldsymbol{\psi}^{\leq}_{\mathbf{avg}}(K_1, t) * \boldsymbol{\psi}^{>}_{\mathbf{avg}}(K_2, t)$$

2. Атом по открытому рту. $\mathcal{K}$ – алгоритм определения открытого рта. Пользователя просят смотреть в камеру с нейтральным выражением лица, потом открыть рот. $\mathcal{K}$ возвращает бинарный ответ, где 1 – рот открыт, 0 – рот закрыт.

Гиперпараметры: соотношение длины подсессий d, жесткость наличия атрибута в подсессии t (в примере выше – $t = 75\%$).

Алгоритм:

$$K_1, K_2 = \boldsymbol{\psi}_{\mathbf{split}}(\{k_i\}, d)$$

$$\mathcal{A}(\{k_i\}) = \boldsymbol{\psi}^{\leq}_{\mathbf{avg}}(K_1, t) * \boldsymbol{\psi}^{>}_{\mathbf{avg}}(K_2, t)$$

3. Атом по поднятым бровям. $\mathcal{K}$ – алгоритм определения 68 ключевых точек лица и выделения координаты линии бровей. Пользователя просят смотреть в камеру с

`

нейтральным выражением лица, потом поднять брови. Фиксируется положение бровей первых нескольких кадров, после чего ожидается пока положение бровей на любом из следующих кадров не отклонится от зафиксированного на порог сдвига.

Гиперпараметры: соотношение длины подсессий d, минимальное расстояние между спокойным и поднятым положением бровей t.

Алгоритм:

$$K_1, K_2 = \boldsymbol{\psi}_{\mathbf{split}}(\{k_i\}, d)$$
$$\mathcal{A}(\{k_i\}) = \boldsymbol{\psi}^{>}_{\mathbf{max}}(K_2 - \boldsymbol{\psi}_{\mathbf{avg}}(K_1), t)$$

4-7. Атом по повороту головы. $\mathcal{K}$ – алгоритм определения углов поворота головы yaw, pitch, roll [55]. Пользователя просят смотреть в камеру с нейтральным выражением лица, потом повернуть голову вправо/влево/вниз/вверх. Фиксируются углы поворота головы первых нескольких кадров, после чего ожидается пока угол поворота головы на любом из следующих кадров не отклонится от зафиксированного на порог сдвига в заданном направлении (например, для атома с поворотом головы вниз контроль осуществляется по параметру pitch в сторону увеличения)

Гиперпараметры: соотношение длины подсессий d, минимальное расстояние между спокойным и поднятым углом поворота головы t.

Алгоритм:

$$K_1, K_2 = \boldsymbol{\psi}_{\mathbf{split}}(\{k_i\}, d)$$
$$\mathcal{A}_{\text{влево}}(\{k_i\}) = \boldsymbol{\psi}^{>}_{\mathbf{max}}(K_2^{yaw} - \boldsymbol{\psi}_{\mathbf{avg}}(K_1^{yaw}), t)$$
$$\mathcal{A}_{\text{вправо}}(\{k_i\}) = \boldsymbol{\psi}^{>}_{\mathbf{max}}(\boldsymbol{\psi}_{\mathbf{avg}}(K_1^{yaw}) - K_2^{yaw}, t)$$
$$\mathcal{A}_{\text{вниз}}(\{k_i\}) = \boldsymbol{\psi}^{>}_{\mathbf{max}}(K_2^{pitch} - \boldsymbol{\psi}_{\mathbf{avg}}(K_1^{pitch}), t)$$
$$\mathcal{A}_{\text{вверх}}(\{k_i\}) = \boldsymbol{\psi}^{>}_{\mathbf{max}}(\boldsymbol{\psi}_{\mathbf{avg}}(K_1^{pitch}) - K_2^{pitch}, t)$$

`

8. Атом по морганию. $\mathcal{K}$ - алгоритм определения статуса глаз (1 – открыты, 0 - закрыты). Пользователя просят моргнуть. Ищется последовательность открыты-закрыты-открыты в ответах метода атрибута. Если последовательность найдена, алгоритм оценки живости возвращает статус “Пройдено”.

Гиперпараметры: количество кадров статуса “открыты” n_o, количество кадров “закрыты” n_c в последовательности открыты-закрыты-открыты; доля допустимой погрешности в последовательности t.

Алгоритм:

$$\forall d = 1, \dots, N - 2n_o - n_c: \mathrm{K}_1, \mathrm{K}_2, K_3 = \{k_i\}, \{k_j\}, \{k_v\},$$

$$i = d, \dots, d + n_o, j = d + n_o + 1, d + n_o + n_c, v = d + n_o + n_c + 1, \dots, d + 2n_o + n_c$$

$$\mathcal{A}(\{k_i\}) = \boldsymbol{\psi}^{\geq}_{\mathbf{avg}}(K_1, 1 - t) * \boldsymbol{\psi}^{\leq}_{\mathbf{avg}}(K_2, t) * \boldsymbol{\psi}^{\geq}_{\mathbf{avg}}(K_3, 1 - t)$$

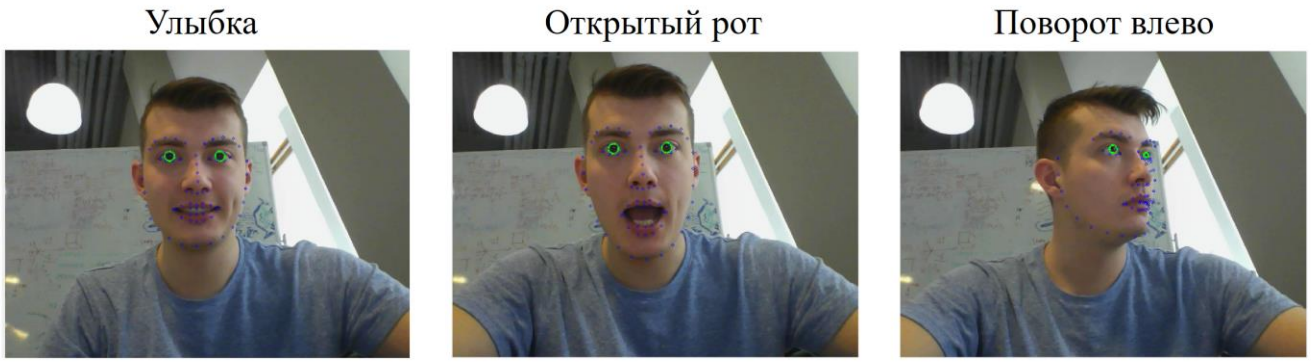

Рис 2.2. Примеры запрашиваемых разными атомами действий.

Качество работы атома зависит от двух факторов – точность следованию инструкции пользователем и точностью метода атрибута. Первый фактор можно измерить только в живом тестировании, задействовав множество людей в разных сценариях. Для статистически значимой оценки качества работы атома в пределах 1%

`

нужно как минимум 10 разных людей, снимающихся в 10 разных локациях. Кроме этого, потребуется зафиксировать виды атак и провести равнозначное по количеству попыток тестирование взлома. Но и без живого тестирования можно получить верхнюю оценку точности работы атома. Верхняя оценка получается при условии, что мы полагаем точность следования инструкциям 100%. Тогда качество атома зависит только от качества метода атрибута.

Настройку гиперпараметров атомов можно провести всего по нескольким трекам, подобрав комфортные пороги для реальных людей и априори полагая, что алгоритм будет защищать от артефактов, которыми нельзя сделать запрашиваемое действие. В частности, любая статическая двумерная атака не сломает атомы с предложенными движениями, а силиконовая трехмерная маска не сломает атом по поднятым бровям. Основной недостаток таких атомов – это невозможность противодействия динамическим атакам. Но если объединить несколько атомов подряд в одном методе, то подобрать противодействие становится сложнее. Каждый из атомов по отдельности уязвим против динамической атаки – записанного на мобильное устройство повторения прохождения процедуры. Для противодействия таким атакам предлагается комбинация атомов в *мультиатомарный* алгоритм.

Вход: набор атомов $\{\mathcal{A}_i\}$, $i = 1, \dots, n$; количество проверок m.
Выход: l
1: **выбрать** $i_1, i_2, \dots, i_m$ случайных чисел из $1, \dots, n$.
2: **запросить у пользователя** треки по действиям $\left\{\mathcal{A}_{i_j}\right\}, j = 1, \dots, m : \{T_{i_j}\}$
3: $a_{i_j} = \mathcal{A}_{i_j}(\mathcal{K}_{i_j}\left(T_{i_j}\right))$
4: $l = \prod_{j=1}^{m} a_{i_j}$

Рис 2.3. Мультиатомарный алгоритм определения живости.

`

Мультиатомарный алгоритм (рис. 2.3) состоит из нескольких атомов, выполняющихся в произвольной последовательности непрерывно. Простая агрегация отдельных атомов в связанную последовательность и расставление компонент в произвольном порядке существенно улучшают защиту от различных видов атак. Так, отдельные атомы не обеспечивают защиту от динамических фальсификаций, но случайная последовательность, подкрепленная требованием непрерывности, такую защиту обеспечить может. Количество возможных случайных выборов m активных атомов из n базовых можно посчитать как количество размещений $A_n^m = \frac{n!}{(n-m)!}$. Например, для 8 активных и 4 базовых это 1680 различных вариантов. Даже если у злоумышленника есть видео отдельных атомов, подготовить нужную комбинацию в процессе авторизации будет практически невозможно.

Предлагаемый алгоритм определения подлинности также защищает от силиконовых и керамических трехмерных масок. Если маска неподвижна, то такая атака не пройдет атомы, основанные на движении мимических мышц. Самые дорогие маски оставляют открытыми области рта и глаз, обходя такие атомы. Для противодействия сложным случаям предлагается атом по поднятым бровям. Данный вид кооперативной проверки не был описан в научных работах и патентах по определению живости, соответственно, существующие трехмерные артефакты взлома не ориентированы на подвижность области бровей и как следствие, не могут пройти предложенный атом.

Кооперативные динамические методы были первыми из семейства систем биометрической защиты, которые были реализованы на практике. Они до сих пор используются в случаях, когда есть основной метод защиты (например, проверка документа), и нужна простая проверка биометрии. В таких сервисах можно встретить отдельные атомы из приведенных выше. Главными недостатками атомарного алгоритма является время прохождения процедуры и степень вовлеченности пользователя. С развитием пользовательских сервисов и упрощением взаимодействия клиентов с приложениями

`

тренд разработки сместился в сторону уменьшения кооперативности в алгоритмах определения подлинности лица.

2.2. Определение живости по оптическому потоку

Несмотря на то, что атомарный метод определения живости защищает от атак уровня A, B и C, минимальное время для прохождения такой процедуры верификации составляет около 30 секунд, что не удовлетворяет требованиям некоторых практических сценариев. С другой стороны, в сценариях (например, подтверждение присутствия сотрудника на рабочем месте), где скорость и удобство важнее, чем степень защиты, допускается жертвование уровнем надежности в угоду комфорту.

Рассмотрим задачу (2.1) в условиях, когда данные для обучения присутствуют, но в недостаточном для создания хорошей нейросетевой модели количестве. В таком случае, предлагается использовать концепцию искусственной модальности.

В данном разделе предлагается кооперативный метод с защитой от атак уровня A и B. требующий меньше времени и усилий со стороны пользователя. Идея метода состоит в использовании оптического потока [56, 73] между двумя кадрами лица, снятыми в разные промежутки времени.

Оптический поток – это движение объектов между двумя изображениями одного трека, вызванное относительным перемещением наблюдаемого относительно наблюдателя.

`

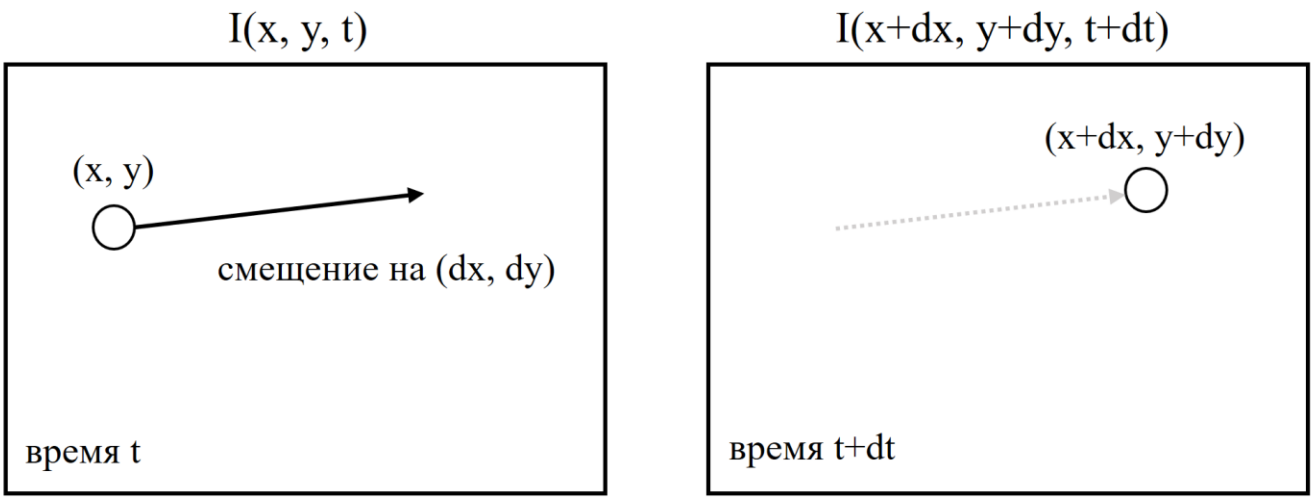

Рис 2.4. Оптический поток между двумя изображениями.

В процессе такого перемещения, пиксели объекта изменяют свое местоположение от первого ко второму изображениям. Полагая интенсивность I перемещенных пикселей постоянной, получаем что

$$I(x, y, t) = I(x + dx, y + dy, t + dt)$$

Считая, что перемещение мало, раскладываем правую часть уравнения по ряду Тейлора:

$$I(x, y, t) = I(x + dx, y + dy, t + dt) = I(x, y, t) + \frac{\partial I}{\partial x} dx + \frac{\partial I}{\partial y} dy + \frac{\partial I}{\partial t} dt,$$

откуда следует, что

$$\frac{\partial I}{\partial x} dx + \frac{\partial I}{\partial y} dy + \frac{\partial I}{\partial t} dt = 0 \qquad (2.6)$$

Разделим (2.6) на dt и положим $V_x = \frac{dx}{dt}$, $V_y = \frac{dy}{dt}$. Получим

$$\frac{\partial I}{\partial x} V_x + \frac{\partial I}{\partial y} V_y + \frac{\partial I}{\partial t} = 0 \qquad (2.7)$$

`

V_x, V_y – горизонтальная и вертикальная компоненты скорости оптического потока для точки (x, y), а производные в уравнении – это градиенты изображения по горизонтали, вертикали и времени. Уравнение (2.7) содержит две неизвестных и не может быт решено однозначно. Наиболее известный алгоритм поиска оптического потока – алгоритм Лукаса-Канаде [47]. Но, с развитием нейронных сетей появились модели, которые получают оптический поток более высокого качества. На данный момент лучшим алгоритмом в определении оптического потока является PWCnet [56]. В дальнейшем, если не указано обратное, под методом оптического потока будет подразумеваться данный алгоритм.

Компоненты скорости оптического потока, посчитанные для каждой точки исходного изображения, могут быть преобразованы в двухканальное изображение (для этого достаточно нормировать значения к диапазону [0,255]).

На рис. 2.5. показана визуализация компонент для двух кадров из одного трека. Отличительной особенностью оптического потока является то, его визуализация сильно коррелирует с трехмерным изображением, которое могло бы быть получено с помощью специальной камеры или по стереопаре. Глубина – сильная в плане признаков модальность для решения задачи определения живости, соответственно, и оптический поток может быть использован как промежуточное звено для определения подлинности изображения.

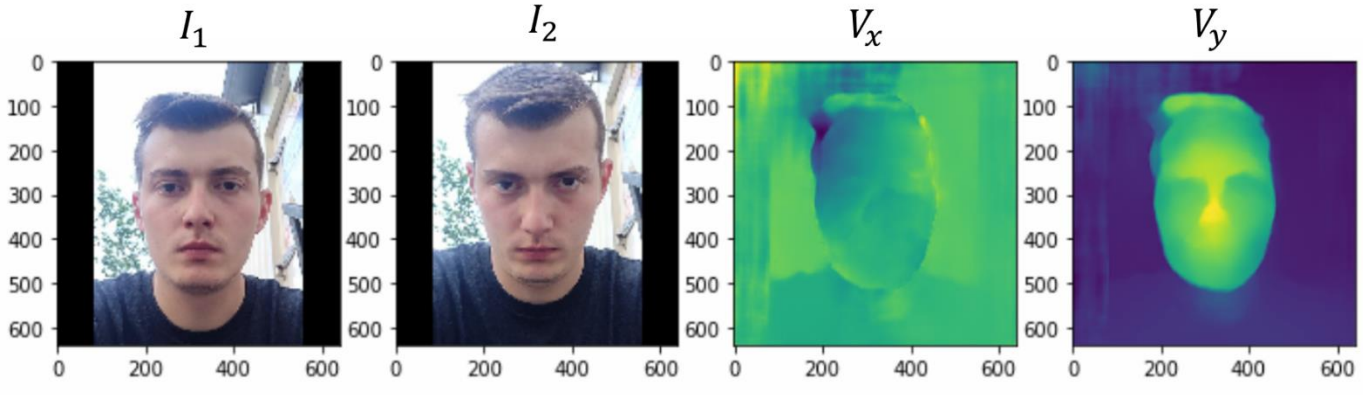

Рис 2.5. Пример работы PWCnet на двух изображениях из одного трека.

`

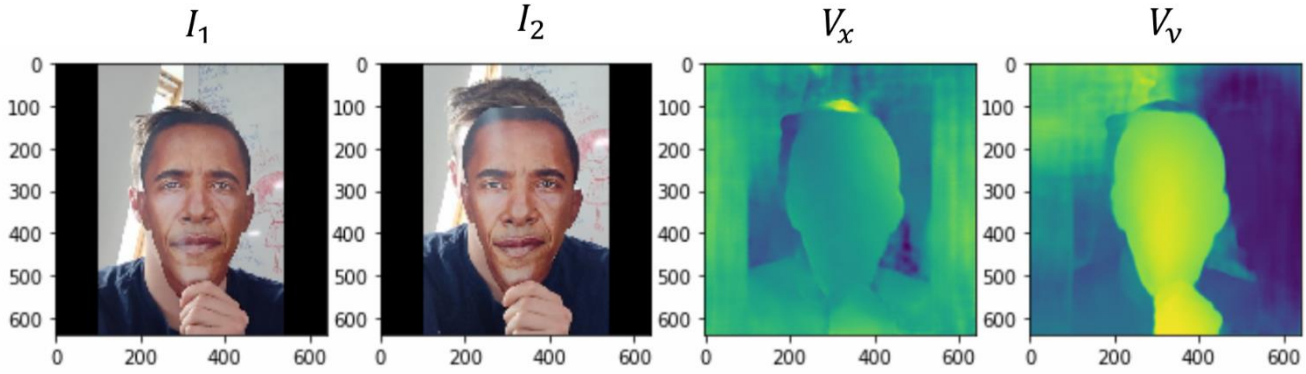

Рис 2.6. Пример оптического потока для вырезанной маски.

На рис 2.6. показан пример визуализации компонент оптического потока для атаки распечатанной вырезанной маской. На примерах 2.5 и 2.6 голова наклонялась на одинаковую величину. На изображении реального человека можно четко увидеть трехмерную структуру лица, в то время как для распечатанной маски интенсивность пикселей в районе лица равномерная.

Для корректной работы оптического потока требуется высокое разрешение фотографии, поэтому методы определения живости на его основе могут применяться только в мобильных и стационарных сценариях. На рис. 2.7 показана деградация компоненты оптического потока при уменьшении разрешения входных данных, где видно, что качество сильно зависит от размерности входящей пары изображений. Но, с уменьшением размерности, падает время на обработку оптического потока. На рис 2.4. показано время работы на графическом процессоре GeForce GTX 1080Ti. Оптимальным по соотношению скорость качество после визуального анализа является размер 448 пикселей. При этом лицо на кадре занимает 224 пикселя. Такой размер обеспечивается большинством современных видеокамер на мобильных и стационарных устройствах, поэтому был принят в качестве основного в дальнейших исследованиях.

`

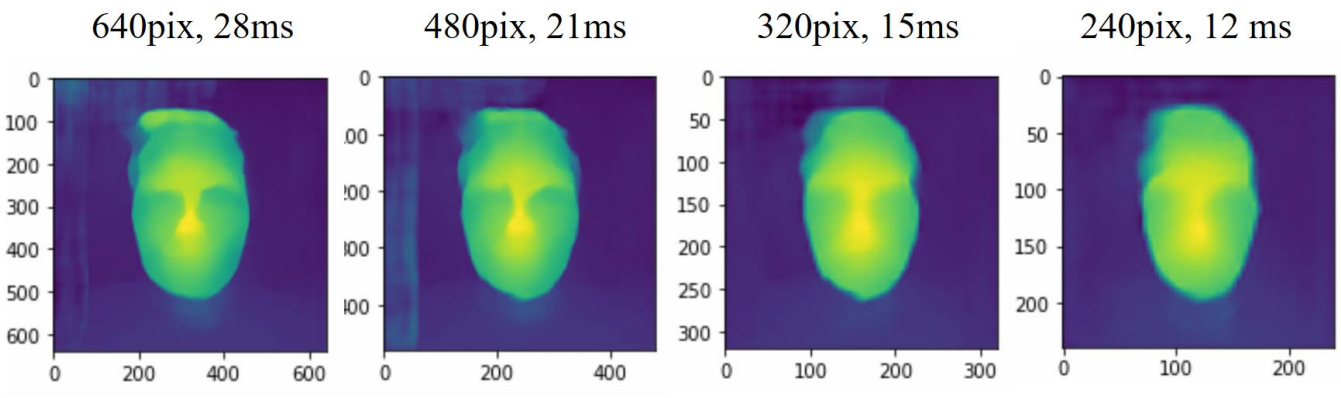

Рис 2.7. Сравнение качества оптического потока для данных разного разрешения.

Кооперативные атомы по повороту головы, описанные в предыдущем разделе, также требуют от пользователя движение головой. Но поворот головы в таких атомах должен составлять порядка 30 градусов (при меньшей амплитуде возможен взлом наклоном распечатанного артефакта). В случае определения живости по оптическому потоку достаточно, чтобы пользователь повернул голову или снимающее устройство на 5-7 градусов – именно таких углов достаточно для визуально различимой трехмерной структуры лица.

Для создания классификатора оценки живости по оптическому потоку необходима обучающая выборка. Но, в отличие от классификатора по сырым RGB изображениям, тут для обучения качественной модели требуется на порядок меньше данных, так как оптический поток убирает множество мелких деталей, сужая признаковое пространство, и тем самым позволяя избежать переобучения алгоритма.

Пусть задана обучающая выборка реальных и поддельных треков в условиях кооперативного условия I – легкого поворота головы вбок, т.е. для каждого трека $T \in \boldsymbol{\mathcal{D}}$ выполняется условие:

$$I\colon\ \mathbf{yaw}(\mathrm{X}_{i+1}) \geq \mathbf{yaw}(X_i) - \epsilon \quad \forall i = 2, \dots, |T|, \tag{2.8}$$

`

где **yaw** – угол поворота головы вбок, $|T|$ – длина трека, ϵ – допустимая погрешность в разнице углов между кадрами.

Пусть $\mathbb{M}_{OF} \subset \mathbb{R}^{2WH}$ – искусственная модальность оптического потока, где

$$\phi(T) = PWCnet(X_1, X_c) = \left[V_x, V_y\right] -$$

функция преобразования модальности, переводящая трек T в матрицу действительных чисел размера $2 \times W \times H$ – компонент скорости оптического потока, а пара кадров выбирается из трека по принципу

$$\forall i\colon \mathbf{yaw}(X_i) - \mathbf{yaw}(\mathrm{X}_1) \geq t, c = \min\{i\},$$

т.е. фиксируется первый кадр трека и выбирается первый по времени кадр такой, что разница углов между кадрами минимум t (гиперпараметр алгоритма).

Далее, выбирается семейство моделей, исходя из бюджета на время обработки одного трека. После чего применяется процедура получения оптимального алгоритма, описанная в главе 1, и получается итоговый алгоритм $\mathbf{f}^*(\phi(T), \mathbf{w}^*)$.

2.3. Практическая реализация оценки живости по оптическому потоку

Рассмотрим реализацию предложенного метода на практике. В качестве сценария применимости был выбран мобильный, для упрощения разработки использовался клиент-серверный подход. Пользователя просят снять фронтальное видео своего лица, при этом в процессе слегка повернуть голову вбок/вниз или подвигать телефон влево-вправо. Полученный ролик длительностью в 2-4 сек загружается в чат-бота в приложении на телефоне. Приложение отправляет видео на заданный сервер, на котором результат обрабатывается, после чего результат оценки живости отсылается обратно на телефон (рис. 2.8).

`

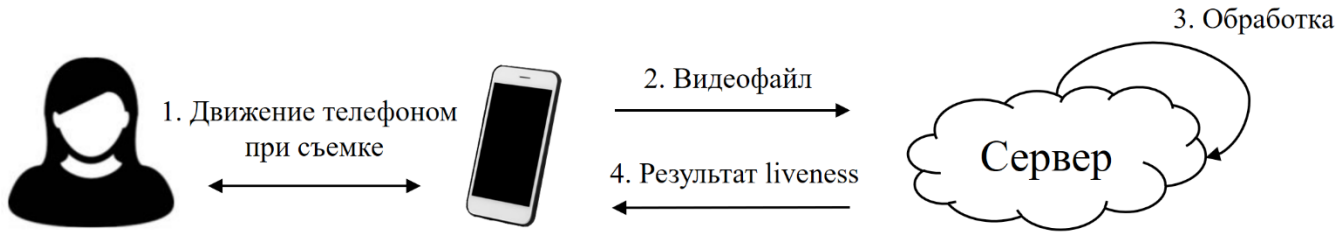

Рис. 2.8. Процесс использования метода оценки живости по оптическому потоку.

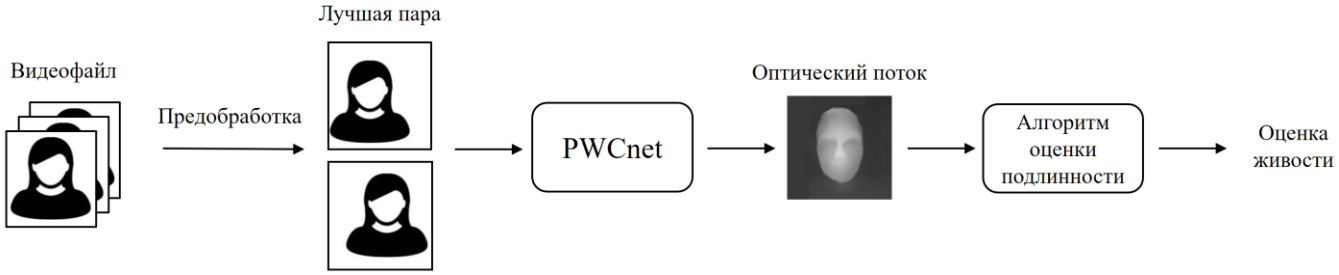

Рис. 2.9. Схема работы сервера оценки живости по оптическому потоку.

Процедура работы сервера (рис. 2.9):

1. Видеофайл обрабатывается детектором лиц, получается трек $\{X_i\}$.
2. Для каждого кадра из трека считаются углы поворота головы $roll_i$, yaw_i, $pitch_i$.
3. Выбирается X_j если $\max(|roll_j - roll_0|, |yaw_j - yaw_0|, |pitch_j - pitch_0|) > 7$.
4. Считается оптический поток $[V_x, V_y] = PWCnet(X_0, X_j)$.
5. Считается живость $l = \mathbf{f}^*([V_x, V_y])$.

`

2.3.1. Сбор данных

Для обучения предложенной модели определения живости требуется выборка реальных и поддельных пар изображений, но таких данных в открытом доступе нет, поэтому выборка была собрана самостоятельно. На первом этапе было скачано 500 роликов из видео хостинга Youtube. Видео отбирались по присутствию фронтально смотрящего в камеру человека. Далее, видео обрабатывались детектором лиц и алгоритмом углов поворота головы. Все возможные пары кадров, подходящие под п.3. процедуры работы сервера, добавлялись в формируемую выборку – итого получилось ~40 000 пар.

Для формирования пар с поддельными лицами, из выборки CelebaHQ [57] были отобраны 60 лиц высокого разрешения. Далее, лица были распечатаны на бумаге размера А4, для 30 артефактов дополнительно вырезали контуры лица. Каждый из полученных артефактов снимался на мобильную камеру в разных условиях освещения, попытками сгибания и поворота артефакта. Для каждой подделки было снято по 5 коротких роликов, получая итого 300 видео атак. Видео были обработаны детектором лиц, после чего из каждого видео случайно были выбраны по 10 пар изображений, всего 30 000 пар. Сформированная выборка $\boldsymbol{\mathcal{D}}$ была разбита на $\boldsymbol{\mathcal{D}_{\mathbf{train}}}$ и $\boldsymbol{\mathcal{D}_{\mathbf{val}}}$ в соотношении 90% и 10%.

Чтобы проверить потенциальную работоспособность в практическом приложении, тестовая выборка должна быть максимально близко похожа на реальные условия применения. Для создания тестовой выборки был разработан telegram-бот, реализующий схему на рис. 2.8. В процессе сбора данных приняло участие 40 человек, которые снимали себя, а также распечатанные и демонстрируемые с экранов телефонов и планшетов подделки. Описание итоговой статистики тестовой выборки показано в таблице 2.1.

`

Таблица 2.1. Статистика тестовой выборки для метода определения живости по оптическому потоку.

Тип данных	Описание	Количество
Real	Реальный человек.	160
P1	Распечатанное изображение (например, на A4) без модификаций. Края бумаги видны в кадре.	63
P2	Распечатанное изображение (например, на A4) без модификаций. Края бумаги не видны в кадре.	34
P3	Распечатанное изображение лица, вырезанное по контуру.	99
D1	Статическое изображение лица (как из социальных сетей), на экране телефона. Границы телефона находятся в кадре.	63
D4	Видео с атакуемым повторяюще близкие движения к тому, что требуется в системе.	59

2.3.2. Обучение модели

В качестве основного алгоритма для решения задачи определения живости по выборке $\mathcal{D}$ использовалась глубокая нейронная сеть. Архитектура – Mobilenet [49] с урезанным до 256 количеством нейронов на последнем линейном слое. На выходе применялась сигмоидная функция активации

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

`

которая возвращает число в диапазоне от 0 до 1, что соответствует вероятности наличия атрибута живости. В качестве функции потерь использовалась бинарная кросс-энтропия

$$\mathcal{L} = -(l \log s + (1 - l) \log(1 - s)), \quad (2.5)$$

где l – метка живости, s – выход нейронной сети. Гиперпараметры сети оптимизировались на $\boldsymbol{\mathcal{D}_{\text{train}}}$, лучшая эпоха выбиралась по значению функции потерь на $\boldsymbol{\mathcal{D}_{\text{val}}}$. Модель обучалась 30 эпох с методом оптимизации ADAM [42], начальным параметром коэффициента обучения $lr = 0.0001$, убывающим каждые 10 эпох в 2 раза. Качество алгоритма проверялось на выборке $\boldsymbol{\mathcal{D}_{\text{test}}}$, распределение которой сильно отличалось от данных в обучении. Критерием качества модели считался TPR в точке $FPR = 0.01$.

После обучения первой версии модели был произведен сбор сложных примеров. Для этого в telegram бота по сбору данных был добавлен ответ от системы пользователю, показывающий число s – вероятность живости – для загруженного видео. Пользователей просили снять настоящие и поддельные видео которые покажут наиболее неправильный результат с точки зрения оценки живости. Было получено около 100 реальных и 100 фальсифицированных видео, которые плохо распознавались системой. Данные видео были разбиты на кадры и добавлены в обучающую выборку. Итоговый результат работы новой модели на тестовой выборке отображен в таблице 2.2. Результаты посчитаны для разных типов атак чтобы показать степень защиты модели для каждого случая индивидуально.

Таблица 2.2. Результат работы предложенного алгоритма для разных типов атак.

Тип атак	Количество	TPR в FPR = 0.01
P1	63	0.994
P2	34	0.969

`

P3	99	0.944
все P	196	0.944
D1	63	0.982
D4	59	0.000

Для сравнения эффективности предложенного метода, рассматривается алгоритм, обученный по исходным изображениям с помощью процедуры построения оптимального алгоритма. Предложенный алгоритм по искусственной модальности оптического потока показывает значительное улучшение по сравнению с базовым алгоритмом на обычных изображениях (рис. 2.10.).

Предложенный метод оценки живости по оптическому потоку хорошо справляется с распечатанными и неподвижными артефактами, но при этом не работает для динамического случая D4, в котором присутствовали видео с запрашиваемыми движениями головой. Дальнейшее улучшение метода возможно добавлением к текущему кооперативному алгоритму некооперативного модуля, нацеленного на выявление экранных атак.

Другим возможным решением является изменение кооперативного поведения на более сложное движение. Видео человека, кивающего головой получить сложно, но возможно. Но метод по оптическому потоку подходит и для другого действия – приближения.

`

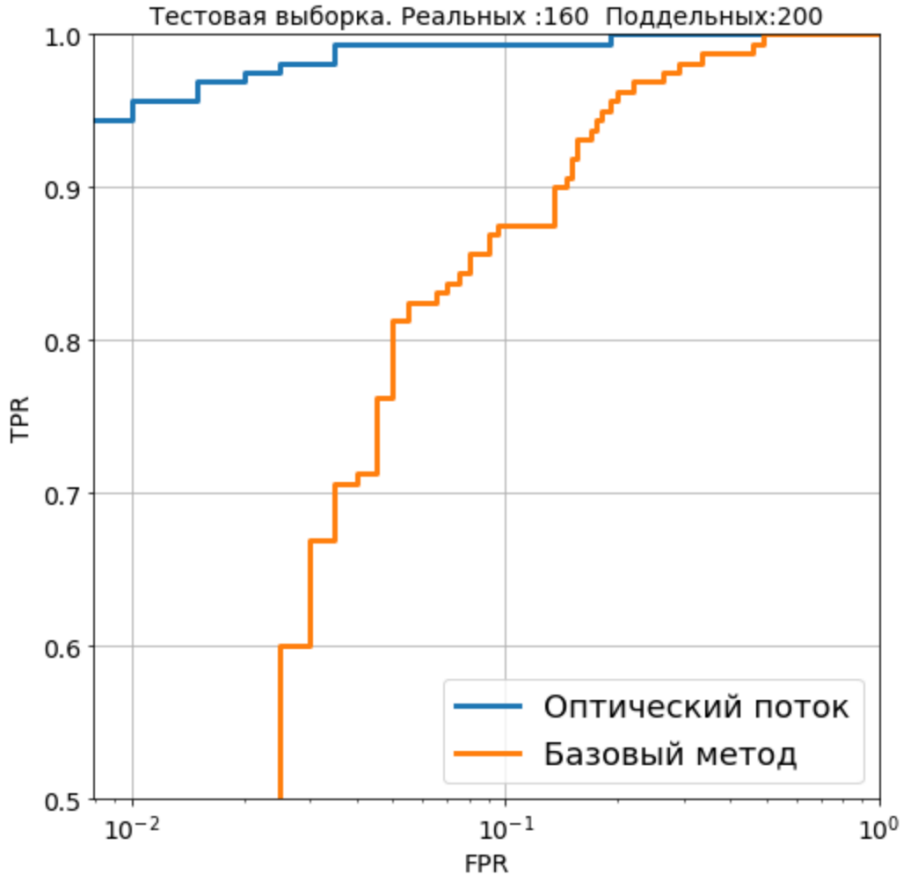

Рис 2.10. Сравнение меры качества алгоритмов, обученных на модальности оптического потока и на исходных изображениях.

Пользователь начинает съемку видео на расстоянии вытянутой руки, после чего в течение нескольких секунд приближает телефон, пока изображение лица не займет весь экран. В таком случае ближнее изображение будет обладать эффектом рыбьего глаза [5] – искажения, когда нос увеличен по сравнению с обычным лицом. Биометрический шаблон такого типа получить намного сложнее, что на практике делает метод устойчивым к динамическим атакам. Но данное кооперативное движение также, как и атомарный алгоритм оценки подлинности является неудобным для пользователя, поэтому на практике широкого спроса алгоритм по приближению лица не получил.

`

2.4. Заключение

В данной главе были освещены основные методы кооперативного оценки живости, предложен простой атомарный метод определения подлинности, обеспечивающий защиту от атак любых уровней сложности, а также предложен алгоритм оценки по оптическому потоку. Алгоритм определения живости по оптическому потоку требует намного меньше времени и усилий со стороны пользователя, но при этом неустойчив к динамическим атакам.

Ввиду стремительного развития области и увеличенного спроса на более простые для пользователя методы, кооперативные алгоритмы стремительно уходят в прошлое, но все еще являются основными методами защиты для большинства систем биометрической идентификации.

Глава 3

Некооперативные методы определения живости для СКУД

`

Проблема определения живости для сценария СКУД стоит обособленно от других методов из-за сильного различия в сценарии. В данном случае, камеры установлены на пропускающих терминалах и система должна оценить подлинность пользователя до того, как он подойдет на расстояние прохода через турникет. Соответственно, основная работа ведется в диапазоне 1.5-3м от камеры, при этом размер лица в таком случае составляет 80-120 пикселей. Сценарий предполагает некооперативное поведение пользователя, то есть живость должна быть определена по треку, записанному в диапазоне подхода человека к турникету.

Положительным моментом является частичная защищенность от экранных атак и полноразмерных распечатанных артефактов, так как ввиду непрерывности записи с камеры границы артефакта будут видны на ранних кадрах трека. А сильное приближение подделки к окуляру камеры сделает изображение размытым, так как фокус камеры настроен на дальнюю съемку. Самые распространенные примеры атак на систему контроля и управления доступом показаны на рис. 3.1.

В данной главе предлагаются три различных метода решения задачи определения живости для СКУД. Все предложенные методы работают в режиме реального времени и были внедрены в системы контроля доступа различных предприятий в России и за рубежом.

Дана контрольная выборка $\boldsymbol{\mathcal{D}_{\text{test}}}$:

$$\boldsymbol{\mathcal{D}_{\text{test}}} \sim p(T|\boldsymbol{\theta_{\text{test}}}, S, I, M)p(T|\{\boldsymbol{PA}\}),$$

собранная на проходных большого предприятия. Данные сняты в неизвестных условиях $\boldsymbol{\theta_{\text{test}}}$ с нескольких десятков камер. Виды атак (табл. 3.1):

$$\{\boldsymbol{PA}\} = \{P1, P2, P3, D1\},$$

`

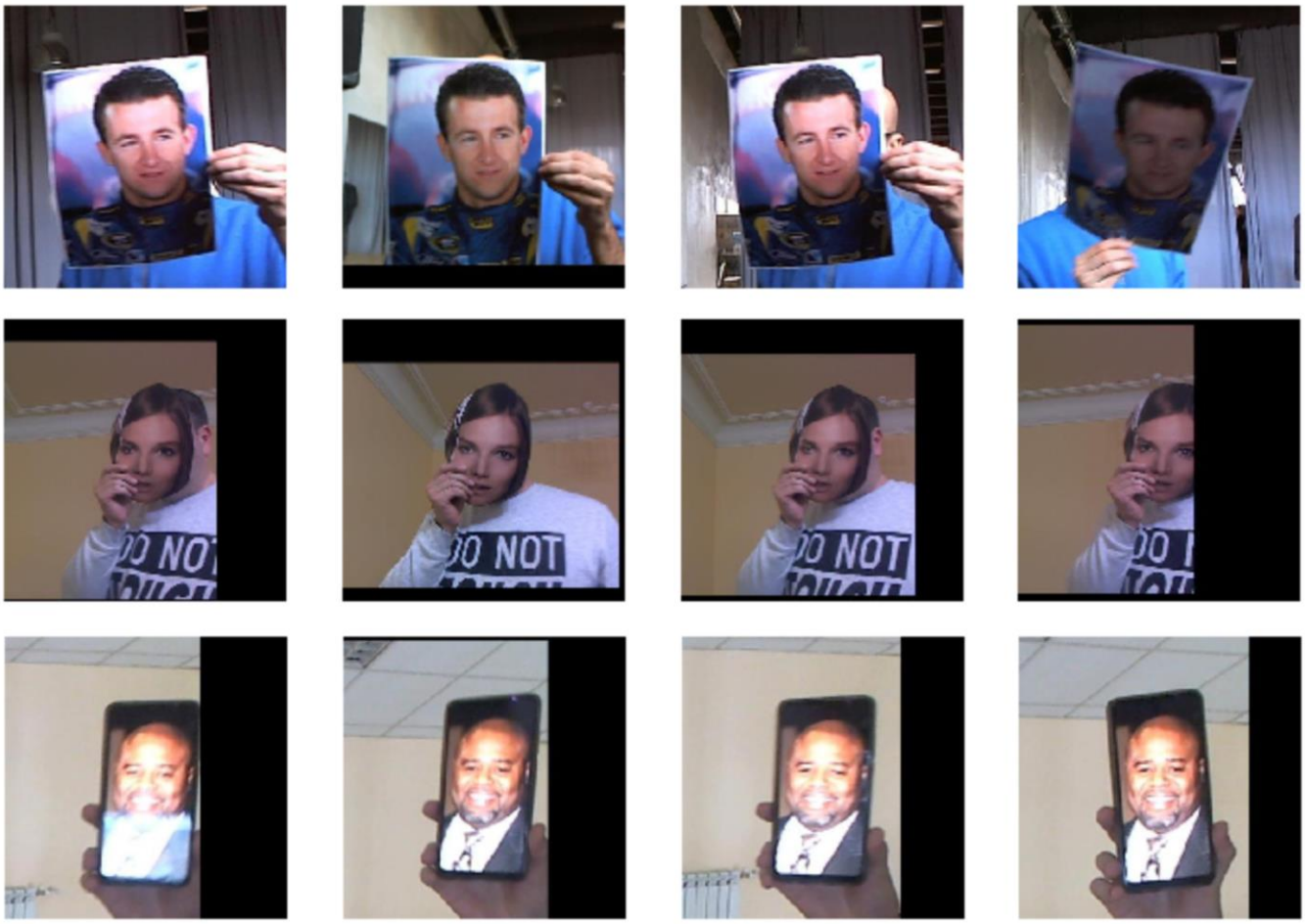

Рис. 3.1. Примеры атак на СКУД (A4, маски и экранные артефакты).

т.е. распечатанные листы A4, полноразмерные распечатки, вырезанные двумерные маски и демонстрация экрана телефонов. Размер выборки – 5097реальных треков и 1425 треков с атаками.

Таблица 3.1. Статистика тестовой выборки для сценария СКУД.

Тип данных	Описание	Количество
Real	Реальный человек.	5097

`

P1	Распечатанное изображение (например, на A4) без модификаций. Края бумаги видны в кадре.	394
P2	Распечатанное изображение (например, на A4) без модификаций. Края бумаги не видны в кадре.	200
P3	Распечатанное изображение лица, вырезанное по контуру.	425
D1	Статическое изображение лица (как из социальных сетей), на экране телефона. Границы телефона находятся в кадре.	406

Требуется построить алгоритм, показывающий максимальное значение TPR в точке FPR=0.01 на заданной контрольной выборке.

3.1. Сбор обучающей выборки

При создании обучающей выборки для решения поставленной задачи, внутренние условия контрольной выборки $\boldsymbol{\theta_{test}}$ повторить невозможно, но можно попробовать покрыть максимальное число внутренних условий за счет сбора данных при постоянно меняющимся заднем плане, как имитация разных турникетов.

Чтобы сделать обучающую выборку максимально разнообразной по заднему фону, была снята профессиональная студия, где 10 человек в течение двух часов имитировали попытки взлома и реальные проходки перед 5 установленными на разной высоте камерами. На рис. 3.2. показаны примеры получившихся треков.

Всего было собрано 1073 трека с различными видами атак, средняя длина трека – 40 кадров, что соответствует 1.5 секундам видеозаписи.

`

Рис. 3.2. Собранные в студии треки разных видов атак.

Помимо полученных данных из студии, на проходных турникетах офиса была установлена камера, снимающая входящий поток людей, добавив к обучающей выборке 2300 реальных треков. Кроме этого, было собрано 2000 треков с видеозаписей youtube, где люди ходят по улицам и снимают прохожих. Было скачано 200 роликов, собранных в разных городах и странах. Данные очень разнообразны, что увеличивает вариативность выборки. Примеры реальных записей с канала показаны на рис 3.3.

Все собранные изображения были пропущены через детектор лиц, кадры были центрированы и была составлена обучающая выборка $\boldsymbol{\mathcal{D}}$.

`

Рис. 3.3. Собранные в Youtube реальные треки прохожих.

3.2. Живость по одному изображению

С учетом собранных данных, построим алгоритм определения живости по одному кадру $\mathbf{f}_X(X_i, \mathbf{w})$. Результат на треках будем считать, как усреднение результатов по кадрам:

$$\mathbf{f}(T, \mathbf{w}) = \boldsymbol{\psi}_{\mathbf{avg}}(\{\mathbf{f}_X(X_i, \mathbf{w})\}).$$

Будем обучать алгоритм по предложенной процедуре. Сначала сгенерируем новые данные, заменяя фон исходных изображений на произвольный. Изображение одного реального человека и его маска сегментации позволяют сгенерировать множество различных фальсификаций (можно менять изображение, на которое переносится маска, варьировать размер маски и ее положение на изображении), что делает выборку богатой и разнообразной, позволяя избежать переобучения.

`

Для еще большего увеличения вариативности предлагается использовать сильную аугментацию, такую как повороты, размытие и цветовые возмущения. Кроме этого, маску сегментации можно растягивать до квадрата, имитируя распечатанные А4, или вырезать не только лицо, но и большую прямоугольную область вокруг него, имитируя атаки с помощью экранов устройств.

Разобьем собранную выборку на обучающую и валидационную по разным внутренним условиям – **id** – исключая переобучение алгоритма на черты лица человека:

$$\boldsymbol{\mathcal{D}} = \boldsymbol{\mathcal{D}}_{\mathbf{train}} \cup \boldsymbol{\mathcal{D}}_{\mathbf{val}} \sim p(T|\mathbf{id_1})p(T|\{\boldsymbol{PA}\}) + p(x|\mathbf{id_2})p(T|\{\boldsymbol{PA}\})$$

Далее, выберем семейство моделей исходя из ограничений по времени работы. Так как итоговый алгоритм должен работать в режиме реального времени, была выбрана авторская архитектура SimpleNet (рис 3.4). Алгоритм на такой архитектуре выполняется 5-6 мс на одном ядре обычного процессора (CPU). Рассмотрим детали обучения.

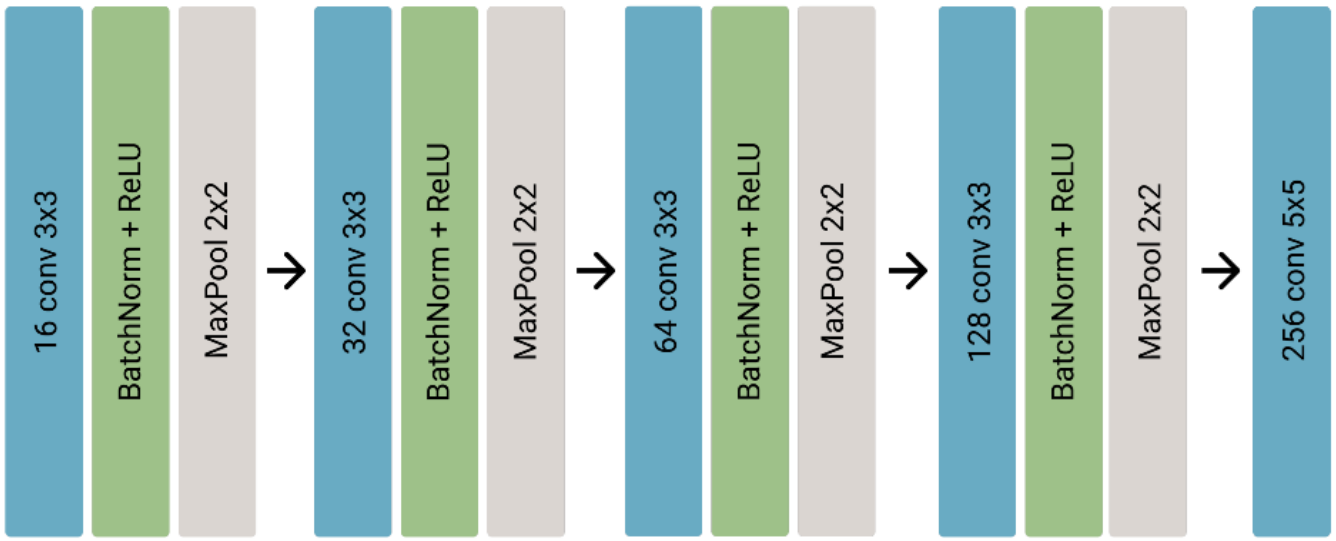

Рис. 3.4. Архитектура SimpleNet.

`

3.2.1. Обучение модели

Для обучения метода определения живости по одному кадру была выбрана архитектура (рис 3.4.). Такая архитектура позволяет использовать алгоритм в режиме реального времени. Кроме скорости, низкая мощность сети позволяет избежать переобучения на детали и концентрироваться на поиске границ между демонстрируемым на камеру распечатанным артефактом и фоном. Глубокая архитектура может выучить мелкие особенности цветных изображений и не будет обобщаться на реальные примеры.

Нейронная сеть принимает на вход центрированное по лицу изображение размера 224 пикселя и возвращает предсказание живости. Модель обучалась 250 эпох на 280 000 реальных данных с помощью ADAM оптимизатора [42] с коэффициентом обучения, изменяемым по косинусу и начальным значением 0.001.

3.2.2. Эксперименты

Были проведены эксперименты по разному соотношению исходных выборок и степени аугментации при генерировании подделок, лучший результат показан в таблице 3.2. по выборке СКУД, описанной в предыдущем разделе.

Таблица 3.2. Точность работы алгоритма оценки живости по одному изображению.

Вид атаки	Кадры, TPR в точке FPR =			Треки, TPR в точке FPR =		
	0.1	**0.01**	**0.001**	**0.1**	**0.01**	**0.001**
P1	0.999	0.977	0.709	1.000	0.991	0.450
P2	0.688	0.134	0.000	0.595	0.116	0.031
P3	0.998	0.962	0.810	1.000	0.994	0.046

`

D1	0.995	0.525	0.000	0.935	0.424	0.017
Все	0.995	0.603	0.000	0.977	0.416	0.031

Предложенный алгоритм хорошо справляется с вырезанными масками P3 – 0.994 TPR в точке FPR=0.01, но плохо работает на полноразмерных артефактах P2 – его точность всего 0.116 в данном домене. Это ожидаемо, так как на таких изображениях отсутствуют границы артефактов.

Несмотря на то, что выборка довольно большая, потенциал модели не был исчерпан. Для этого был проведен эксперимент по построению зависимости значения функционала качества – функции потерь L на тестовой выборке – от размера обучающей выборки.

Из исходной обучающей выборки в 280000 изображений было выбрано по 5 случайных подвыборок для заданных размеров: 10000, 20000, 40000, 70000, 14000 изображений. Далее, по описанной выше процедуре строился оптимальный алгоритм на архитектуре SimpleNet и считалось значение функции потерь на котрольной выборке.

На рис. 3.5. показана зависимость качества алгоритма от размера обучающей выборки. Видно, что график строго убывающий и замедления падения не наблюдается, то есть, если бы было больше разнообразных данных для обучения, точность алгоритма была бы выше.

Для улучшения точности на контрольной выборке предлагается использовать переход в искусственную модальность.

`

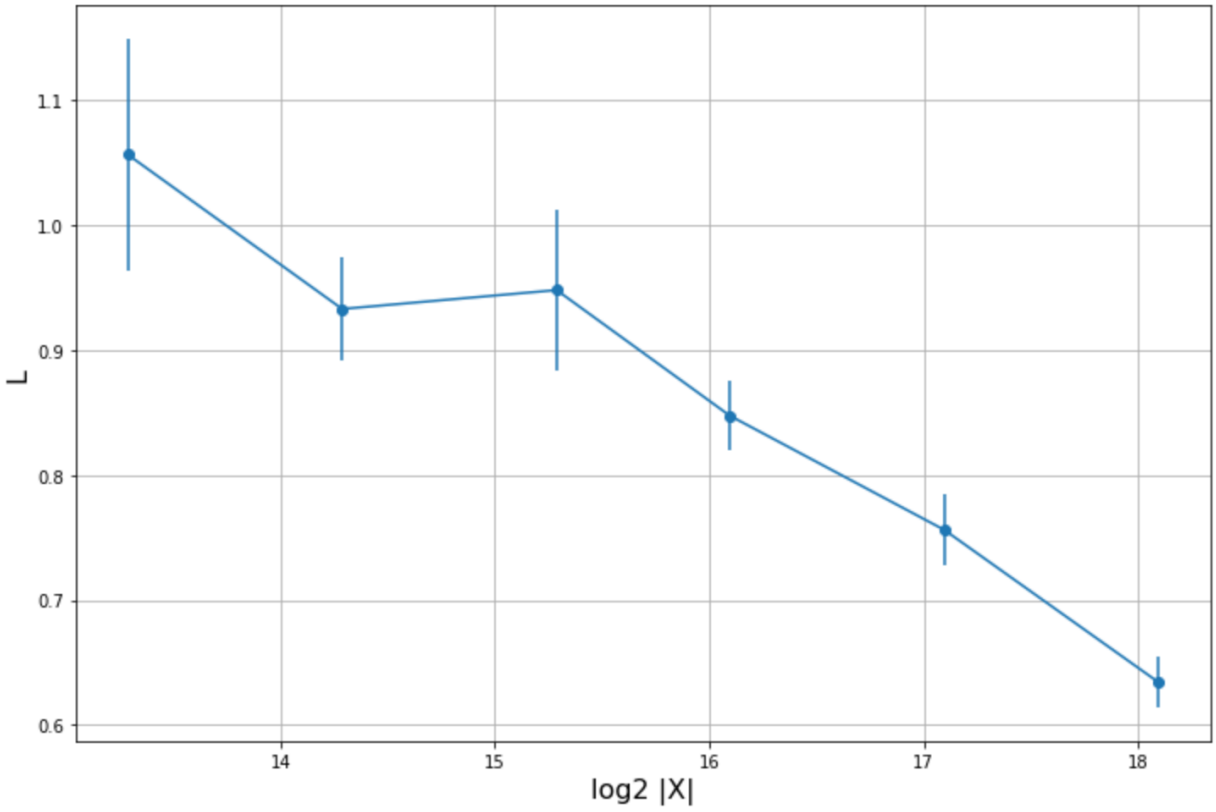

Рис. 3.5. Зависимость значения функции потерь на контрольной выборке от размера обучающей выборки.

3.3. Живость по границам изображения

Идея метода основана на предположении, что силуэты реальных пользователей и злоумышленников с артефактами отличаются. Когда взломщик держит артефакт перед собой, контур артефакта чаще всего вылезает за пределы контура человека. Этот признак можно выделить методом компьютерного зрения – фильтром поиска границ Canny [71], а потом построить классификатор, направленный на поиск отличий между силуэтами реальных людей и злоумышленников.

`

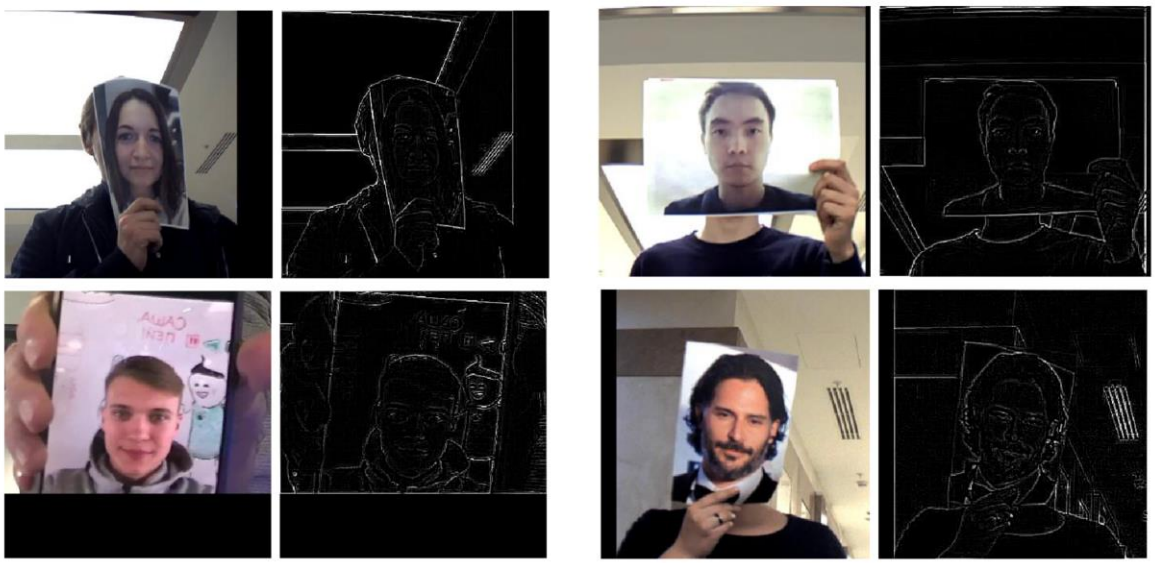

Рис. 3.6. Пример работы алгоритма выделения границ на поддельных изображениях.

На рис 3.6 и 3.7. показаны примеры работы алгоритма выделения границ для поддельных и реальных изображений. Для поддельных четко видны контуры артефактов, в то время как силуэты реальных людей выглядят иначе.

Карту контуров изображения можно рассматривать как искусственную модальность, аналогичную карте оптического потока. Новое отображение RGB картинок снижает сложность конечной задачи определения живости так как содержит меньше степеней свободы (потеряна информация о чертах лица человека), что позволяет обучить модель определения подлинности на меньшем числе данных без риска переобучения. Более того, новая модальность была выбрана так, чтобы уже содержать полезные для живости признаки: различие силуэтов добросовестных и злоумышленников.

`

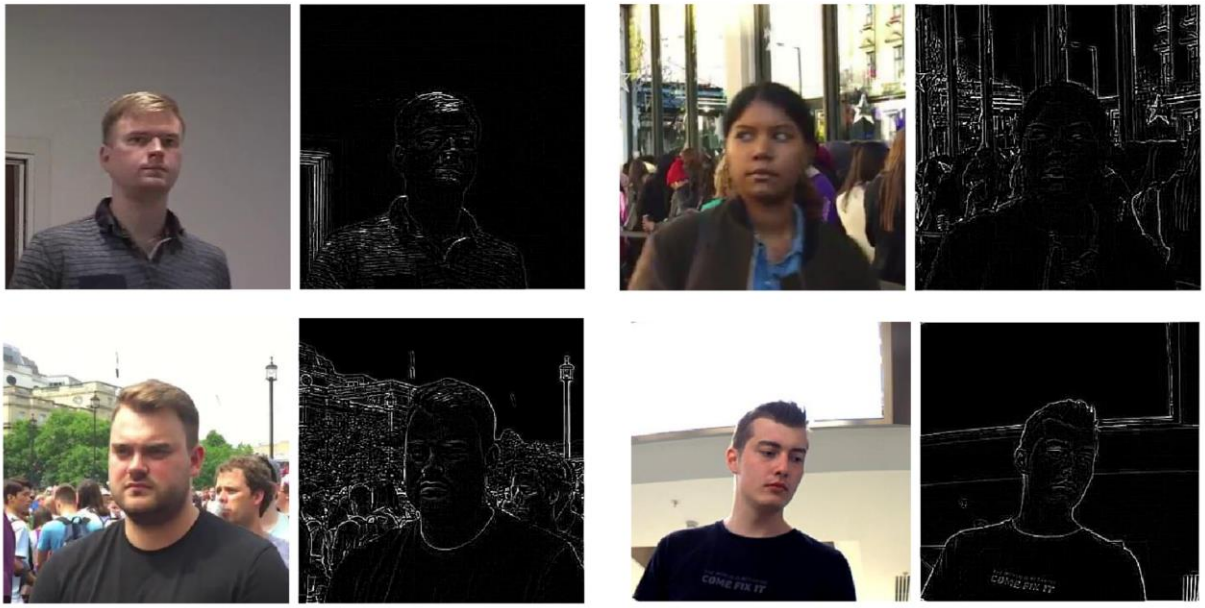

Рис. 3.7. Пример работы алгоритма выделения границ на реальных изображениях.

Построим алгоритм определения живости по карте границ изображения $\mathbf{f}_X(\phi(X_i), \mathbf{w})$, где ϕ – функция преобразования искусственной модальности. Результат на треках будем считать, как усреднение результатов по кадрам:

$$\mathbf{f}(T, \mathbf{w}) = \boldsymbol{\psi}_{\mathbf{avg}}(\{\mathbf{f}_X(\phi(X_i), \mathbf{w})\}).$$

Обучение алгоритма по картам границ аналогично обучению по обычному изображению. Семейство моделей выбирается исходя из ограничений по времени работы – рассматривается SimpleNet. Далее, обучаются модели и выбирается одна оптимальная по значению функции потерь на валидационной выборке

3.3.1. Обучение модели

Ввиду того, что карта границ содержит мало детальной информации, для обучения алгоритма определения живости по таким данным достаточно неглубокой

`

нейронной сети. Поэтому в данном случае использовалась архитектура SimpleNet (рис. 3.4).

Карта контуров имеет размерность $1 \times 112 \times 112$, на выходе модели применяется сигмоидная функция активации, функция потерь – бинарная кросс-энтропия. Структура обучения аналогична описанной в разделе определения живости по оптическому потоку, лучшая эпоха выбиралась по валидационной выборке, составляющей 10% от $\boldsymbol{\mathcal{D}}_{\mathbf{train}}$.

Предложенный некооперативный метод определения подлинности работает по одному кадру, никак не учитывая поведение на соседних кадрах. Однако, с учетом скорости работы алгоритма и наличием нескольких кадров в треке на этапе тестирования, возможно улучшить точность работы, агрегируя результаты по треку. Наиболее надежная и показывающая лучший результат стратегия – усреднение предсказаний модели по всем кадрам трека. В таблице 3.3. показаны результаты работы итоговой модели определения живости по картам сегментации отдельно для кадров и для треков, а также разбитые по типам атак.

3.3.2. Эксперименты

Таблица 3.3. Точность работы алгоритма определения живости по маске границ.

Вид атаки	Кадры, TPR в точке FPR =			Треки, TPR в точке FPR =		
	0.1	**0.01**	**0.001**	**0.1**	**0.01**	**0.001**
P1	0.926	0.691	0.000	0.856	0.565	0.182
P2	0.827	0.563	0.000	0.727	0.507	0.317
P3	0.925	0.746	0.563	0.888	0.666	0.000

`

D1	0.989	0.867	0.640	0.995	0.839	0.691
Все	0.929	0.722	0.379	0.869	0.578	0.182

Точность алгоритма на типах подделок P2 ожидаемо ниже остальных срезов, так как данный вид атак содержит полноразмерные подделки и границы реальных и поддельных изображений выглядят одинаково, что усложняет работу модели. Для полной тестовой выборки, усреднение предсказаний модели по треку повышает точность модели с 0.416 до 0.578 % TPR в FPR=0.01, т.е. при допустимой ложноположительной ошибке 1% система не пропустит 42.2% реальных пользователей. Однако, точность текущего алгоритма сложно повысить ввиду ограничений используемых промежуточных модальностей. Поэтому для улучшения качества предлагается еще один алгоритм оценки живости, дополняющий слабые места вышеописанного метода.

Новый метод работает лучше предыдущего на всей выборке, но так как они оба быстрые и противодействуют разным типам атак, можно использовать ансамбль моделей, просто покадрово усредняя предсказания обеих алгоритмов. В таблице 3.4 показана точность объединенных алгоритмов по маскам границ и по одному кадру

Таблица 3.4. Точность работы объединенной модели определения живости.

Вид атаки	Кадры, TPR в точке FPR =			Треки, TPR в точке FPR =		
	0.1	**0.01**	**0.001**	**0.1**	**0.01**	**0.001**
P1	0.998	0.942	0.759	0.999	0.935	0.240
P2	0.864	0.630	0.307	0.783	0.560	0.494
P3	0.996	0.933	0.771	0.999	0.959	0.012

`

D1	0.998	0.936	0.795	0.998	0.855	0.617
Все	0.990	0.850	0.619	0.989	0.753	0.240

Объединенная модель показывает значительно лучшие результаты по сравнению с компонентами по отдельности. Точность работы на каждом из видов атак колеблется от 0.560 TPR в FPR=0.01 для P2 до 0.935 для P1, а общая точность на всей выборке составляет 0.753 TPR в FPR=0.01, т.е. добавление нового алгоритма снизило ложно-отрицательную до 24.7%.

3.4. Живость по динамике трека.

Алгоритмы оценки живости по синтетическим лицам и картам сегментации работают только по одному кадру и не зависят от результатов на соседних, а агрегация по треку является усреднением независимых оценок каждого кадра. Однако, сценарий СКУД предполагает, что для обработки доступен весь трек, то есть можно оценивать не только статические кадры, но и *динамические временные* признаки, которые отличаются у живого человека и подделки.

Использование динамической информации было описано в главе 2. В частности, алгоритм оптического потока позволяет построить грубую карту глубины лица, если оно двигается. В кооперативном методе система просит пользователя подвинуть голову на определенный градус, в сценарии СКУД такой возможности нет. Но, на практике очень часто человек двигает голову или его мимика меняется за несколько секунд, которые он подходит к турникету.

Для проверки потенциала динамических признаков в рассматриваемом сценарии, алгоритм оценки живости по оптическому потоку был применен к тестовой выборке СКУД. Для каждого кадра из трека выбирался наиболее далекий по углам головы кадр

`

из того же трека, т.е. каждому кадру трека была поставлена в соответствие пара кадров, что дало возможность посчитать оптический поток и соответствующий скор liveness.

Результаты работы метода показаны в таблице 3.5. Алгоритм оптического потока отлично подходит для некоопертивного сценария, в котором длина трека достаточно длинная.

Таблица 3.5. Точность работы оценки подлинности по оптическому потоку на выборке СКУД.

Вид атаки	Кадры, TPR в точке FPR =			Треки, TPR в точке FPR =		
	0.1	**0.01**	**0.001**	**0.1**	**0.01**	**0.001**
P1	0.970	0.890	0.755	0.996	0.961	0.679
P2	0.958	0.857	0.761	0.995	0.977	0.966
P3	0.946	0.822	0.000	0.987	0.939	0.792
D1	0.981	0.909	0.800	0.999	0.977	0.840
Все	0.962	0.859	0.687	0.994	0.962	0.792

Метод работает стабильно для всех видов атак, а его итоговая точность практически не уступает объединенному решению двух предыдущих алгоритмов. Но ввиду очень медленной скорости работы (~4 с. на один трек) и большого размера файла, содержащего веса нейронной сети, данный метод неприменим на практике. Однако, проведенный эксперимент доказывает эффективность динамических методов для задачи СКУД и перспективность работы в этом направлении.

`

В данном разделе предлагается быстрый алгоритм оценки живости по динамике трека, который можно использовать в прикладных задачах. Идея метода основана на изменяемости лица пользователя в процессе подхода к турникету. При поднесении статического артефакта, его мимика и поворот головы не будет меняться, в отличие от реального человека (рис. 3.8). Соответственно, можно обучить модель, принимающую на вход весь трек и выдающую единственное число – оценку живости всего трека.

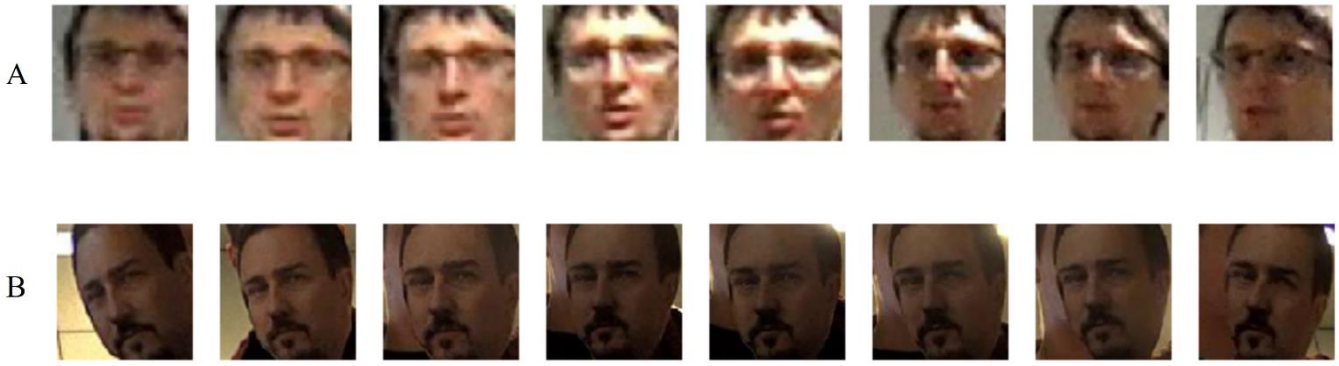

Рис. 3.8. Пример реального (A) и поддельного (B) треков в сценарии СКУД.

3.4.1. Описание алгоритма

Пусть дан трек $T = \{X_i\},\ i = 1, \dots, N$, на котором непрерывно присутствует лицо одного человека и соответствующая оценка живости l. Из трека выбирается L=8 кадров, распределенных равномерно, т.е. каждый $[\frac{N}{L}]$-й кадр, где [] – оператор выбора целой части. Полученная последовательность подается на вход нейронной сети, описанной на рис. 3.11.

Из каждого кадра с помощью детектора лиц выбирается лицо и приводится к размеру 112×112. Каждый кадр подается на вход в SimpleNet (рис. 3.4) и переводится

`

в дескриптор размера $1 \times d$, где $d = 256$. Дескрипторы конкатенируются в один тензор $8 \times d$, который обрабатывается пространственной сверткой 8×1, после чего итоговый дескриптор трека проходит через полносвязный слой с сигмоидной функцией активации, возвращая оценку живости (рис. 3.9). Модель обучается с помощью стандартной бинарной кросс-энтропии, при этом веса сети SimpleNet являются общими для всех 8 кадров.

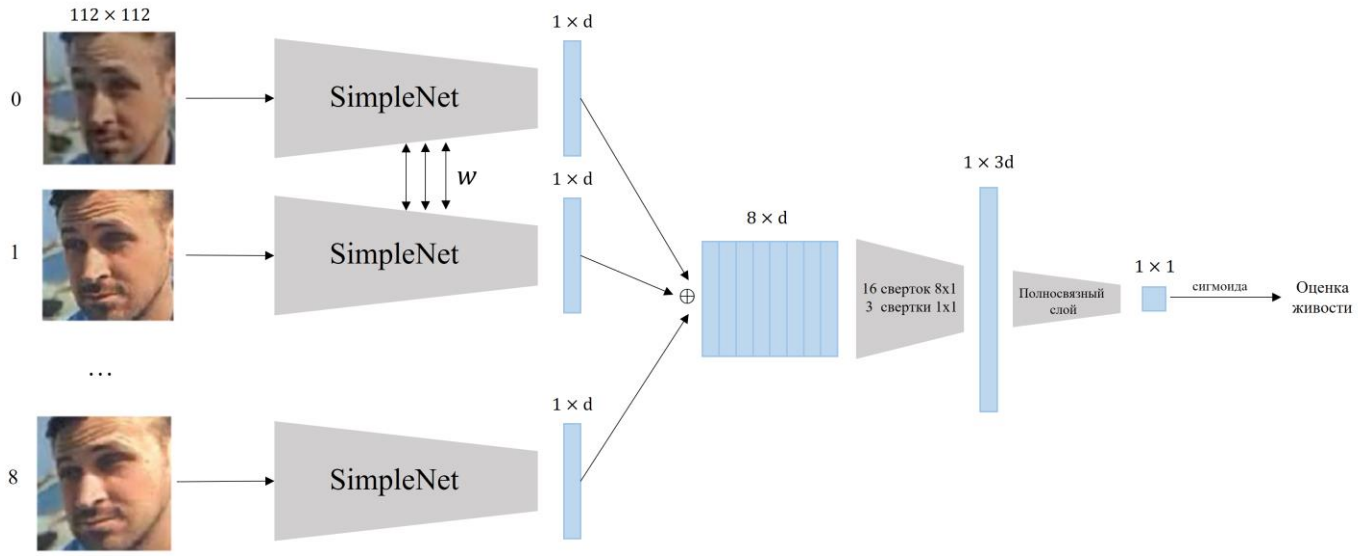

Рис. 3.9. Архитектура нейронной сети определения живости по динамике трека. $\oplus$ – оператор конкатенации.

Идея алгоритма состоит в том, что базовая сеть SimpleNet учит дескриптор, совпадающий для лиц с одинаковой мимикой/поворотом головы и различный для лиц с изменением мимики. Это свойство потом ловится сверточными слоями, которые смотрят на дескрипторы всех кадров одновременно и в конце обрабатывается полносвязным для перевода в одно число. В качестве базовой сети выбрана очень легкая архитектура SimpleNet, что позволяет не только сделать алгоритм быстрым (20 мс на CPU для всего трека), но и не дает сети переобучиться на сложные признаки.

`

3.4.2. Эксперименты

Работая с обычными изображениями, выбор легких архитектур позволяет избежать переобучения на определенный домен, так как данных для обучения обычно немного. В данном случае для обучения использовались те же треки, что и в предыдущих двух алгоритмах, но так как поддельных примеров недостаточно для обучения нейросети, дополнительные поддельные треки генерировались синтетически.

Для этого выбирался случайный кадр из реального трека и дублировался 8 раз, имитируя неподвижность мимики. После чего каждое из 8 одинаковых изображений лиц случайно аугментировалось поворотом, пространственным сдвигом, цветовой коррекцией и размытием, имитируя движения атакующего артефакта. Это позволило значительно расширить обучающую выборку для поддельного класса и повысить итоговую точность на тестовом сете.

Таблица 3.6. Точность работы оценки живости по динамике трека на выборке СКУД.

Вид атаки	Треки, TPR в точке FPR =		
	0.1	**0.01**	**0.001**
P1	0.944	0.648	0.173
P2	0.982	0.860	0.606
P3	0.976	0.829	0.663
D1	0.997	0.886	0.817
Все	0.979	0.811	0.285

`

Результаты обучения модели показаны в таблице 3.6. Так как нейронная сеть обрабатывает весь трек целиком, то результат по кадрам отдельно посчитать невозможно. Предложенный метод работает лучше, чем алгоритм по обычным изображениям в низких FPR, особенно для экранных и полноразмерных атак, где манипуляции с мимикой/поворотами лица незначительны.

3.5. Заключение

В данной главе были предложены три алгоритма определения подлинности изображения лиц для сценария СКУД, где автоматическая система оценивает живость человека, подходящего к турникету, по записанной видеопоследовательности. Была собрана репрезентативная тестовая выборка в реальных условиях, состоящая из 1500 треков атак и 5000 треков реальных людей. Атаки были разделены на четыре вида: распечатанные лица на А4, края бумаги видны в кадре; распечатанные полноразмерные портреты, края бумаги не видны в кадре; вырезанные по контуру лица маски; изображения с экрана телефона. Для обучения модели была собрана выборка из 1000 треков различных атак. Помимо этого, было собрано около 2000 треков реальных людей с сайта youtube и 2300 с турникета офисного здания.

Идея предложенного алгоритма определения подлинности по картам границ состояла в различии контуров поддельного и реального изображений. Для реальных примеров эти контуры антропоморфны, для атак – отличаться от реальных. Сначала изображения переводились в соответствующие карты границ, после чего обрабатывались неглубокой нейронной сетью. Так как при переходе в новую модальность эффективное признаковое пространство значительно уменьшается, собранных данных было достаточно для создания устойчивой модели. Алгоритм хорошо работает на А4, полноразмерных и экранных атаках, так как именно в них отличия в контурах значительны.

Идея предложенного алгоритма определения подлинности по динамике трека заключалась в предположении, что мимика и углы поворота головы реального человека

`

меняются в процессе подхода к турникету, в то время как статические артефакты остаются неизменными. Для увеличения вариативности обучающей выборки был реализован процесс генерации подделок из реальных треков путем дублирования одного кадра и применения аугментации. Кроме этого, была предложена архитектура нейронной сети, обрабатывающая весь трек целиком и содержащая временную агрегацию внутри себя. Алгоритм хорошо работает на всех видах атак в низких FPR, проседая в высоких FPR (т.е. в тестовой выборке присутствовали треки реальных людей, у которых мимика менялась очень незначительно и алгоритм принимал их за подделку).

Таблица 3.7. Точность работы ансамбля из трех алгоритмов определения живости для сценария СКУД.

Вид атаки	Треки, TPR в точке FPR =		
	0.1	**0.01**	**0.001**
P1	1.000	0.975	0.882
P2	0.975	0.912	0.572
P3	1.000	0.993	0.641
D1	1.000	0.977	0.770
Все	0.999	0.948	0.641

В таблице 3.7 показана точность ансамбля из трех предложенных алгоритмов на тестовой выборке. TPR=0.948 в точке FPR=0.01, то есть при допустимости 1% ложноположительных проходов будет 6.2% ложноотрицательных, что в 4 раза меньше ошибки при агрегации двух алгоритмов. На рис. 3.10. показаны результаты предложенных алгоритмов на всей тестовой выборке.

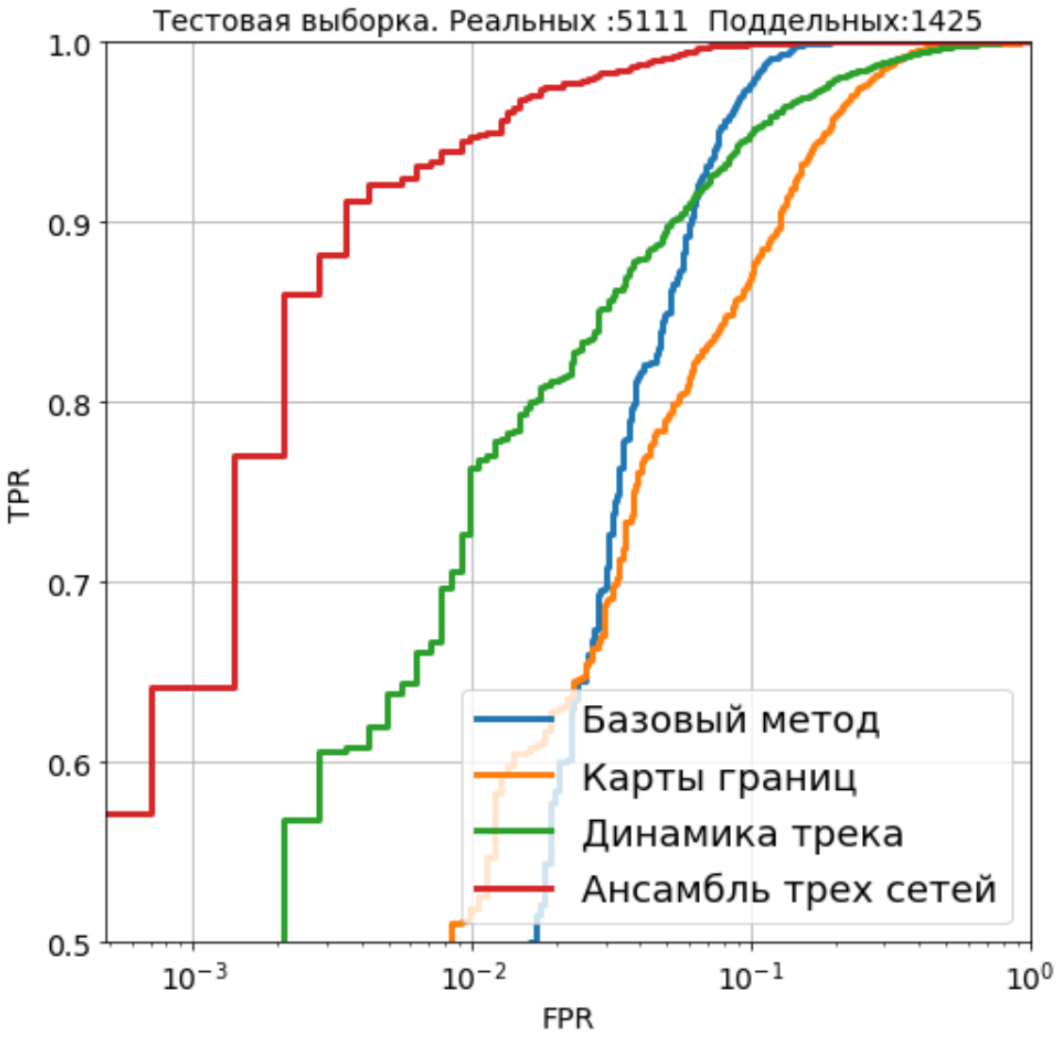

Рис. 3.10. ROC-кривая предложенных алгоритмов на тестовой выборке СКУД.

Итоговая точность модели и скорость работы в реальном времени позволила применять ее в прикладных условиях. На данный момент часть предложенных алгоритмов используется более чем на 1000 турникетах по всему миру.

`

Глава 4

Методы определения живости по мультимодальным данным

Помимо систем контроля и управлением доступом, алгоритм оценки подлинности лица требуется и в других сценариях, таких как Мобильный и ПК или АТМ. Такие сценарии можно назвать условно-кооперативными, так как при авторизации пользователь смотрит в камеру. При этом, в некоторых сценариях возможна установка дополнительных камер, добавляющих к доступным данным модальности ИК и глубины модальности, что облегчает работу алгоритмам оценки живости [18, 21, 22].

Первой открытой исследовательской выборкой большого размера, включающей все три модальности, является CASIA-SURF [25]. Авторы выборки организовали конкурс, приуроченный к конференции CVPR 2019, по достижению максимальной точности на тестовой части выборки. В данной главе предлагается алгоритм определения живости по мультимодальным изображениям, который показал лучший результат среди других алгоритмов на выборке CASIA-SURF.

4.1. Живость по мультимодальным данным

Когда есть возможность установить вместо обычных камер специализированные, например, с сенсорами ИК и глубины, сделать надежный алгоритм определения подлинности лица становится в разы проще, так как дополнительные модальности обеспечивают модель очень информативными признаками [24, 26]. Карта глубины позволяет показывает трехмерную структуру демонстрируемого объекта, тем самым значительно упрощая отсечение двумерных артефактов, а инфракрасный диапазон помогает с трехмерными масками, так как изображение глаз у живых людей в ИК отличается от статических изображений глаз.

`

Большинство выборок для определения живости лиц содержат только изображения в формате RGB [21, 22]. До недавнего времени выборки с другими модальностями были очень ограничены в количестве примеров [23, 24], что увеличивало риск переобучения модели на тренировочную часть. Недавно опубликованная выборка CASIA-SURF [25] на порядок лучше предыдущих как с точки зрения количества данных, так и количества доступных модальностей (RGB, ИК, глубина), что позволяет эффективно применить инструментарий нейронных сетей для решения задачи оценки подлинности.

4.2. Описание выборки

CASIA-SURF [25] включает в себя 21000 видеозаписей 1000 субъектов, для каждого субъекта записано одно реальное видео и шесть поддельных видео, содержащих разные виды атак с лицом этого человека. Видео записаны с помощью камеры Intel RealSense SR300 и имеют три синхронизированных канала: RGB, ИК, глубина. Выборка разделена на обучающую, валидационную и тестовую подвыборки, содержащих 300, 100 и 600 уникальных субъектов соответственно. Из каждого видео выбран каждый десятый кадр, переводя каждый ролик в набор изображений. Кроме того, выборки были также разделены по типам атак, в тестовой выборке присутствуют фальсификации, которых не было в обучающей выборке (рис. 4.1). После публикации выборки авторы статьи запустили соревнование на лучшее решение для тестовой части, сделав доступными 40 000 изображений для обучения и валидации.

Примеры настоящих и поддельных изображений из CASIA-SURF показаны на рис 4.1. Атаки отличаются формой (плоская, согнутая) и вырезанными частями лица (табл. 4.1.) для создания объемности подделке. Атаки, представленные в тестовой части, полностью отличаются от содержащихся в обучающей выборке. В таком разбиении данных для демонстрации высокой точности модель должна обладать обобщающей способностью и избегать переобучения на конкретные виды атак, что являлось большой проблемой в ранее опубликованных выборках.

`

Таблица 4.1. Виды фальсификационных атак из CASIA-SURF.

Поверхность	Глаза	Нос	Рот	Выборка
Плоская	✓			Тест
Согнутая	✓			Тест
Плоская	✓	✓		Тест
Согнутая	✓	✓		Тест
Плоская	✓	✓	✓	Обучение
Согнутая	✓	✓	✓	Обучение

Вместе с публикацией CASIA-SURF [25] авторы также предложили базовый метод решения. Нейронная сеть обрабатывает каждую из модальностей отдельно, используя архитектурные блоки из resnet-18 [37] в качестве основы. Далее совершается перебалансировка признаков каждой ветви, выбираются наиболее информативные признаки и подавляются остальные. Выходы с каждой из трех ветвей объединяются в один и обрабатываются еще двумя resnet-блоками. Завершают архитектуру глобальный слой усреднения и два полносвязных слоя. Авторы провели тщательные эксперименты и показали преимущества предложенной модели.

`

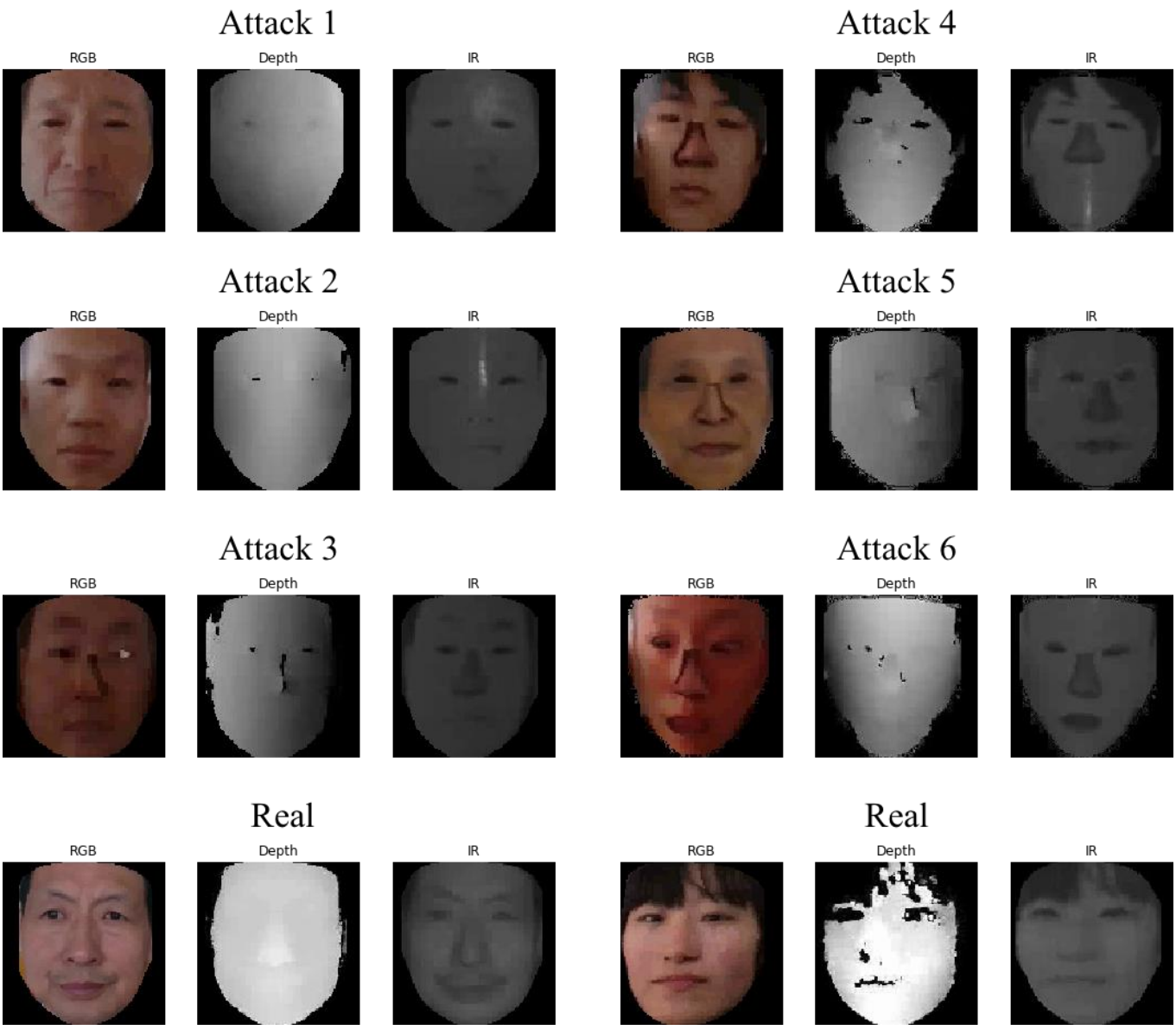

Рис. 4.1. Примеры реальных и поддельных изображений из датасета CASIA-SURF.

4.3. Предлагаемый метод

В CASIA-SURF атаки, представленные в обучающей выборке, отличаются от тестовых атак. Для увеличения устойчивости модели к новым атакам мы выделили из обучающей выборки три части. Каждая часть содержит все изображения двух разных атак, данные по третьей атаке используются как валидационная выборка. После чего

`

обучаются три нейронные сети на каждой из частей. Во время тестирования, все модели рассматриваются как одна, выходы с классификационного слоя усредняются по трем значениям выходов каждой из обученной сети.

Перенос признаков. Множество задач компьютерного зрения [39] с небольшим доступным объемом данных для обучения в качестве инициализации применяют обученные модели других задач, в которых выборка достаточно большая [40]. Дообучение параметров сети, которая была инициализирована предобученными параметрами разных задач, приводит к различным результатам на тестовой выборке. В наших экспериментах мы тестируем четыре разные модели, предобученные на разных датасетах распознавания лиц и классификации пола. Кроме того, в этих задачах используются разные архитектуры базовой модели и функции потерь для увеличения вариативности итоговых параметров. После дообучения на задаче определения живости четыре итоговые модели применяются как одна путем усреднения предсказаний.

4.4. Архитектура модели

Предлагаемая архитектура основана на resnet-34 и resnet-50 с SE-модулями (squeeze and excitation) [37, 77], как показано на рис. 4.2. В базовом алгоритме [25] каждая модальность обрабатывается первыми тремя блоками архитектуры resnet, дальше три ветви объединяются с помощью SE-модуля и обрабатываются оставшимся res-блоком. В отличие от базового метода мы обогатили модель дополнительными блоками агрегации на каждом слое ветвей (*мультиуровневая агрегация признаков* – МУАП). Агрегационный блок берет признаки с соответствующих res-блоков подсетей модальностей и из предыдущего агрегационного блока и обрабатывает их. Такая архитектура позволяет нейронной сети находить корреляцию между признаками не только высокого, но и низкого уровня.

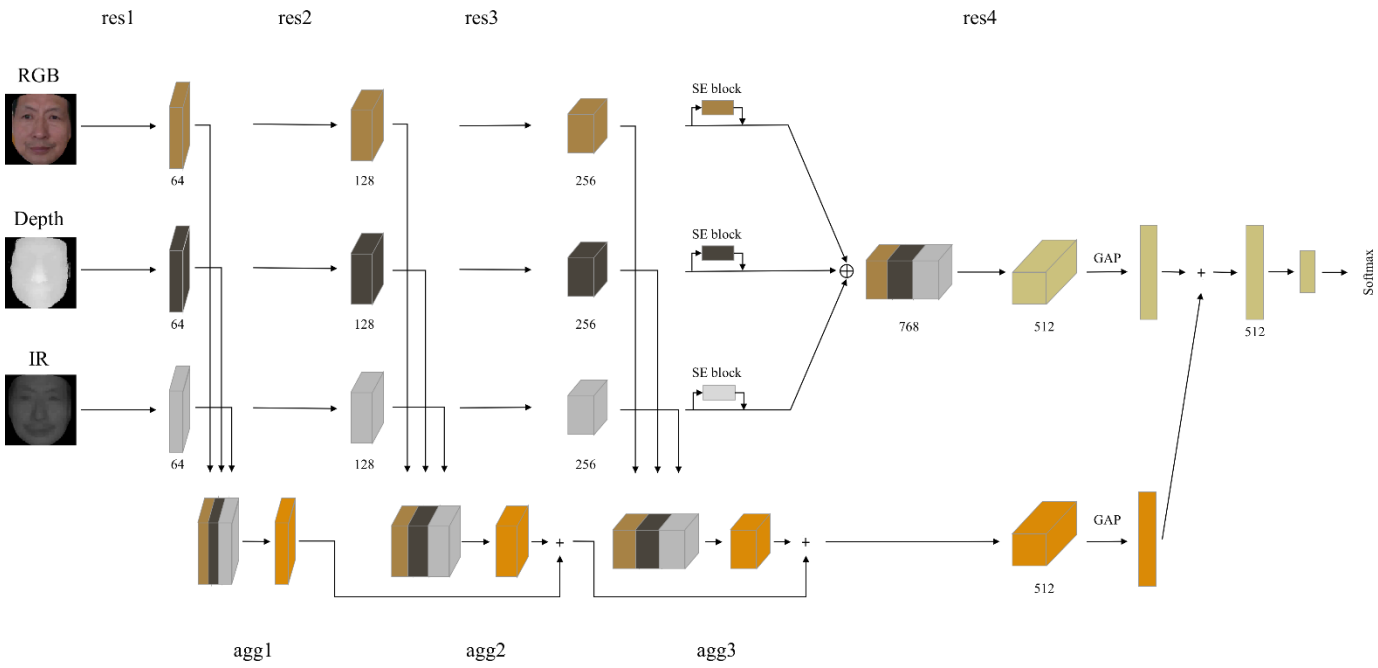

Рис. 4.2. Предлагаемая архитектура. GAP – общий слой усреднения; $\oplus$ – оператор объединения; + – оператор почленного суммирования.

4.5. Эксперименты

Код был написан с помощью библиотеки Pytorch [41], нейронные сети обучались на четырех видеокартах NVIDIA 1080Ti. Обучение одной модели занимает 3 ч, обученная модель извлекает предсказания по 1000 изображениям за 8 с. Все нейронные сети были обучены с помощью оптимизатора ADAM [42], параметр скорости обучения изменялся по косинусу, в качестве функции потерь использовалась стандартная двухклассовая кроссэнтропия. Модель обучалась 30 эпох с начальным параметром скорости обучения 0.1 и размером батча 128. Эти же параметры применялись для обучения моделей распознавания лиц.

Изображения в выборке CASIA-SURF уже вырезаны по контуру лица, поэтому никакого дополнительного выравнивания лиц не потребовалось. Изображения изменялись до 125×125 пикселей, после чего вырезался центральный регион размером 112×112. В процессе обучения картинки случайно отзеркаливались по горизонтали с вероятностью 0.5.

`

Также были проверены другие стратегии предобработки, но они не принесли улучшений по сравнению с описанной.

Базовый метод. Выборка разбита на обучающую, валидационную и тестовую части, но так как в момент соревнования Chalearn LAP тестовая часть была недоступна, далее все результаты приводятся для валидационной части. В первую очередь мы воспроизвели базовый метод из [25], основанный на resnet-18 и обучили пять сетей на пяти частях по стратегии кросс-валидации. Все части были разделены по субъектам, все изображения одного субъекта принадлежали только одной части. Итоговая модель - результат усреднения предсказаний пяти полученных моделей. В табл. 4.2. приведены результаты из статьи базового метода и результаты нашего воспроизведения. Далее, мы расширили основу сети до resnet-34, что сильно увеличило точность на целевой выборке. Ввиду ограничений вычислительных ресурсов мы обучали только модели resnet-34 и resnet-50, не тестируя более глубокие сети.

Таблица 4.2. Результаты на валидационной выборке CASIA-SURF.

Метод	Инициализация	Обучающая выборка	TPR в точке FPR=10^{-4}, %
Zhang, Wang et al. [25]	Нет	Одна часть	56.80
resnet-18	>>	Пять частей по субъектам	60.54
resnet-34	>>	>>	74.55
resnet-34	>>	Три части по атакам	78.89
resnet-34	ImageNet [40]	>>	92.12
resnet-34	CASIA-Web face [43]	>>	99.80

`

A. resnet-34 + МУАП	CASIA-Web face [43]	>>	99.87
B. resnet-50 + МУАП	MSCeleb-1M [19]	>>	99.63
C. resnet-50 + МУАП	Asian dataset [44]	>>	99.33
D. resnet-34 + МУАП	AFAD-lite [45]	>>	98.70
Усреднение A,B,C,D		>>	100.00

Разбиение обучающей выборки. В данном эксперименте сравниваются результаты моделей, обученных стандартным методом кросс-валидации по пяти частям и предложенным методом разбиения обучающей выборки на три части по типам атак. Изображения реальных людей в таком разбиении случайно разделены по этим частям. Несмотря на то, что новая модель получена усреднением трех сетей, а не пяти, которые к тому же обучены на меньшем количестве данных, чем в стандартном разбиении, ее результаты лучше на 4.3% (табл. 4.2.). Это может быть объяснено тем, что разбиение по типам атак в обучающей выборке позволяет лучше адаптироваться к неизвестным примерам фальсификации из целевого сета.

Инициализация весов. В текущем разделе мы исследуем зависимость целевой точности от задания начальных параметров. Параметры каждой из трех ветвей архитектуры инициализируются весами сети, обученной на ImageNet, после чего дообучаются на CASIA-SURF. В сравнении со случайной инициализацией, применение предобученной сети увеличивает результат с 78.89 до 92.12%. Если же вместо общей выборки классификации изображений ImageNet использовать выборку для задачи распознавания лиц CASIA-Web [43], точность достигает почти идеального значения 99.80%.

`

МУАП. При дополнении стандартной архитектуры предложенным блоком МУАП новая модель после обучения показывает уменьшение ошибки в 1.5 раза по сравнению с базовой моделью (табл. 4.2.).

Ансамбль моделей. Для улучшения устойчивости решения используются четыре модели, предобученные на четырех различных выборках: A. CASIA-WebFace [43], B. MSCeleb-1M [19], C. AsianDataset [44] и D. AFAD-lite [45]. Разные исходные задачи, данные и функции потерь приводят к разным обученным весам сверточных фильтров, в итоге финальная модель как усреднение сетей A, B, C и D позволяет достичь 100.00% TPR в FPR=10^{-4} (табл. 4.2.).

Таблица 4.3. Влияние дополнительных модальностей на целевую метрику.

Модальность	TPR в точке FPR =		
	$\mathbf{10^{-2}}$	$\mathbf{10^{-3}}$	$\mathbf{10^{-4}}$
RGB	71.74	22.34	7.85
ИК	91.82	72.25	57.41
Глубина	100.00	99.77	98.40
RGB+ИК+Глубина	100.00	100.00	99.87

4.6. Влияние мультимодальности

Чтобы показать преимущество мультимодальных данных в задаче определения живости, мы исследовали сети, обученные только на одной модальности. Для честного сравнения использовалась та же архитектура, что и для мультимодальных изображений, только вместо подавания на вход (RGB, ИК, глубина), модели обучались на входах (RGB, RGB, RGB), (ИК, ИК, ИК) и (глубина, глубина, глубина).

`

Как видно в табл. 4.3., использование только одного канала RGB приводит к низкой точности. Соответствующая модель переобучилась на тренировочной выборке и достигла только 7.85% TPR в FPR=10^{-4}. Модель на инфракрасных данных оказалась лучше, показав 57.41% TPR в FPR=10^{-4}. ИК-данные содержат меньше мелких деталей, поэтому сети, основанные на них, сложнее переобучаются и в общем более устойчивы на неизвестных данных, что и показал текущий результат. Наиболее высокий результат 98.40% TPR в FPR=10^{-4} был получен на модальности глубина, подтвердив важность информации о форме для задачи проверки подлинности лица.

Но сеть, обученная на объединении модальностей, показала еще лучшую точность, понижая ложноотрицательную ошибку с 1.6 до 0.13 % и доказывая важность мультимодального подхода.

4.7. Заключение

В данном разделе был представлен новый метод для решения задачи детектирования фальсифицированных изображений лиц, который показал лучший результат на конкурсе "Chalearn LAP face anti-spoofing 2019". Были предложены три направления работы: данные, архитектура нейронной сети и инициализация весов. Комплексный подход выявил существенные улучшения точности по сравнению с базовым методом. Тщательный выбор обучающей подвыборки по типам атак позволяет модели лучше противостоять незнакомым попыткам взлома. Предложена новая архитектура сети с модулем мультиуровневевой агрегации признаков, что улучшило обмен полезными признаками между подсетями разных модальностей как на поверхностных, так и на глубоких слоях модели. Использован метод переноса признаков с обученных моделей распознавания лиц, что улучшило стабильность модели и увеличило точность на целевой выборке.

`

Глава 5

Методы определения живости по видеопоследовательности

В главе 2 рассматривался кооперативный метод оценки живости для мобильных и стационарных сценариев в условиях наличия небольшой выборки. В случае, когда доступна выборка большего размера, становится возможным создать некооперативный алгоритм с высокой точностью, эксплуатируя отличия в динамических признаках реальных и поддельных видео.

В 2020 году была опубликована расширенная выборка CASIA-SURF CeFa [46], включающая в себя новые виды атак (3D маски) и новые расы реальных людей, и также устроили конкурс на лучшее решение. В обеих конкурсах алгоритмы автора диссертации заняли первое место. На данный момент, CASIA-SURF CeFa [46] является самой большой выборкой данных для задачи определения подлинности лиц по количеству объектов, национальностей и типов атак. Для тестирования обобщающей способности моделей оценки живости, авторы предоставили различные тестовые протоколы, где в контрольной выборке присутствуют неизвестные атаки и национальности. Все данные представлены в виде коротких видеозаписей и точность меряется по ролику целиком, позволяя использовать алгоритмы, связанные с временным изменением кадра в процессе. Все видео записаны в трех модальностях: RGB, ИК и глубина. Авторы выборки устроили конкурс на лучшее решение по всем модальностям и по RGB отдельно. Мы решили сфокусироваться на втором соревновании. Примеры изображений из выборки показаны на рис. 5.1.

Классификация реальных и поддельных видео проще, чем покадровая классификация, так как можно использовать различия в мимике лица с течением времени [27, 36]. Но в то же время малое для нейронных сетей количество видеозаписей не позволяет использовать обучение в лоб. Мы предлагаем перейти от задачи классификации треков к задаче классификации изображений, вводя понятие искусственных модальностей, богатых полезными для определения живости признаками и в то же время без

`

лишних деталей, которые могут привести к переобучению. В качестве таких модальностей предлагается использовать оптический поток и ранг-пулинг [48]. В дополнение к этому, для расширения вариативности обучающей выборки предлагается использовать аугментацию последовательности, как в алгоритме по динамике трека из Главы 3. Кроме этого тут применяется очень легкая архитектура, позволяющая использовать решение не только в исследовательских, но и в прикладных целях.

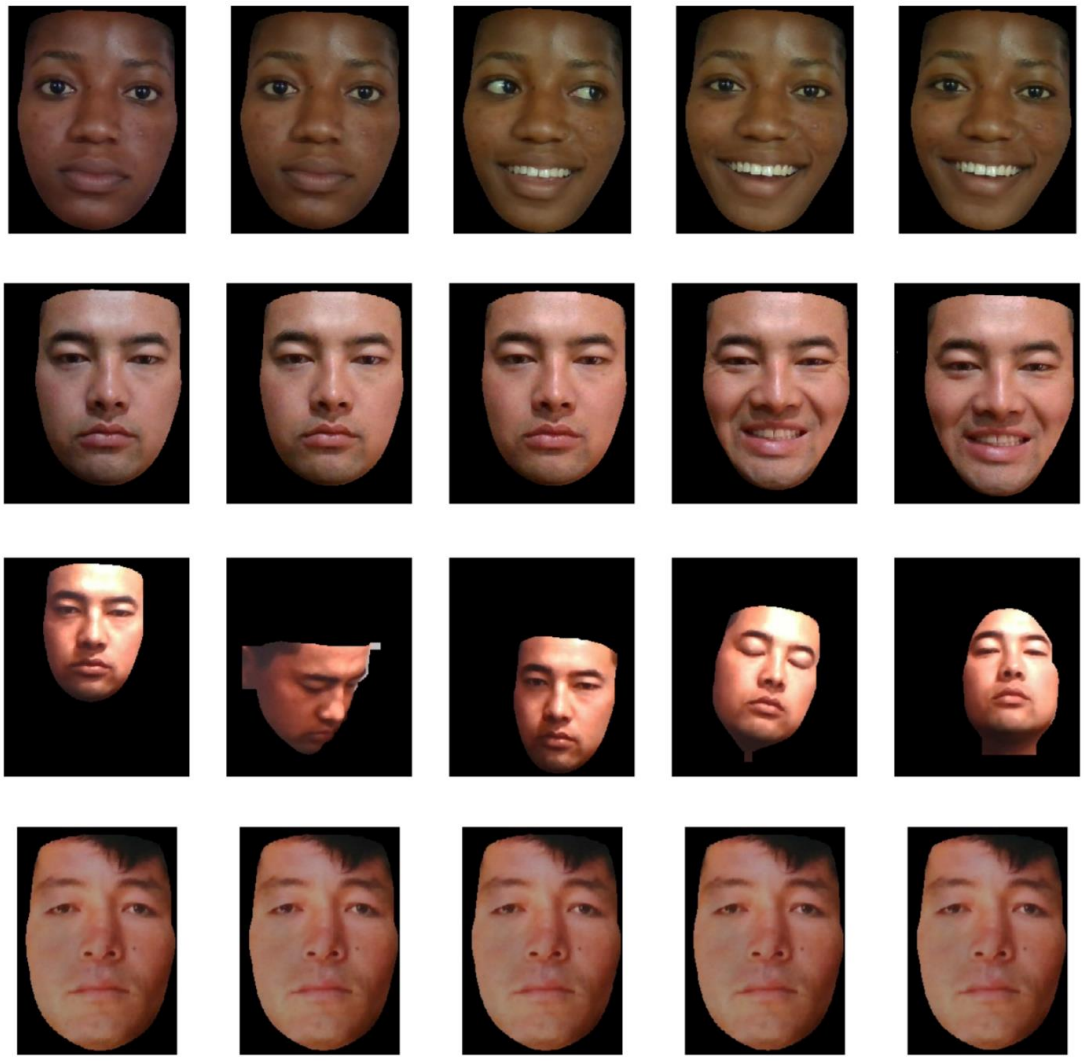

Рис. 5.1. Примеры реальных и поддельных изображений из выборки CASIA-SURF CeFa. Первая строка – реальный трек (обучение), вторая – реальный трек (тест. Третья строка – подделка(обучение), четвертая – подделка(тест).

`

Наше решение показало лучший результат на тестовой выборке CASIA-SURF CeFa по RGB изображениям и заняло первое место в соответствующем соревновании, приуроченном к конференции CVPR 2020.

5.1. Описание выборки

CASIA-SURF CeFa содержит записи 1607 разных людей трех национальностей и 4 вида атак, включая 3D маски. Для соревнования организаторы сформировали три протокола, где обучающая и тестовая выборка полностью отличается по национальностям и типам атак. Каждый протокол включает в себя 200 реальных и 200 поддельных видео для тренировки, 400 реальных и 1800 поддельных видео для теста. Видео были представлены как последовательность кадров с убранным фоном и центрированными лицами. Существуют различные метрики оценивания точности алгоритма определения живости. Одной из популярных метрик является average classification error rate (ACER), используемая в работах [21, 22, 38, 39]. Она применяется и в данном соревновании.

5.2. Предлагаемый метод

Так как различие в обучающей и тестовой выборке очень велико (рис. 5.1.), будем использовать искусственные *модальностей*. Хорошая искусственная модальность должна содержать мало мелких деталей (чтобы избежать переобучения), но в то же время обладать полезными для задачи определения живости признаками. Мы предлагаем использовать оптический поток и ранк-пулинг, которые обладают необходимыми свойствами.

Так как по условиям соревнования нельзя было использовать дополнительные выборки и предобученные модели, для оптического потока мы выбрали непараметрический метод из [47]. В финальном решении мы используем две карты потока, одна – между первым и последним кадром трека, другая – между первым и вторым. Для поддельных изображений обе карты должны быть примерно одинаковы, в то время как для реальных примеров первая карта покажет больше движения, чем вторая. Более

`

того, поток между первым и последним изображениями будет похож на трехмерную карту головы объекта по особенностям своего построения (подробнее в главе 1).

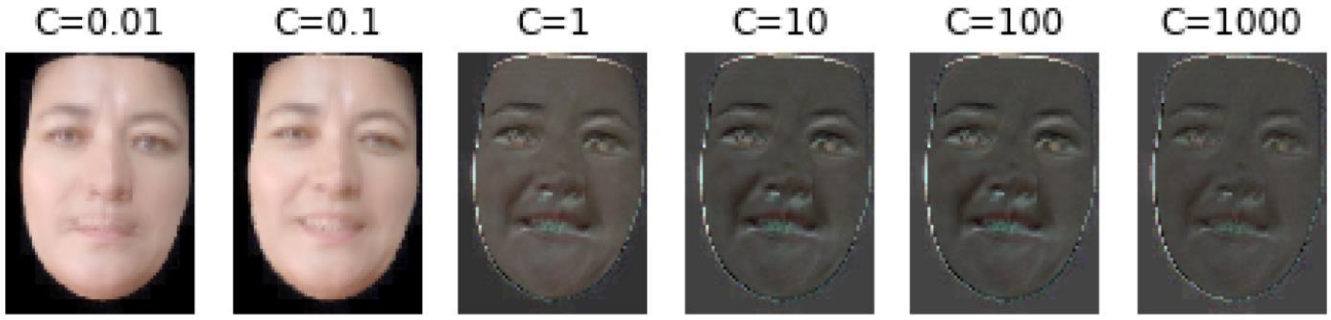

Рис. 5.2. Ранг-пулинг для разных значений параметра регуляризации C.

Модальность ранк-пулинга кодирует видеопоследовательность в вектор признаков с помощью процесса оптимизации, который может быть сформулирован как метод опорных признаков для задачи регрессии SVR [48]. После решения оптимизационной задачи вектор признаков можно визуализировать, получая динамическое изображение, которое отображает временную эволюцию покадровых признаков. В данной задаче мы выбрали гиперпараметры C=1 и C=1000, т.е. низкий и высокий уровень регуляризации для SVR, получив два визуально различных представления (рис. 5.2). C=1 сохраняет больше информации об объекте, в то время как C=1000 показывает изменение черт лица со временем.

Помимо предоставленных модальностей, для увеличения вариативности выборки мы используем *аугментацию последовательности* – преобразование реальных последовательностей в синтетические. Для этого в процессе обучения выбирается один кадр из реального трека и дублируется нужное число раз, после все кадры новой последовательности индивидуально аугментируются поворотами, сдвигом и цветовой коррекцией. Новое семейство поддельных треков больше похоже на распечатанные атаки, присутствующие в тестовой выборке.

`

5.3. Архитектура модели

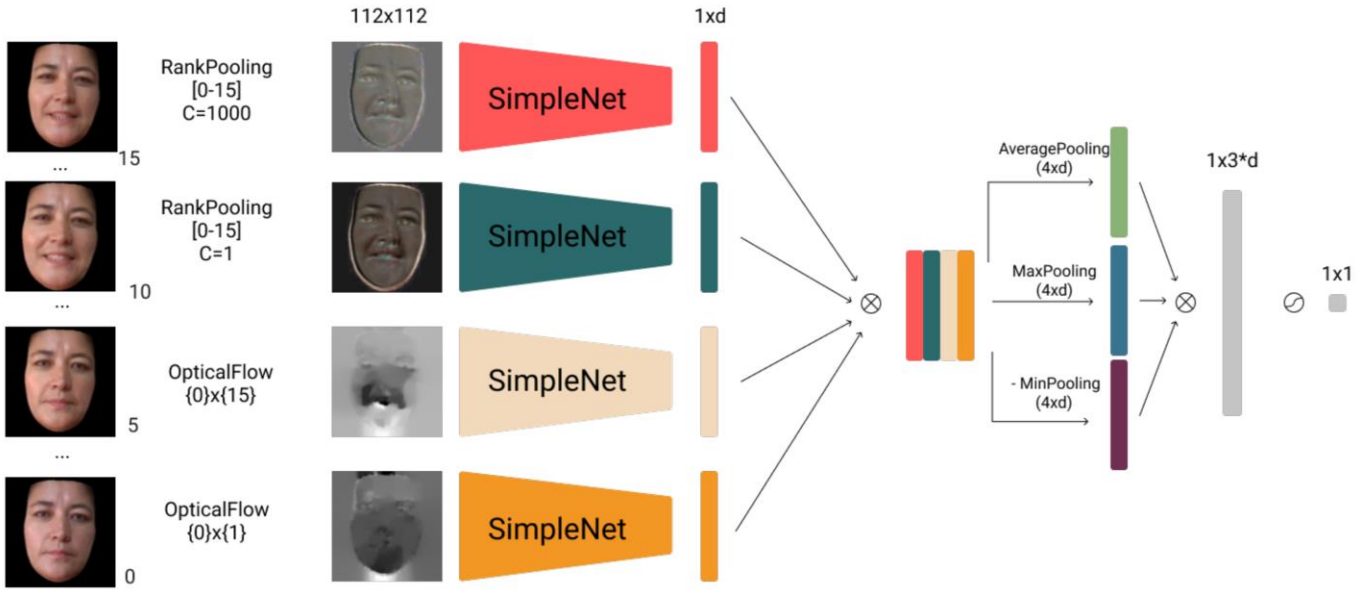

Рис. 5.3. Предлагаемая архитектура. 16 равномерно выбранных изображений из трека используются для получения 4х модальностей: 2 ранк-пулинга и 2 оптических потока. Модальности обрабатываются независимыми базовыми сетями SimpleNet, после чего агрегируются полносвязным слоем.

Для классификации полученных искусственных модальностей предлагается новая архитектура, показанная на рис 5.3. Используемые базовые нейросети SimpleNet достаточно глубокие для извлечения полезных признаков из изображений модальностей, но достаточно узкие, чтобы избежать переобучения.

Каждый из четырех полученных тензоров обрабатывается отдельной сетью SimpleNet, которые возвращают дескрипторы размера $1 \times d$. Дескрипторы конкатенируются, после чего к полученной $4 \times d$ матрице применяются операторы Max, Min и Avg пулинга, получая $3 \times d$ матрицу. Обработка завершается полносвязным слоем с сигмоидой.

`

В отличие от обычной конкатенации дескрипторов, предлагаемые операторы пулинга всегда приводят к фиксированной матрице размера $3 \times d$, позволяя легко расширить количество обрабатываемых модальностей и добавить новую даже уже к обученным другим. Кроме того, использование Max и Min пулинга позволяет выбрать признаки с модальностей, которые могут отличаться по важности от изображения к изображению и это работает лучше обычного оператора усреднения.

5.4. Эксперименты

Код написан в python с помощью библиотеки pytorch [41], обучался и тестировался на одной видеокарте NVIDIA RTX 2080. Модель обучалась с размером батча 32 и 8 CPU потоками. Для обучения достаточно 1.5G памяти. Модель тренировалась 10 эпох с оптимизатором ADAM [42] и начальным коэффициентом обучаемости 0.0001. Время обучения – 1 час, для каждого из трех протоколов применялись одинаковые настройки.

Предобработка изображений. Пусть дан трек $\{X_i\}$ и его метка $l = \{0,1\}$, где 0 – подделка, 1 – реальный. Из трека равномерно выбирается $L = 16$ изображений, После чего, с вероятностью 0.5 применяется аугментация последовательности, т.е. выбирается случайный кадр X_j и дублируется L раз, при этом метке класса присваивается 0.
У каждого изображения трека убираются черные границы, после чего дополняются до квадрата и изменяются до размера 112×112. Наконец, выбирается случайный параметр цветовой коррекции и применяется ко всем изображениям трека, эмулируя различный цвет кожи, после чего для каждого кадра X_i применяются независимые от других повороты, сдвиги, и цветовая коррекция.
Обработанные таким образом треки пропускаются через подсчет модальностей, получаются 4 тензора размера 112×112: RankPooling$(\{X_i\}, C = 1000)$, RankPooling$(\{X_i\}, C = 1)$, Flow(X_0, X_{15}), Flow(X_0, X_1). Далее, все подается на вход нейронной сети (рис. 5.3.).

`

Базовый эксперимент. Для задания базовой планки, мы обучили модель на сырых RGB изображениях без искусственных модальностей. Первый и последний кадр трека конкатенируются в тензор $6 \times 112 \times 112$, после чего подается на вход архитектуре из рис. 5.3 для честного сравнения. Такой метод достигает точности 23.42% ACER на тестовой выборке (табл. 5.1.). Большое стандартное отклонение BPCER говорит о нестабильности модели и неспособности к обобщению.

Таблица 5.1. Результаты на тестовой выборке CASIA-SURF CeFa.

Метод	APCER, %	BPCER, %	ACER, %
Базовый	23.83 ± 1.70	25.20 ± 22.00	23.42 ± 12.14
Ранк-пулинг(C=1000)	14.11 ± 13.52	11.25 ± 12.75	12.68 ± 4.39
+аугментация последовательности	0.68 ± 0.21	13.91 ± 10.03	7.30 ± 5.00
+Ранк-пулинг(C=1)	1.07 ± 0.53	13.00 ± 10.75	7.03 ± 5.20
+Оптический поток	0.11 ± 0.11	5.33 ± 2.37	2.72 ± 1.21

Ранк-пулинг. Чтобы показать преимущество использования искусственных модальностей, мы заменили входные данные базового эксперимента на одно изображение – ранк-пулинг (C=1000). Ошибка на тестовой выборке упала до 12.68%, доказывая, что использования динамических признаков без мелких деталей лучше, чем чистые RGB данные.

Аугментация последовательности. Этот эксперимент показывает, насколько можно улучшить итоговый результат простым преобразованием. Добавление аугментации последовательности к предыдущему эксперименту улучшило целевую метрику ACER до 7.3%. Но использование такой аугментации без индивидуальных цветовых и пространственных шумов для каждого изображения трека бесполезно, так как ранк-

`

пулинг для одинаковой последовательности будет неинформативным. Добавление еще одного ранк-пулинга с C=1000 дало небольшое улучшение до 7.03%, поэтому было решено остановиться на двух представителях данной модальности.

Оптический поток. Добавление модальности оптического потока к предыдущему эксперименту показало результат в 2.72% ACER на подвыборке CASIA-SURF CeFa RGB, что на текущий момент является лучшим результатом в мире. Карты оптического потока подчеркивают разницу в движении мимических мышц у реальных и поддельных людей. Атаки с использованием распечатанных масок выглядят менее интенсивно, если посмотреть на них сквозь призму оптического потока. Это доказывается результатом APCER=0.11%, т.е. всего 2 из 1800 примеров атак были неправильно классифицированы. Недостаток предложенного метода выражается в высокой ложноотрицательной ошибке BPCER=5.33%, где большинство неправильно классифицированных примеров относится к реальным трекам без значительных движений. Выбирать искусственные модальности следует осторожно – в нашем построении мы предполагали, что лицо будет меняться со временем, но в некоторых реальных сценариях это не обязательно правда.

5.5. Заключение

В данном разделе был предложен метод решения задачи определения живости с помощью создания искусственных модальностей и аугментации последовательности. Было показано, что аккуратный выбор промежуточного представления данных, как ранк-пулинг или оптический поток, уменьшают риск переобучения и повышают итоговую точность модели по сравнению с наивным использованием исходных изображений. Также была представлена быстрая и масштабируемая архитектура нейронной сети, применимая в прикладных задачах. Наконец, был показан простой трюк по обогащению поддельных данных, что всегда является узким местом для подавляющего большинства задач определения подлинности. В результате, предложенное решение

`

заняло первое место в соревновании Chalearn Singlemodal Face Anti-spoofing Attack Detection на конференции CVPR 2020.

`

Заключение

В первой главе были введены основные термины и описана специфика задачи определения живости. Была предложена система классификации сценариев применения метода и кооперативности пользовательского поведения. Описаны основные виды атак на биометрические системы и введено понятие уровня сложности атаки. Проведен обзор существующих методов, рассмотрены их достоинства и недостатки.

В главе 2 были предложены кооперативные методы оценки подлинности, основанные на интерактивном взаимодействии с пользователем. Был представлен атомарный алгоритм определения живости, обеспечивающий высокий уровень защиты, но неудобный для пользователя. Далее, были показаны улучшенные алгоритмы, требующие меньшего уровня кооперативности, основанные на оптическом потоке. Численные эксперименты на собранной выборке подтвердили эффективность предложенного метода.

В главе 3 была рассмотрена задача определения подлинности для систем контроля и управления доступом. Собрана обучающая и тестовые выборки, а также описаны методы синтеза новых данных, эффективно увеличивающие размер и вариативность выборки для обучения. Были предложены три алгоритма определения живости в описанном сценарии. Все алгоритмы оптимизированы под скорость работы для возможности внедрения в промышленные объекты. Первый алгоритм базируется на идее различия реальных и подделок по одному изображению. Метод хорошо работает на вырезанных масках, но неустойчив к полноразмерным артефактам. Второй алгоритм основывается на идее различия границ подлинных и поддельных изображений. Метод устойчив к распечатанным маскам и экранным демонстрациям, но пасует перед полноразмерными подделками. Третий метод эксплуатирует идею динамического изменения лицевой мимики и углов наклона головы при подходе человека к турникету. Ансамбль алгоритмов показывает высокую точность и может быть использован в прикладных условиях.

`

Четвертая глава посвящена некооперативным алгоритмам оценки живости для мобильных и стационарных устройств по мультимодальным данным. Предложен алгоритм, работающий с мультимодальными изображениями (RGB, ИК, Глубина). Разработано универсальное улучшение мультимодальных архитектур нейронных сетей, позволяющее лучше агрегировать признаки на всех уровнях детализации. Эксперименты показали, что новые модальности добавляют полезные признаки и улучшают точность на целевой метрике. Предложенное решение заняло первое место на самой крупной на момент разработки алгоритма мультимодальной выборке CASIA-SURF в 2019 году.

В пятой главе предлагается алгоритм определения подлинности для мобильных и стационарных устройств по видеопоследовательности, устойчивый против неизвестных атак. Помимо скорости работы в режиме реального времени, предложенный метод стал лучшим на соревновании по оценки живости “Chalearn Face Anti-Spoofing Challenge” в 2020 году.

Благодарность

Я благодарю своего научного руководителя В.И. Цуркова и профессора кафедры интеллектуальных систем И.А. Матвеева за полезные замечания в процессе подготовки текста диссертационной работы.

Я выражаю признательность компании “ВижнЛабс” за предоставленные вычислительные ресурсы для проведения экспериментов и помощь при сборе обучающих выборок. Также благодарю руководителя отдела исследований компании И.А. Лаптева за ценные советы по разработке практически применимых алгоритмов.

Наконец, выражаю благодарность моему коллеге А.Н. Паркину, в соавторстве с которым были выиграны два конкурса по определению живости изображений лиц.

`

Список литературы

1. *Schuckers S.* Spoofing and Anti-Spoofing Measures // Information Security Technical Report. – 2002. – Vol. 7, no. 4. – P. 56-62.
2. *G. Kim, S.Eum, J. K. Suhr, D. I. Kim, K. R. Park, J. Kim.* Face liveness detection based on texture and frequency analysis. // 5th IAPR Int. Conf. on Biometrics (ICB). – New Delhi, India, 2012. – P. 67-72.
3. *J. Maatta, A. Hadid, M. Pietikainen*. Face Spoofing Detection From Single images Using MicroTexture Analysis. // Proc. Int. Joint Conf. on Biometrics (UCB). -– Washington, D.C., USA, 2011.
4. *J. Li Y. Wang, T. Tan, A.K. Jain.* Live face detection based on the analysis of Fourier spectra. // Proc. of Biometric Technology for Human Identification. – Orlando, FL, USA, 2004. – P. 296-303.
5. *S. Kim, S. Yu, K. Kim, Y. Ban, S. Lee.* Face liveness detection using variable focusing // Int. Conf. on Biometrics (ICB). – 2013. – P. 1-6.
6. *H. K. Jee, S. U. Jung, J. H. Yoo.* Liveness detection for embedded face recognition system. // Int. Journal of Biological and Medical Sciences – 2006 – Vol. 1(4). – P. 235-238.
7. *W. Bao, H. Li, N. Li, W. Jiang.* A liveness detection method for face recognition based on optical flow field. // Int. Conf. on Image Analysis and Signal Processing – 2009. – P. 233-236.
8. *K. Kollreider H. Fronthaler, J. Bigun.* Evaluating liveness by face images and the structure tensor. // Proc. of 4th IEEE Workshop on Automatic Identification Advanced Technologies. – Washington DC, USA, 2005. – P. 75-80.
9. *K. Kollreider H. Fronthaler, J. Bigun.* Non-intrusive liveness detection by face images. // Image and Vision Computing. – 2009. – Vol. 27(3). – P. 233-244.

`

10. *L. Sun, G. Pan, Z. Wu, S. Lao.* Blinking-Based Live Face Detection Using Conditional Random Fields. // ICB 2007, Seoul, Korea. – P. 252-260.
11. *G. Pan, Z. Wu, Lin Sun. Liveness Detection for Face Recognition.* // Recent Advances in Face Recognition. – I-Tech, 2008. – P. 236.
12. *J. Yang, Z. Lei, S. Liao, S. Li.* Face Liveness Detection with Component Dependent Descriptor. // Int. Conf. on Biometrics (ICB) – 2013 – P. 1-6.
13. *A. Lagorio, M. Tistarelli, M. Cadoni.* Liveness Detection based on 3D Face Shape Analysis. // Biometrics and Forensics International Workshop (IWBF) – 2013 – P. 1-4.
14. T. *Wang, J. Yang, Z. Lei, S. Liao, S. Z. Li.* Face Liveness Detection Using 3D Structure Recovered from a Single Camera. // Int. Conf. on Biometrics. – Madrid, Spain – 2013.
15. *X. Tan, Y. Li, J. Liu, L. Jiang.* Face liveness detection from a single image with sparse low rank bilinear discriminative model. // ECCV. – 2010.
16. *B. Peixoto, C. Michelassi, A. Rocha.* Face liveness detection under bad illumination conditions. // ICIP. – 2011. – P. 3557-3560.
17. *J. Yan, Z. Zhang, Z. Lei, D. Yi, S. Z. Li.* Face liveness detection by exploring multiple scenic clues. // ICARCV. – 2012.
18. *Phillips J., Yates A., Hu Y. u ðp.* Face recognition accuracy of forensic examiners, superrecognizers, and face recognition algorithms // Proc. of the National Academy of Sciences. – 2018. – V.15. – P.6171–6176.
19. *Guo Y., Zhang L., Hu Y., He. X, Gao J.* MS-Celeb-1M: A Dataset and Benchmark for Large Scale Face Recognition // European Conf. on Computer Vision. – Amsterdam. – 2016.
20. *Parkhi O., Vedaldi A., Zisserman A.* Deep Face Recognition // British Machine Vision Conf. Swansea, – UK. – 2015.
21. *Boulkenafet Z., Komulainen J., Li L., Feng X., Hadid A.* Oulu-npu: A Mobile Face Presentation Attack Database with Real-world Variations // Conf. on Automatic Face and Gesture Recognition. – Washington, DC. –2017.

`

22. *Costa-Pazo A., Bhattacharjee S., Vazquez-Fernandez E., Marcel S.* The Replay-Mobile Face Presentation-attack Database // Proc. of the Int. Conf. on Biometrics Special Interests Group (BioSIG). – Darmstadt. – 2016.
23. *Chingovska I., Erdogmus N., Anjos A., Marcel S.* Face Recognition Systems under Spoofing Attacks // Face Recognition Across the Imaging Spectrum. – Springer, Cham. – 2016.
24. *Erdogmus N., Marcel S.* Spoofing in 2D Face Recognition with 3D Masks and Anti-spoofing with Kinect // IEEE Sixth Int. Conf. on Biometrics: Theory, Applications and Systems (BTAS). – Arlington, VA. – 2014.
25. *Zhang S., Wang X., Liu A. u ∂p.* A Dataset and Benchmark for Large-scale Multi-modal Face Anti-spoofing // CVPR. – Long Beach, CA. – 2019.
26. *Liu A., Wan J., Escalera S. u ∂p.* Multi-modal Face Anti-spoofing Attack Detection Challenge at CVPR2019 // CVPR workshop. – Long Beach, CA. – 2019.
27. *Pan G., Sun L., Wu Z., Lao S.* Eyeblink-based Anti-spoofing in Face Recognition from a Generic Webcamera // Int. Conf. on Computer Vision. –Venice. – 2007.
28. *Wang L., Ding X., Fang C.* Face Live Detection Method Based on Physiological Motion Analysis. // Tsinghua Science and Technology. – 2009. – Vol.14. – P.685-690.
29. *.Bharadwaj S., Dhamecha T., Vatsa M., Singh R.* Computationally Efficient Face Spoofing Detection with Motion Magnification. // CVPR workshop. –Portland. – 2013.
30. *Feng L., Po L., Li Y. u ∂p.* Integration of Image Quality and Motion Cues for Face Antispoofing: A Neural Network Approach // Journal of Visual Communication and Image Representation. – 2016. – V.38. – P.451-460.
31. *Patel K., Han H., Jain A.* Secure Face Unlock: Spoof Detection on Smartphones // Transactions on Information Forensics and Security. – 2016. V.11. – P.2268-2283.
32. *Li L., Feng X., Boulkenafet Z., Xia Z., Li M., Hadid A.* An Original Face Antispoofing Approach Using Partial Convolutional Neural Network // Sixth Int. Conf. on Image Processing Theory, Tools and Applications (IPTA). –Oulu. – 2016.

`

33. *Liu Y., Jourabloo A., Liu X.* Learning Deep Models for Face Anti-Spoofing: Binary or Auxiliary Supervision // CVPR – Salt Lake City. – 2018.
34. *Chingovska I., Anjos A., Marcel S.* On the Effectiveness of Local Binary Patterns in Face Antispoofing // Proc. of the Int. Conf. on Biometrics Special Interests Group (BioSIG). – Darmstadt. – 2012.
35. *Zhang Z., Yan J., Liu S. и др.* A Face Antispoofing Database with Diverse Attacks // International Conference on Biometrics. – New Delhi. –2012.
36. *Wen D., Han H., Jain A.* Face Spoof Detection with Image Distortion Analysis // Transactions on Information Forensics and Security. – 2015. –V.10. – P.746–751.
37. *He K., Zhang X., Ren S., Sun J.* Deep Residual Learning for Image Recognition // CVPR. – Las Vegas. – 2016.
38. *Jourabloo A., Liu Y., Liu X.* Face Despoofing: Anti-spoofing via Noise Modeling // European Conf. on Computer Vision. – Munich. – 2018.
39. *Визильтер Ю.В., Желтов С.Ю.* Использование проективных морфологий в задачах обнаружения и идентификации объектов на изображениях // Изв. РАН. ТиСУ. – 2009. – № 2. – P.125–138.
40. *Deng J., Dong W., Socher R., Li L.-J., Li K., Fei-Fei L.* ImageNet: A Large-Scale Hierarchical Image Database // CVPR. – Miami. – 2009.
41. *Paszke A., Gross S., Chintala S. и др.* Automatic Differentiation in PyTorch // NIPS workshop. – Long Beach, CA. – 2017.
42. *Kingma D., Ba J. Adam: A Method for Stochastic Optimization* // Int. Conf. on Learning Presentations. – San Diego. – 2015.
43. *Yi D., Lei Z., Liao S., Li S.* Learning Face Representation from Scratch // arXiv. – 2014. – V.1411.7923.
44. *Zhao J., Cheng Y., Xu Y. и др.* Towards Pose Invariant Face Recognition in the Wild // CVPR. – Salt Lake City. – 2018.
45. *Niu Z., Zhou M., Wang L., Gao X., Hua G.* Ordinal Regression With Multiple Output CNN for Age Estimation // CVPR. – Las Vegas. – 2016.

`

46. *A. Liu, Z. Tan, X. Li, J. Wan, S. Escalera, G. Guo, S. Li.* Static and Dynamic Fusion for Multi-modal Cross-ethnicity Face Anti-spoofing. // ArXiv. – 2019.
47. *C. Liu.* Beyond pixels: Exploring new representations and applications for motion analysis. //. . –MIT. – 2009.
48. *B. Fernando, E. Gavves, J. Oramas, A. Ghodrati, T. Tuytelaars.* Rank pooling for action recognition. // TPAMI. – 2017.
49. *Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T.; Andreetto, M., Adam, H.* MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. // arXiv – 2017. – V.1704.04861.
50. B. *Zhou, A. Khosla, A. Lapedriza, A. Torralba, A. Oliva.* Places: A 10 million Image Database for Scene Recognition.// arXiv – 2016. – V. 1610.02055.
51. *Nguyen, Sang & Truong Quang, Vinh.* FPGA implementation for real-time Chroma-key effect using Coarse and Fine Filter. // Int. Conf. on Computing, Management and Telecommunications, ComManTel. - 2013. – P. 157-162.
52. *J.Chung, A. Zisserman.* Lip Reading in the Wild. // Asian Conf. on Computer Vision. – 2016.
53. *X. Liang, C. Xu, X. Shen. J. Yang.* Deep Human Parsing with Active Template Regression. // ICCV. – 2015.
54. *Shie Mannor, Dori Peleg, and Reuven Rubinstein.* The cross entropy method for classification. // Proc. of the 22nd Int. Conf. on Machine learning (ICML). – 2005. – P.561-568.
55. *Y. Sun, X. Wang, X. Tang.* Deep Convolutional Network Cascade for Facial Point Detection. // CVPR. – 2013. – P. 3476-3483.
56. *Deqing Sun, Xiaodong Yang, Ming-Yu Liu, Jan Kautz.* PWC-Net: CNNs for Optical Flow Using Pyramid, Warping, and Cost Volume. // Proc. of the IEEE Conf on Computer Vision and Pattern Recognition (CVPR). – 2018. – P. 8934-8943.
57. *Z. Liu, P. Luo, X. Wang, X. Tang.* Large-scale CelebFaces Attributes (CelebA) Dataset. // ICCV. - 2015.

`

58. A. *Tao, K. Sapra, B. Catanzaro*. Hierarchical Multi-Scale Attention for Semantic Segmentation. // arXiv. – 2020.
59. *T. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. Zitnick, P. Dollár*. Microsoft COCO: Common Objects in Context. // arXiv. – 2014.
60. *J. Perng, P. Liu, K. Zhong, Y. Hsu.* Front object recognition system for vehicles based on sensor fusion using stereo vision and laser range finder. // IEEE Int. Conf. on Consumer Electronics – Taipei. – 2017. – P. 261-262.
61. *J. Ciberlin, R. Grbic, N. Teslić, M. Pilipović*. Object detection and object tracking in front of the vehicle using front view camera. // Zooming Innovation in Consumer Technologies Conf. (ZINC). – Novi Sad, Serbia. – 2019. – P. 27-32.
62. Global Face and Voice Biometrics Industry. // Global industry – 2020. - https://www.reportlinker.com/p05818230/Global-Face-and-Voice-Biometrics-Industry.html
63. Biometric presentation attack detection — Part 3: Testing and reporting. // ISO/IEC 30107-3:2017 – 2020 – Information Technology.
64. *X. Zhu, C. Vondrick, C. Fowlkes, D. Ramanan.* Do We Need More Training Data? // arXiv– V1503.01508. – 2015.
65. *Воронцов К.В.* Математические методы обучения по прецедентам. – Machine Learning. – 2015.
66. *M. Tan, Q. Le*. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. // ICML. – 2019.
67. Kelley H.J. *Gradient theory of optimal flight paths*. // Ars Journal – V. 30(10). – 1960. – P.947–954.
68. *Figueroa R. L.* Predicting Sample Size Required for Classification Performance. // BMC medical informatics and decision making. – V. 12(1):8. – 2012.
69. *J. Cho.* How Much Data Is Needed to Train A Medical Image Deep Learning System to Achieve Necessary High Accuracy? – arXiv. – V.1511.06348. – 2015.

`

70. *S. Lei.* How Training Data Affect the Accuracy and Robustness of Neural Networks for Image Classification. // ICLR. – 2019.
71. *J. Canny.* A Computational Approach to Edge Detection. // IEEE Transactions on Pattern Analysis and Machine Intelligence. – 1986.
72. *A. Parkin, O. Grinchuk*. Recognizing Multi-Modal Face Spoofing With Face Recognition Networks. // Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition Workshops. – 2019.
73. *Y. Zhu, S. Newsam.* Densenet for dense flow. // IEEE Int. Conf. on image processing (ICIP). – 2017. – P.790–794.
74. *P. Viola, M. Jones.* Robust Real-Time Face Detection // Int. J. Comput. Vision. — 2004. — may. — Vol. 57, no. 2. — P. 137–154.
75. *S. Zhang, X. Zhu, Zhen Lei, H. Shi, X. Wang, S. Z. Li.* S3FD: Single Shot Scale-invariant Face Detector. // ICCV. – 2017.
76. *K. Gurney.* An Introduction to Neural Networks. // Taylor & Francis, Inc. – 1997. – Philadelphia, PA.
77. *S. Zagoruyko, N. Komodakis.* Wide Residual Networks. // arXiv. – 2017. – V. 1605.0146.