

Министерство образования и науки Российской Федерации
Московский физико-технический институт (национальный
исследовательский университет)

Физтех-школа радиотехники и компьютерных технологий
Кафедра интеллектуальных информационных систем и технологий

Выпускная квалификационная работа бакалавра

Исследование методов классификации
движений человека

Автор:

Студент Б01-815а группы
Токарев Андрей Сергеевич

Научный руководитель:

Ст. Преподаватель
Воронков Илья Михайлович



Москва 2022

Аннотация

Исследование методов классификации движений
человека

Токарев Андрей Сергеевич

После достижения результатов в задаче сегментации человека и отслеживании его на видео стало интересна возможность понимая занимаемой им позы и ее классификация. Для этого было необходимо научиться распознавать некоторые ключевые точки на теле человека, которые несут информацию о его положении. Некоторые группы исследователей уже добились хороших результатов в данном направлении и двигаются дальше в задачах анализа взаимодействия между людьми и предсказания будущих движений человека.

В данной работе представлен обзор задачи классификации движений на основе информации о ключевых точках на теле человека, а также приведены технический обзор и исследование систем глубокого обучения для оценки позы на основе публичных наборов данных.

Содержание

1 Введение	4
2 Постановка задачи	9
2.1 Задача распознавания ключевых точек на теле человека	9
2.2 Задача классификации движений/позы человека	11
3 Обзор существующих моделей	13
3.1 Модели для распознавания ключевых точек на теле человека	13
3.1.1 BlazePose	13
3.1.2 MoveNet.SinglePose	15
3.1.3 OpenPose	17
3.1.4 MMPose	18
3.1.5 AlphaPose	19
3.1.6 DeepPose	20
3.2 Модели для классификации позы человека	22
3.2.1 Классификатор от MediaPipe	22
3.2.2 mmakos	23
3.2.3 MMaction2 by OpenMMLab	24
4 Исследование моделей	25
4.1 Описание эксперимента	25
4.2 Поиск данных	29
4.3 Результаты эксперимента	32
5 Заключение	37
Список литературы	38
А Приложение	43

1 Введение

Понимание движений человека является необходимой частью нашей жизни. При общении людьми часто используется жестикуляция, так как это помогает выражать чувства, эмоции и доносить свои мысли до окружающих . Из анализа позы человека можно сделать вывод о его состоянии. К примеру, хромота или нахождение в неестественном положении говорят о необходимости не только медицинской, но, возможно, и вашей помощи. Ещё можно обратиться к психоанализу, а точнее к разделу о языке телодвижений. В нем по позе можно сделать вывод о характере человека или о текущем состоянии, его заинтересованности в беседе. Также работает распознавание движений. Если мы видим бегущих в панике людей, то наш мозг получает сигнал об опасности и спасает нас. Из приведенных ситуаций становится понятно, почему определение позы и классификация движений являются важными аспектами нашей жизни. В связи с развитием информационных технологий, человечество задумалось над выполнением данной задачи с помощью компьютера. Тогда можно будет добавить дополнительный источник информации для взаимодействия искусственного интеллекта с человеком.

При рассмотрении данной задачи через призму машинного обучения, получим, что нам нужно классифицировать положение человека, данные о котором необходимо каким-то образом получать. Первый способ - надеть на добровольца датчики и, считывая координаты каждого из них, построить на компьютере его позу и, таким образом, восстановить скелет для последующего анализа. Второй способ - искать особые точки на фотографии с помощью компьютерного зрения. Установим камеру и начнем анализировать положение и скелет человека, исходя из картинки. Тогда не придется закупать большое количество датчиков для снятия данных, а нужна будет только камера и вычислительные мощно-

сти для работы алгоритмов глубокого обучения. Несмотря на сложность реализации первого варианта, его удобно использовать для подготовки тренировочных датасетов [1].

Движение - это растянутый во времени процесс. Он анализируется по видеозаписям, каждая из которых представляет собой последовательность кадров. Поэтому первостепенно научиться работать с изображением. Как же собирать данные для модели классификации?

Восстановление скелета (Skeletal Representation), детекция (Pose Detection) и оценка позы (Pose Estimation), распознавание движения (Action Recognition) являются расширением одной задачи: распознавание ключевых точек на теле человека (Key-points Detection). Задачи, которая имеет прикладной смысл не только в связке с классификацией движений. В работе мы будем рассматривать распознавание только на картинке, то есть в 2-х мерном пространстве. Но ведь можно восстанавливать положение человека (скелет человека) в 3-х мерном пространстве [2, 3]. Используя генеративные нейронные сети можно воссоздавать не только скелет человека, но и тело человека [4]. Объединяя две предыдущие задачи можно получить набор данных из 3-х мерных людей в различных позах. Некоторые исследователи уже пробуют реализовать этот симбиоз на практике [5].

Если перейти в тематику биологических и медицинских наук, то можно развить данную тему на примере восстановления структуры тканей человека. Получается, мы сможем по фотографии моделировать расположение мышечных, жировых и других тканей в теле человека. Это поможет более детально изучать проблемы персонально, каждого человека и подбирать индивидуальные курсы лечения или диеты.

Восстановление скелета человека поможет спасателям анализировать положение человека под завалами и строить планы по его спасению,

имея более детальную информацию. Правда в данном случае необходимо быстродействие алгоритма и очень важно получить изображение человека.

В современном мире, где повсюду слышны разговоры о технологиях дополненной реальности и мета вселенной, найдем ещё одно применение для алгоритмов детектирования позы. Для нахождения в виртуальной вселенной необходимо транслировать человека туда, а значит можно с помощью видеокамер определять положение, восстанавливать скелет и получать итоговое изображение или 3-х мерную модель. Чем-то напоминает фильм "Первому Игроку Приготовиться" Стивена Спилберга. Добавим алгоритм генерации аватара вместо реального человека и получим рабочий алгоритм трансляции живого человека в мета вселенную.

Второй частью работы является задача классификации (Pose Classification), которая использует данные, полученные в первой части. Таким образом, мы построили алгоритм анализа движений человека на статическом изображении. По изображению мы не можем давать оценку поведению человека, но своеобразный "помощник" из полученного алгоритма будет хороший. Рассмотрим некоторые идеи применения.

Начать можно с медицины. Восстановление больных после операций, травм и несчастных случаев - это длительный и трудоемкий процесс, требующий постоянного присмотра врача. Если человек учится двигаться, то нужен тренер, который укажет на ошибки и исправит вас. Решение нашей задачи помогает таким пациентам. Анализ движений может сравнивать человека с эталоном и указывать на ошибки. Также при наблюдении за больным алгоритм может идентифицировать отклонения от нормального поведения и вызвать врача (к примеру увидеть приступы эпилепсии у человека). Это может спасти множество жизней по всему миру, просто вызывая врача в необходимый момент, а также помочь в

востановлении.

Также можно выявлять у здорового человека заболевания или дефекты скелета. Можно анализировать сколиоз или сутулость и подсказывать людям, что надо стараться держать спину прямо. Хотя лучше направлять к врачу на консультацию и лечить дефекты позвоночника сразу. Проведя исследование населения, мы получим статистику тех или иных отклонений. Так уже сделали производители кроссовок и с помощью gait-анализа [6] помогут выбрать подходящую обувь.

Посмотрим теперь на спорт. Из классификатора можно сделать хорошего судью соревнований в тех видах, где надо различать, отслеживать положения тела. К примеру, GOOGLE придумали использовать классификатор как счетчик подтягиваний, приседаний или отжиманий [7] и это можно поместить в современный смартфон. Если углубиться дальше, то решение можно обернуть интерфейсом и создать хорошего робота-фитнес-тренера. Ведь настроив камеру смартфона на наблюдение за вами во время тренировки, приложение будет подсказывать вам правильную позу для упражнения и укажет на ошибки, если таковые имеются.

При развитии моделей в будущем, можно будет найти другие варианты применения технологии классификации движений человека. Можно анализировать поведение группы людей, но для этого надо хорошо восстанавливать скелет нескольких человек на одном изображении [8, 9, 10]. Также есть возможность предсказывать будущие действия человека при изучении уже имеющихся [11]. Если опять затронуть идею генеративных нейронных сетей, то можно генерировать движения человека по заданному начальному условию. Следовательно, можно создавать искусственные видеозаписи или добавлять неигровых персонажей (пр - non-player character) в виртуальную реальность. В медицине можно моделировать

восстановление двигательной активности человека или моделирование протезов индивидуально под каждого пациента.

Как можно заметить, применений можно придумать множество - необходимо реализовать проект и получить модель. В текущий момент в мире уже существует какое-то количество решений описанных выше задач. В предложенной вашему чтению работе я рассмотрю некоторые из них, приведу качественную оценку результатам эксперимента и сделаю вывод с определением дальнейшего моего развития в данной теме. (МОЖЕТ СТОИТ УБРАТЬ ПРО МОЕ РАЗВИТИЕ В ДАННОЙ ТЕМЕ?)

2 Постановка задачи

Здесь надо максимально формально описать суть задачи, которую потребуется решить, так, чтобы можно было потом понять, в какой степени полученное в результате работы решение ей соответствует. Текст главы должен быть написан в стиле технического задания, т.е. содержать как описание задачи, так и некоторый набор требований к решению

Как уже было сказано в главе 1, будет произведена классификация движений человека на изображении. Из изображения надо получить данные о принимаемой субъектом позе и классифицировать её на род деятельности человека. Получается мы решаем две задачи: предобработки данных, то есть извлечение расположения ключевых точек на теле человека, и их последующая категоризация. Рассмотрим их по отдельности.

2.1 Задача распознавания ключевых точек на теле человека

Первоначально необходимо понять каким образом можно распознать позу человека, чтобы в дальнейшем взять оттуда информацию для классификации. Человек смотрит на другого человека и анализирует его позицию исходя из данных о его расположении частей тела анализируемого. Получается нам необходимо найти части тела человека, каждая из которых ограничена какими-либо суставами. Последние можно и взять за ключевые точки, которые будут распознаваться моделью. Если соединить выходные данные, то получим своеобразный "скелет" человека.

Необходимо определиться сколько точек на теле человека необходимо различать. На текущий момент стандартом является топология СО-СО (см. рис. 1а), которая включает в себя 17 ориентиров на теле человека [12, 13]. Данная топология не учитывает расположение ступней и кистей

рук, а также рассматривает всего 5 точек на лице человека: нос, два глаза и два уха. Но стандартом многие исследователи не ограничиваются и добавляют дополнительные точки. Приведу два примера:

1. Топология от BlazePose (см. рис. 1b)

Включает в себя 33 точки расположенные на теле человека. Данная топология представляет собой объединение COCO, BlazeFace [14] и BlazePalm [15]. В итоге мы получаем дополнительную информацию о направлении стоп и кистей, а также больше понимаем насчет точек на лице. Данная модель расположения точек используется в одноименной модели (BlazePose [16]) и ориентирована на использование в фитнес приложениях. Также у данной компании есть более развитая модель, которая определяет положение всех пальцев кисти и распознает мимику на лице [17].

2. Halpe (см. рис. 1c)

Данная топология - это совместный проект AlphaPose [10] и НАКЕ [18]. Представлено две модели: на 26 и на 136 точек. Здесь добавлено рассмотрение ориентации стоп, распознавание шеи, паха и макушки головы. В расширенной модели присутствует ещё 68 точек на лице, а также по 21 на ладонях.

В итоге мы разобрались с тем что нам необходимо искать в нашей работе и сейчас необходимо понять как это делать. Данная работа проводится в два этапа:

1. Локализация человека и его частей тела

2. Упорядочивание и распределение суставов в правильном порядке.

В реальном мире мы имеем два подхода к поиску ключевых точек и восстановлению скелета на изображении:

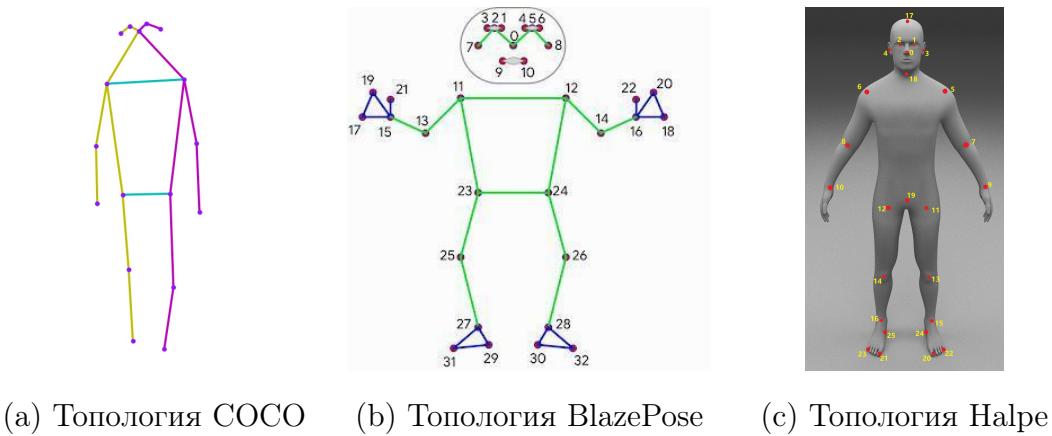


Рис. 1: Примеры расположения точек на теле человека.

- Bottom-up

Когда сначала распознаем точки, потом собираем их в скелет

- Top-down

Сначала происходит локализация людей или объектов, а потом происходит распознавание ключевых точек

ДУМАЮ ТУТ СТОИТ ПРОДОЛЖИТЬ ИЗ РАЗДЕЛА МНОГО ТЕОРИИ ДЛЯ РАЗБАВКИ РАБОТЫ. О ТОМ КАК ДЕТЕКТИРОВАТЬ И ИСКАТЬ ЭТИ ТОЧКИ. ТАКЖЕ МОЖНО ДОБАВИТЬ ПРО РАЗЛИЧНЫЕ ФУНКЦИИ И ФОРМУЛКИ. ДОЛЖНО ПОЛУЧИТЬСЯ КРАСИВО И ИНТЕРЕСНО.

2.2 Задача классификации движений/позы человека

Исходя из информации о полученных ключевых точках можно сделать вывод о категории движения, которое может совершаться в данной позе.

Полученные координаты ключевых точек можно использовать как признаки для различных классификаторов. Думаю тут не стоит сильно заморачиваться с описанием задачи машинного обучения классификации. Может просто забить, а может расписать и добавить ещё несколько

страниц. СПРОСИТЬ ПРО ДАННЫЙ РАЗДЕЛ!!!

3 Обзор существующих моделей

3.1 Модели для распознавания ключевых точек на теле человека

В данном разделе мы рассмотрим 6 различных моделей. В разд. 4 мы выберем 4 наиболее удобные в использовании и в обучении и проведем эксперимент по оценке данных моделей.

Так же хочется сказать, что, помимо приведенных, есть множество моделей от одиночных авторов, не объединенных в лаборатории [19, 20]. Они в основном брали какую-то из представленных ниже моделей и проводили небольшое улучшение.

А теперь перейдем к моделям.

3.1.1 BlazePose

MediaPipe является одним из проектов компании GOOGLE и в своей работе решает задачи компьютерного зрения. В нем уже были представлены модели для распознавания лица (Face Detection) и его поверхности (Face Mesh), ладоней (Hands), объектов (Object Detection и Objectron) и другие [21]. Для нас же интересна задача поиска ключевых точек, которую и решает модель BlazePose [16]. На момент исследования модель умеет отслеживать движения человека на видеофрагменте и строить по-кадровую маску человека.

Для предложенной модели была создана топология, которая представляет собой суперпозицию топологии COCO и двух других топологий, уже использовавшихся в других подпроектах MediaPipe. Об этом более подробно написано в разд. 2.1.

В BlazePose используется top-down подход оценки позы человека. Сна-

чала запускается Pose Detector (см рис. 2), который возвращает координаты интересующей нас области (region-of-interest или ROI). Алгоритм используем расширение модели BlazeFace для определения наличия человека в кадре. Поэтому данная модель чувствительна к видимости головы, лица в частности, на фотографии. Взяв идею витрувианского человека Леонардо Да Винчи, исследователям понадобилось ещё две точки для точной локализации человека на изображении.

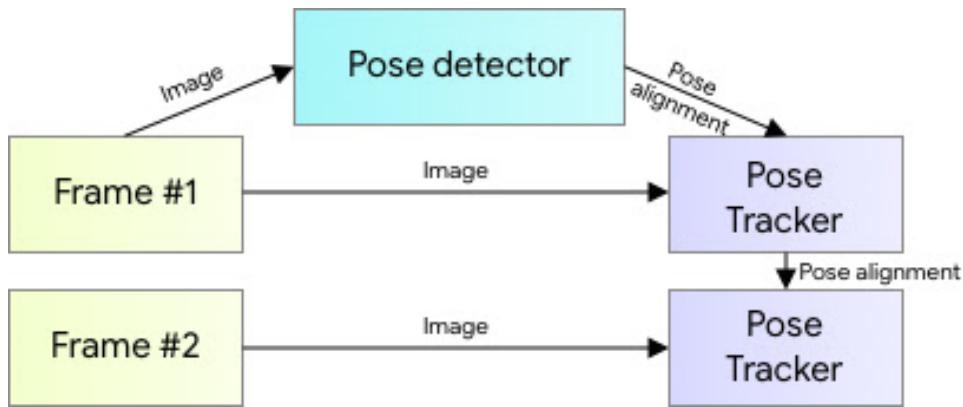


Рис. 2: Структура модели BlazePose для работы в реальном времени.

[16]

Следующим шагом Pose Tracker производит локализацию каждой точки в заданной ROI. Данное действие производится путем комбинированной обработки тепловой карты и данных о смещении с использованием регрессионной модели (см рис. 3). (ВОЗМОЖНО СТОИТ ПОЧИТАТЬ ПРО ТАКУЮ ОБРАБОТКУ И ПОПОДРОБНЕЕ ЕЕ ОПИСАТЬ)

Как можно заметить из рис. 2, при аналитике видеофрагмента Pose Detector используется только на первом кадре, ведь позже данные об интересующей нас области передаются от кадра к кадру. Это упрощает вычисления и позволяет ускорить работу модели в реальном времени.

Развитием данной модели есть ее полное объединение с моделями BlazeFace и BlazeHand в модель Holistic [17]. Она рассматривает намного большее количество точек на лице и ладонях.

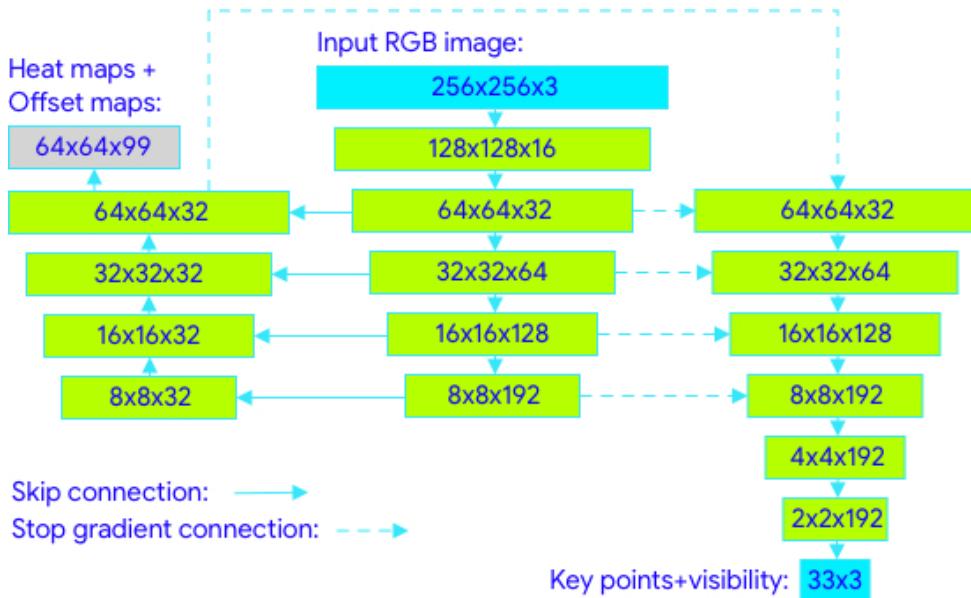


Рис. 3: Архитектура модели Pose Tracker. [16]

3.1.2 MoveNet.SinglePose

SinglePose создана для работы в веб-интерфейсах или на мобильных устройствах в режиме реального времени. [22] Модель представлена в двух спецификациях: lightning и thunder. Первая является менее требовательной в плане мощностей и вычислений и способна обрабатывать до 50 кадров в секунду. В то же время, по заверениям создателей, вторая модель имеет большие запросы по ресурсам, но дает лучшую точность распознавания, правда со скоростью до 30 кадров в секунду.

За расположение ключевых точек выбрана классическая топология COCO. Поэтому модель возвращает координаты 17 точек, которые нормированы на размер изображения (лежат в отрезке $[0, 1]$).

Представленная модель является восходящей, то есть использует bottom-up подход к решению задачи. Она реализована на архитектуре MobileNetV2 [23] с Feature Pyramid Networks [24], которая используется для извлечения признаков. Обработка результатов магистральной части сети происходит с помощью прогнозирующих головок (см. рис. 4), которые используют CenterNet [25] с изменениями, которые повышают

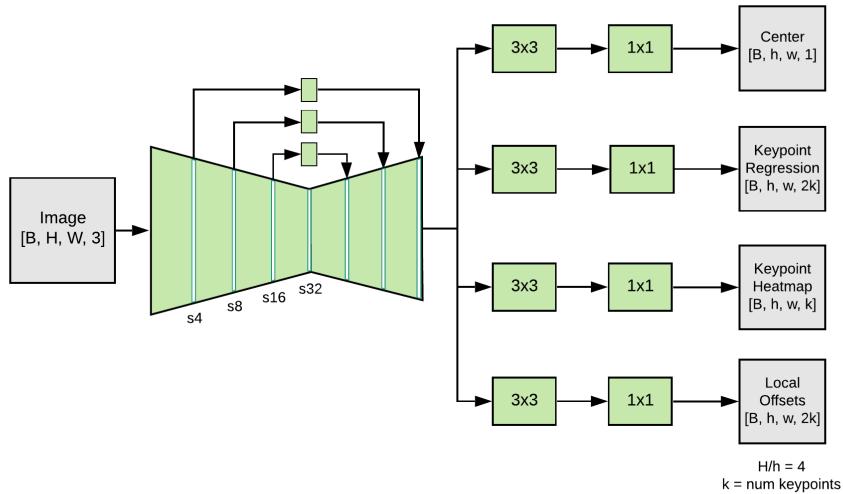


Рис. 4: Архитектура модели MoveNet. [22]

быстродействие модели.

Как можно заметить на рис. 4, 4 прогнозирующие головки отвечают за прогноз карты центральной точки человека, поле регрессионных векторов ключевых точек, карту предсказанных положений ключевых точек и поле смещения ключевых точек. Результаты каждого предсказания вырабатываются параллельно и далее путем последовательной обработки (см. рис. 5) уточняются координаты каждой ключевой точки.

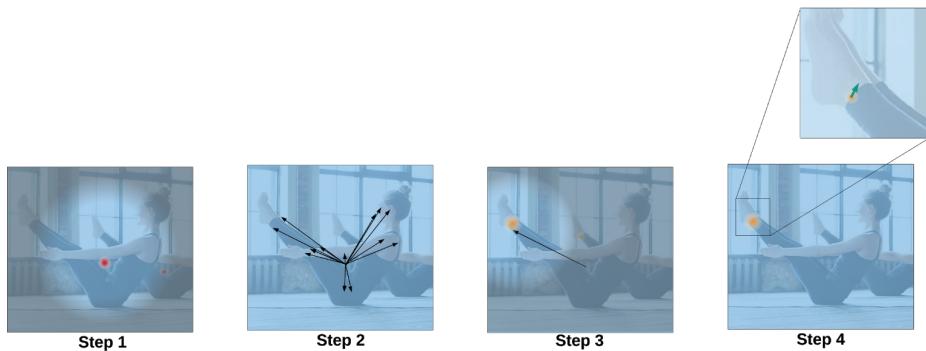


Рис. 5: Шаги работы prediction head в MoveNet. [22]

Такая система позволяет получать точные результаты оценки позы в реальном времени и настроена на обработку в реальном времени изображений, поступающих с камер устройств пользователя.

Следующим этапом развития проекта стала модель MoveNet.MutiPose для распознавания сразу нескольких людей на изображении. Разработчики не стали изменять традициям и представили ее также в двух вариантах: lightning и thunder. А также она использует ту же MobileNetV2 в основе своей работы.

3.1.3 OpenPose

OpenPose (OP) - это подпроект CMU-Perceptual-Computing-Lab из университета Карнеги-Меллона. В нем представлены различные модели: от локализации точного положения кистей и лица, до определения позы, исследуя 135 точек на теле человека. Также OP может распознавать нескольких человек одновременно и поддерживает отслеживание скелета человека на видеозаписи в реальном времени через веб-камеру.

В текущей работе рассмотрим модель, работающую по топологии из 25 точек - чем-то похожую на топологию Halpe (см. разд. 2.1). Особенностью является определение положения стоп за счет детекции 3 дополнительных точек на каждой из них.

OpenPose тоже использует сверточные нейронные сети для решения задачи распознавания ключевых точек с помощью bottom-up подхода. В своей работе исследователи используют понятия карты достоверности обнаружения точки, карты двумерных векторных полей ориентации конечностей (Part Affinity Fields или PAFs) и графы соответствия обнаруженных точек определенным людям на изображении.

Первым шагом происходит обработка изображения и построение карт достоверности и PAFs, про которое поговорим позже. Вторым шагом происходит сопоставление точек и конечностей отдельным людям с использованием графов соответствия. Они помогают решить данную задачу и быстрее решать задачу построения скелетов нескольких человек. В итоге на выходе мы имеем скелеты нескольких человек. Все шаги представле-

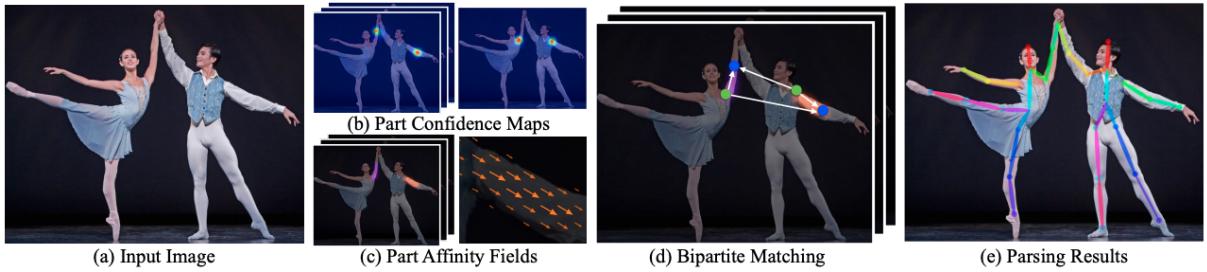


Рис. 6: Последовательность распознавания ключевых точек моделью OpenPose. [8]

ны на рис. 6.

В первоначальной версии модели [26] вычисление карт достоверности и PAFs происходило параллельно, в два этапа. Но в последней работе [8] предсказания поставили последовательно, так как из PAFs интуитивно можно предсказать и уточнить карты достоверности обнаружения точки (см. рис. 7). Ещё были заменены ядра свертки размером 7×7 на блоки из 3-х последовательных ядер 3×3 (см. рис. 7). Эти преобразования помогли почти в 200 раз увеличить скорость распознавания и в 7 раз улучшить точность.

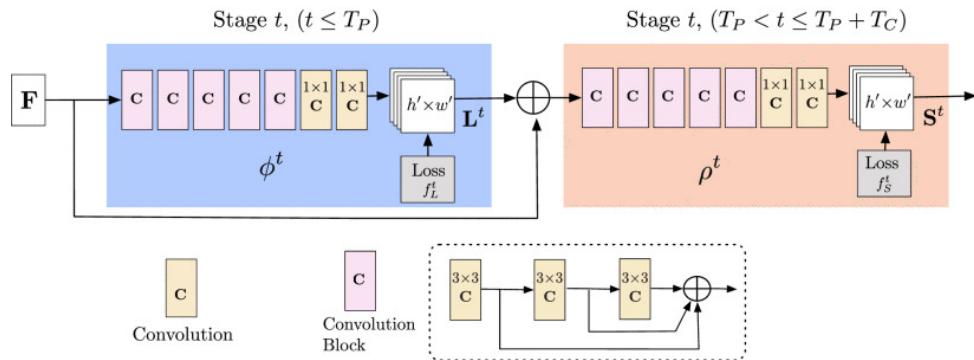


Рис. 7: Архитектура модели OpenPose 2D Pose Estimation. [8]

3.1.4 MMPose

MMPose - является подпроектом лаборатории Open-MMLab [27]. Изначально проект рассматривал решение задачи детектирования объектов, но позже развивались подпроекты, связанные с компьютерным зрени-

ем. Изначально был проект MMSkeleton, основанный на исследовании ST-GCN [28]. Позже появился проект MMpose, включающий в себя распознавание ключевых точек и восстановление скелета человека, модель Animal для тех же задач, но на теле животных. Причем задачи, связанные с человеком, имеют решения с получением предсказаний в 2-х мерном и в 3-х мерном вариантах.

Данная модель постоянно улучшается и подключает новые мировые достижения в свои работы. MMpose использует top-down подход в своей работе и другие наработки лаборатории Open-MMLab. Для работы необходимо сначала использовать detection model, а потом уже передать данные о локализованных людях в pose model.

Детектор натренирован на датасете COCO и выдает предсказания в соответствии с его 17 точечной топологией. Также есть скрипты для обучения на других наборах данных, за что спасибо разработчикам. Правда эти наборы данных надо сначала скачать, но об этом в разд. 4.

3.1.5 AlphaPose

AlphaPose является основной наработкой проекта Machine Vision and Intelligence Group из Шанхайского университета транспорта (Shanghai Jiao Tong University или SJTU). Это первая модель, которая получила значение метрики mAP на датасете COCO выше 70 (0.7) и выше 80 на MPII. Поддерживается как на Linux, так и на Windows. Обрабатывает видео и поддерживает слежение за человеком в реальном времени через восстановление его скелета.

Исследователи не против объединяться с другими проектами для улучшения качества модели и проработки новый фичей. Одной из таких коопераций стала топология Halpe (см. разд. 2.1), на основании которой и производится оценка позы человека. Доступными являются варианты также с 17 точками от COCO и 25 и 135 точек от Halpe.

В работе Alpha использует top-down подход. Первым этапом идет Faster R-CNN для детектирования человека и выдачи прямоугольников. После используется модель RMPE для предсказания различных поз, которые может принимать человек. Последним этапом идет работа p-Pose NMS для устранению избыточных предсказаний. На выходе мы получаем изображение с восстановленными позами людей. Все шаги представлены на рис. 8. [10]

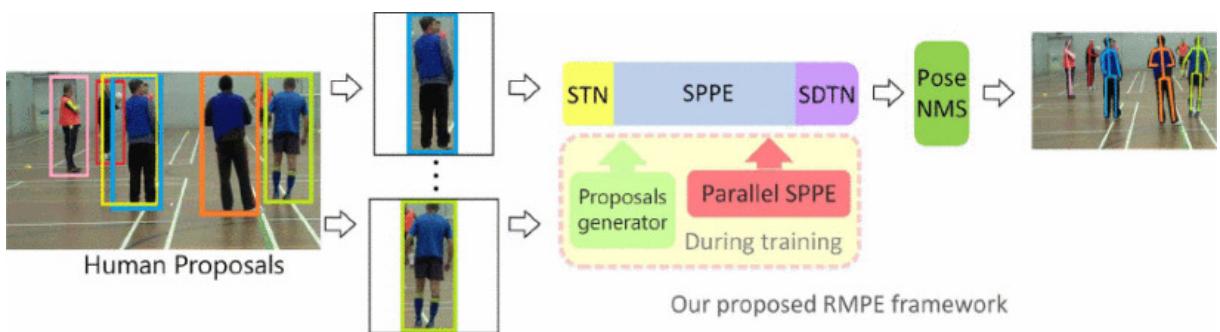


Рис. 8: Пример работы сети AlphaPose. [10]

Рассмотрим поближе модель предсказания. Она использует симметричное преобразование spatial transformer network (STN) и обратное ему spatial de-transformer network (SDTN). Они введены для исправления ошибок локализации, так как SPPE, представленная между ними (см. рис. 9), является очень чувствительной к неточностям прямоугольника. При обучении параллельно основной модели добавляется ещё одна SPPE (см. рис. 9) для корректировки преобразования STN. Таким образом мы приблизим данные, получаемые предсказателем, к идеальным и получим наиболее точный результат.

3.1.6 DeepPose

DeepPose является одним из самых первых решений задачи распознавания ключевых точек на теле человека с помощью глубокого обучения. Статья "DeepPose: Human Pose Estimation via Deep Neural Networks" [29]

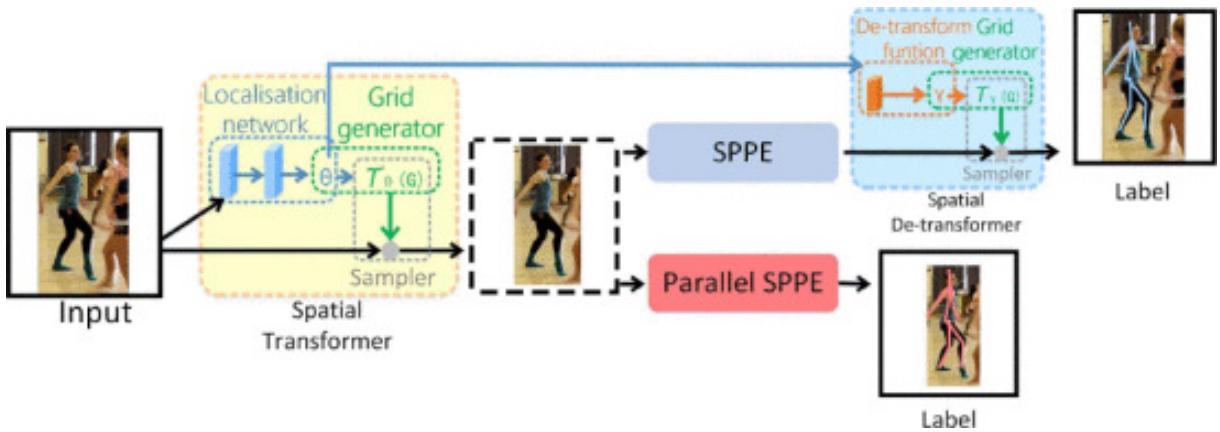


Рис. 9: Архитектура сети RMPE. [10]

была представлена исследователями из GOOGLE на конференции CVPR в 2014 году.

В своей работе они представили каскад из DNN-регрессоров для локализации суставов тела. На тот момент принятых топологий ещё не было и поза кодировалась координатами суставов, нормализованными на размер изображения. Первым этапом применялась CNN для локализации точки, а вторым каскадом применялись DNN для уточнения результата (см. рис. 10). Таким образом получалось довольно точно распознавать ключевые точки на фотографии.



Рис. 10: Архитектура сети DeepPose. [29]

3.2 Модели для классификации позы человека

В данном разделе мы рассмотрим 4 различных модели, которые можно использовать для классификации движений человека.

3.2.1 Классификатор от MediaPipe

Проект представил модель для оценки позы человека (см. разд. 3.1.1) и, добавив к нему k-NN, получили алгоритм классификации позы человека. Таким образом MediaPipe представило программу для счета приседаний или отжиманий с камеры смартфона в реальном времени. [30]

Для работы использовалась топология BlazePose, а точнее вектор расстояний между некоторыми точками этой топологии (см. рис. 11). Так как в реальных изображениях бывают разные масштабы и размеры, то все позы нормализуются и переориентируются в вертикальное положение.

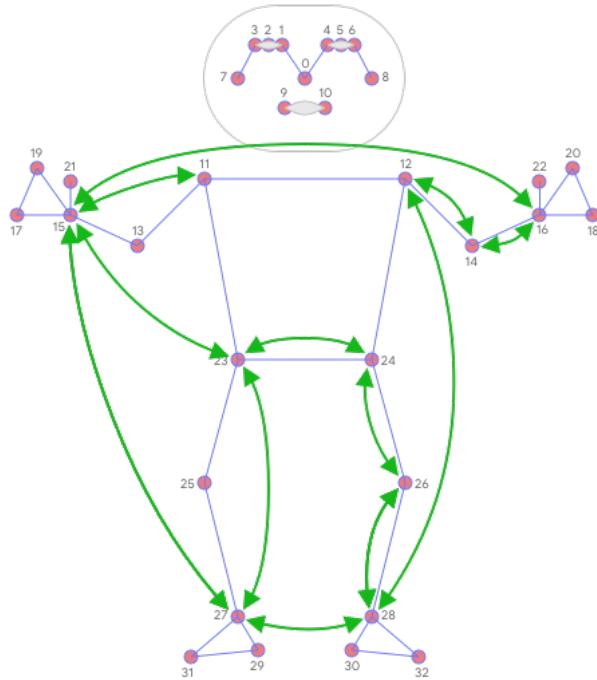


Рис. 11: Расстояния между точками топологии BlazePose, которые используются в классификации. []

Для обеспечения точности классификации разработчики решили запускать k-NN дважды: первый раз для отбора векторов признаков для похожих с целевой поз, а во второй раз для отбора точной позы среди выбранных ранее. Различия в том, что используются различные метрики минимальное покоординатное расстояние и среднее покоординатное расстояния для первого и второго прогона соответственно.

3.2.2 mmakos

Модель является бакалаврской работой студента Варшавского университета Михала Макоса [31]. Классификатор работает с видеоданными и разделяет модели на два супер класса: статические и динамические.

Разработчик взял за основу модель распознавания позы человека OpenPose (см. разд. 3.1.3) и преобразовал входные данные из 4-х мерного формата в 2-мерное изображение (см. рис. 12). По сути три координаты каждой ключевой точки преобразовывались в BGR формат и создавался столбец пикселей для текущего момента времени. При рассмотрении 15 точек и 32 кадров получаются переходные изображения размером (15, 32, 3).

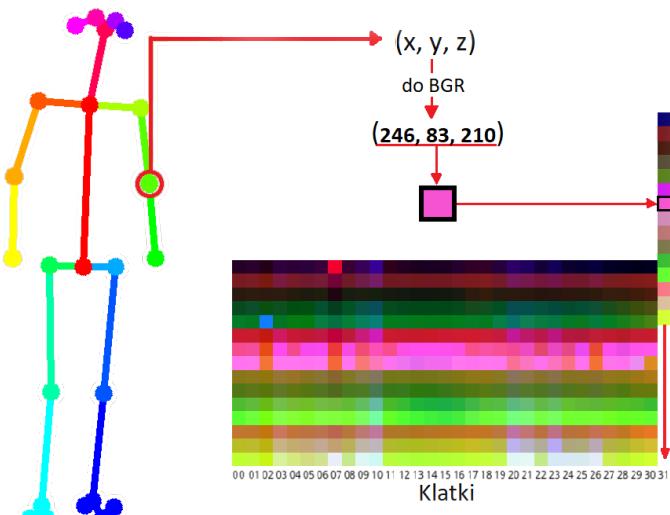


Рис. 12: Преобразование данных в модели от M. Makos. []

Для классификации используется версия сети VGG. По решению автора все позы делятся на два супер-класса: динамические и статические, а потом уже классифицируются внутри супер-классов более точно.

3.2.3 MMaction2 by OpenMMLab

Что-то сложное от китайцев.

4 Исследование моделей

В данной главе будет представлено описание и результаты исследования моделей: BlazePose, MoveNet.SinglePose, OpenPose и MPMpose.

4.1 Описание эксперимента

Эксперимент включает рассмотрение только моделей распознавания ключевых точек на теле человека, так как это является основной частью решения задачи классификации движений и многие классификаторы движений строятся на выходных данных о позе.

Из представленных в разд. 3.1 моделей были выбраны 4 представителя по некоторым критериям:

- Доступность модели для исследований

Необходимо оценить длительность установки и возможности работы с различными операционными системами. Эксперимент проводился на платформе Google Colab, поэтому необходимо было рассмотреть возможность использования модели на в Colaboratory.

- Новизна модели

Представленная выборка была создана в основном в 2010-х, но модель DeepPose является самой старой. Новые разработки опирались на результаты, полученные в ней, и таким образом получали более хорошие результаты.

- Наличие документации

Все модели производят классификацию по двум осям изображения: высота и ширина, а также по параметру видимость ключевой точки. Некоторые модели выдают данные нормированные на размер изображения (число из отрезка $[0,1]$), а некоторые точное значение

в пикселях. Поэтому для работы необходимо было понимать как работает API модели, какие у нее входные - выходные данные.

- Тренировка модели на датасете COCO

Все используемые претренированные модели были обучены на наборе данных COCO [12] в совместительстве с каким-либо другим датасетом. В некоторых примерах не было возможности использовать претренированную модель и из-за этого они были отсечены.

В итоге было выбрано 4 модели наиболее подходящие под критерии:

1. BlazePose
2. MoveNet.SinglePose
3. OpenPose
4. Mmpose

Для проведения качественного анализа и выявления лучшей модели необходимо их сравнить. Поэтому рассмотрим метрики, которые подходят для задач в 2-х мерном пространстве:

- Percentage of Detection Joints

PDJ оценивает точность распознавания ключевой точки в зависимости от диагональных размеров человека. При рассмотрении задачи распознавания человека мы получаем координаты точек, которые характеризуют прямоугольник, внутри которого вписан человек. Диагональ этого прямоугольника используется при вычислении метрики PDJ (см. рис. 13). Формулу можно представить в следующем виде:

$$PDJ = \frac{\sum_{i=1}^n \text{bool}(d_i < \text{threshold} * \text{diag})}{n}, \quad (1)$$

где

d_i - расстояние между предсказанной и правильной точкой,

$threshold$ - порог, задаваемый исследователем,

$diag$ - размер диагонали прямоугольника, внутри которого находится человек,

$bool()$ - логическое условие, возвращает 1, если оно верно и 0 в ином случае,

n - размер выборки.

С помощью значения порога можно варьировать допустимую погрешность расстояния между истинной и предсказанной точками.

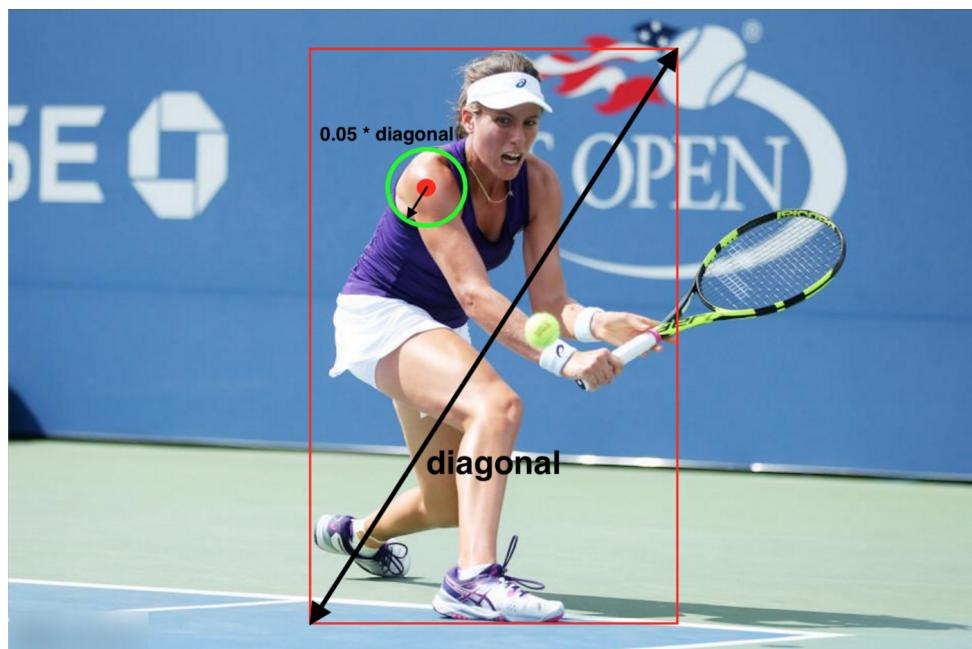


Рис. 13: Визуальное представление метрики PDJ.

Оригинальное изображение

- Percentage of Correct Key-points

ПСК очень похожа на предыдущую метрику, только погрешность рассматривается относительно высоты человека. Формулу можно

представить в следующем виде:

$$PDJ = \frac{\sum_{i=1}^n \text{bool}(d_i < \text{threshold} * \text{bodyheight})}{n}, \quad (2)$$

где

d_i - расстояние между предсказанной и правильной точкой,

threshold - порог, задаваемый исследователем,

bodyheight - высота прямоугольника, внутри которого находится человек,

$\text{bool}()$ - логическое условие, возвращает 1, если оно верно и 0 в ином случае,

n - размер выборки.

- Object Key-point Similarity

OKS является основной при оценке задачи Keypoint Detection COCO [12]. Она использует третью координату выходного предсказания и расстояние между реальной и предсказанной точками. Формулу можно представить в следующем виде:

$$OKS = \frac{\sum_i \exp\left(-d_i^2/2s^2k_i^2\right) \delta(v_i > 0)}{\sum_i \delta(v_i > 0)}, \quad (3)$$

где

d_i - расстояние между предсказанной и правильной точкой,

s - площадь объекта,

k_i - константа ключевой точки, контролирующая спад,

v_i - видимость.

При оценке задачи детекции ключевых точек COCO вводится метрика Average Precision (AP) через OKS. Изменяя границу допустимого значения OKS можно получать различные значения precision и AP.

В метриках РСК и РДЖ необходимо знать размеры прямоугольника, ограничивающего человека. Для этого необходимо использовать модель распознавания объектов, которая будет давать нам эти данные. При работе с метрикой РСК можно обойтись без такой модели, потому что во всех топологиях есть точки, которые обозначают верхнюю и нижнюю границы человека. Отсюда погрешность вычисления через модель и вычисления разности ординат верхней и нижней точек будет мала. Что требует меньше затрат для оценки.

4.2 Поиск данных

Первым делом необходимо было проверить модели на неразмеченных данных. Для этого были выбраны фотографии высокого разрешения, где человек изображен во весь рост (см. рис. 15).

Для качественной оценки работы моделей с помощью метрик необходимо было найти размеченные данные. Приведу описание датасетов, которые были мной рассмотрены:

- COCO Dataset и MPII

Данные наборы являются основными при работе с компьютерным зрением и большинство исследователей используют их как тренировочные для своих моделей. Поэтому использовать их в качестве тестовых не целесообразно. [13, 32]

- HUMAN 3.6M

Данные собирались специально для задачи классификации движений человека в студии. С помощью датчиков фиксировались положения всех суставов и ключевых точек. Это идеальный датасет, но доступ к нему ограничен и создатель не выходит на связь. [1]



(a)



(b)



(c)



(d)

Рис. 14: Изображения для визуальной оценки моделей.

- LSP

Данные собраны со спортивных соревнований и пред обработаны до обозначения одного человека на изображении размером не менее 150 пикселей в высоту. Единственная проблема - не описаны точки на лице, поэтому оценивать можно только распознавание 12 суставов или, другими словами, точек на теле человека. [33]

- LSP Extended

Является расширенной версией предыдущего набора данных. По объёму превосходит его в пять раз. Остальные характеристики не

поменялись. [34]

- Halpe

Датасет для распознавания точек на всем теле человека. К каждой фотографии идет аннотация из 136 точек, краев ограничивающего прямоугольника и категории детектируемого объекта (во всех фотографиях стоит категория человек).

Набор состоит из двух частей: HICO-DET и COCO. Все они аннотированы описанные выше способом. Для наших моделей необходимо будет отобрать только 17 точек, совпадающих с топологией COCO. [35]

Выше были представлены наборы данных для задачи распознавания точек на теле человека. Но основной темой является классификация движений человека, поэтому необходимы фотографии с меткой класса для позы, представленной на данных. Некоторые из уже представленных (COCO, MPII) тоже могут использоваться для классификации позы, но по тем же причинам, что и описаны выше, они не будут рассмотрены в эксперименте. Приведу описание датасетов для классификации движения человека по позе на изображении:

- Stanford-40

Набор данных состоит примерно из 10 тыс. изображений, на каждом из которых человек делает одно из 40 действий. На один класс приходится от 180 до 300 фотографий. [36]

- Yoga-82

Сложность распознавания поз йоги в том, что многие из них не могут быть точно аннотированы. Для решения этой проблемы и был собран этот набор данных вмещающий примерно 28 тыс. изображений. 82 различные позы разделены по 6 подклассам. При желании

можно использовать метки и классов, и подклассов для классификации. [37]

Итого были выбраны наборы данных LSP, LSPE и Halpe.



Рис. 15: Примеры изображений из датасета LSP.

4.3 Результаты эксперимента

Для каждой модели была рассмотрена локализация ключевых точек на фотографиях высокого качества (см. рис. 15), а также набор данных низкого качества с обработанными изображениями.

Метрики были рассчитаны с порогами 0.05, 0.2, 0.5. В дополнение к этому был проведен временной анализ классификации одного изображения в среднем по целому датасету.

Перейдем к результатам по каждой модели.

BlazePose

Результаты работы модели можно посмотреть на рис. 19. Как можно заметить, некоторые точки не распознаются и не отображаются на итоговом результате. А также есть небольшая погрешность при распознавании глаз. Но модель хорошо сработала на человека в маске.

Исходя из значений метрик (см. в табл. 1) можно сделать вывод, что модель имеет высокую погрешность при среднем времени обработки одной фотографии 0.063 секунды.

Metric	PCK@0.05	PCK@0.2	PCK@0.5	PDJ@0.05	PDJ@0.2	PDJ@0.5
LSP	0.035	0.304	0.741	0.042	0.359	0.815
LSPE	0.005	0.059	0.277	0.009	0.108	0.457
Halpe	0.004	0.051	0.222	0.005	0.077	0.313
Metric	AP@0.5	AP@0.75	AP			
LSP	0.012	0.004	0.006			
LSPE	0.03	0.001	0.001			

Таблица 1: Результаты вычисления метрик для BlazePose.

MoveNet.SinglePose

Результаты работы модели можно посмотреть на рис. 20. Заметны ограхи в распознавании локтей и ладоней. Как и в прошлой модели есть неточности при распознавании частей лица.

Исходя из значений метрик (см. в табл. 2) можно сделать вывод, что модель показала себя плохо при оценивании с порогом 0.05, но лучший результат среди всех для порогах 0.2 и 0.5. Среднее время обработки одной фотографии составило 0.009 секунды, что также является наилучшим показателем.

Metric	PCK@0.05	PCK@0.2	PCK@0.5	PDJ@0.05	PDJ@0.2	PDJ@0.5
LSP	0.356	0.928	0.983	0.43	0.95	0.993
LSPE	0.144	0.539	0.745	0.227	0.635	0.806
Halpe	0.04	0.052	0.223	0.006	0.078	0.32
Metric	AP@0.5	AP@0.75	AP			
LSP	0.168	0.064	0.08			
LSPE	0.312	0.059	0.111			

Таблица 2: Результаты вычисления метрик для MoveNet.SinglePose.

OpenPose

Результаты работы модели можно посмотреть на рис. 21.

Значения метрик см. в табл. 3.

Среднее время обработки одной фотографии составило 0.037 секунды.

Визуально лучший результат обработки изображений, хотя по метрикам немного уступает следующей модели. Но полностью обыгрывает ее во времени обработки.

Metric	PCK@0.05	PCK@0.2	PCK@0.5	PDJ@0.05	PDJ@0.2	PDJ@0.5
LSP	0.708	0.837	0.882	0.746	0.845	0.899
LSPE	0.362	0.524	0.633	0.42	0.569	0.69
Halpe	0.575	0.638	0.686	0.613	0.658	0.72

Metric	AP@0.5	AP@0.75	AP
LSP	0.166	0.085	0.092
LSPE	0.538	0.362	0.362
Halpe	0.562	0.398	0.384

Таблица 3: Результаты вычисления метрик для OpenPose.

MMPose

Результаты работы модели можно посмотреть на рис. 22.

Значения метрик см. в табл. 4.

Среднее время обработки одной фотографии составило 0.409 секунды.

Для своей работы требует детекции человека, из-за чего показывает лучший результат в точности, но худший результат по времени обработки одного изображения.

Metric	PCK@0.05	PCK@0.2	PCK@0.5	PDJ@0.05	PDJ@0.2	PDJ@0.5
LSP	0.728	0.85	0.914	0.76	0.858	0.933
LSPE	0.403	0.56	0.651	0.463	0.587	0.72
Halpe						

Metric	AP@0.5	AP@0.75	AP
LSP	0.225	0.223	0.123
LSPE	0.598	0.462	0.443
Halpe			

Таблица 4: Результаты вычисления метрик для MMPose.

Визуализация результатов

На рис. 16 и рис. 17 представлены средние для трех датасетов (LSP, LSPE и Halpe) метрик PCK и PDJ соответственно. Модель BlazePose является явным аутсайдером. OpenPose и MMPose показывают качественные результаты при любых порогах допустимой погрешности. А MoveNet имеет лучшие результаты при большей ошибке.

Результаты растут с увеличением порога, так как увеличивается и допустимая погрешность локализации. Следовательно и количество верных точек тоже становится больше.

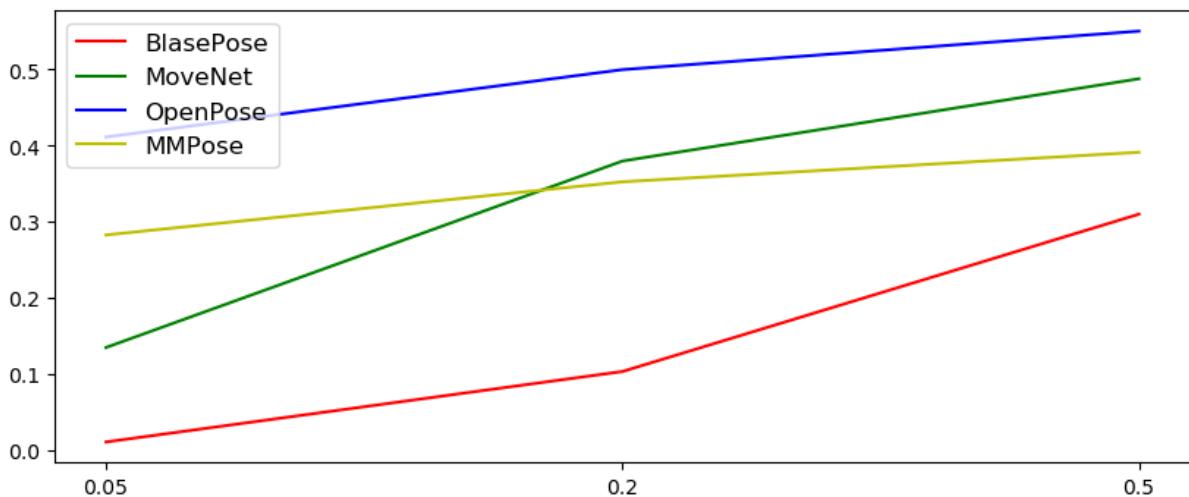


Рис. 16: Средние значения метрики PCK.

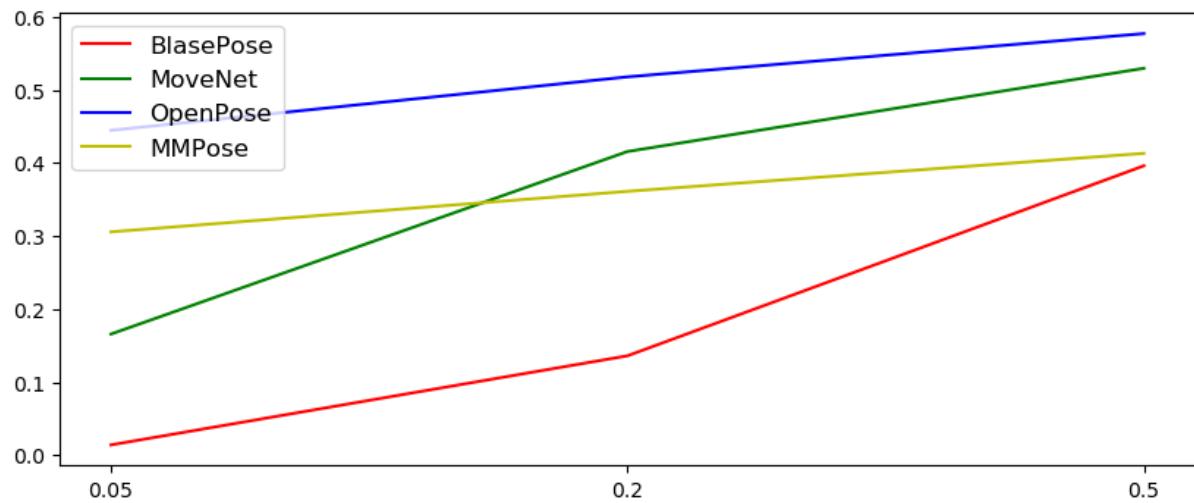


Рис. 17: Средние значения метрики PDJ.

На рис. 18 представлены значения метрики AP также по всем трем наборам данных. Здесь первые две модели (BlazePose и MoveNet) ушли в аутсайдеры из-за плохого результата на изображениях из Halpe, в то время как две другие (OpenPose и MMPose) наоборот выросли за счет этих фотографий.

Результаты данной метрики убывают, так как с увеличением порога метрики растет минимально допустимое значение OKS, которое необходимо для определения верного распознавания.

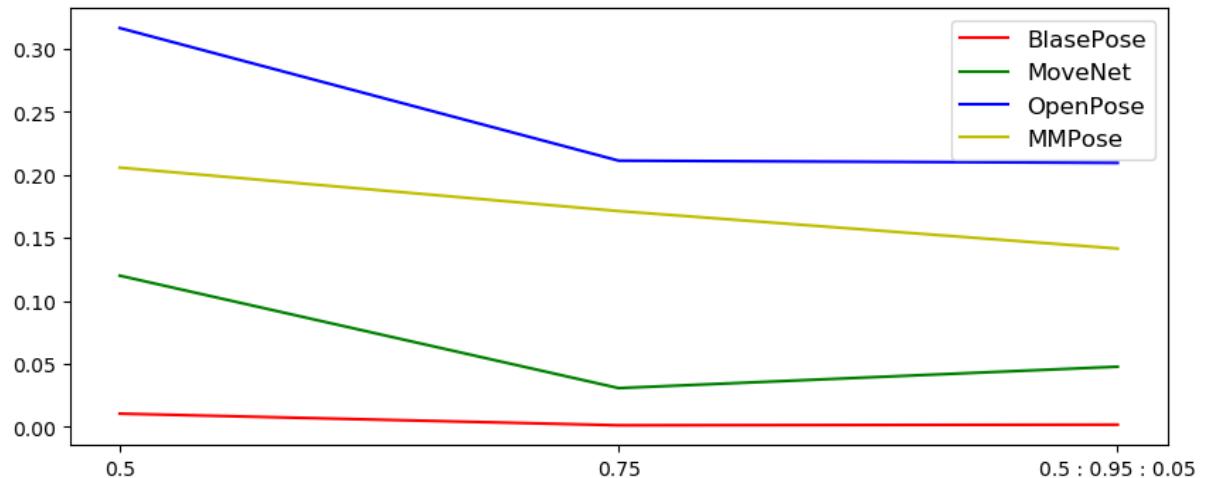


Рис. 18: Средние значения метрики AP.

5 Заключение

Здесь надо перечислить все результаты, полученные в ходе работы. Из текста должно быть понятно, в какой мере решена поставленная задача.

В ходе выполнения работы были проанализированы несколько моделей, решающих задачу распознавания ключевых точек на теле человека. Она является первой частью при анализе позы человека.

Список литературы

- [1] Human3.6M: Large Scale Datasets and Predictive Methods for 3D Human Sensing in Natural Environments / Catalin Ionescu, Dragos Papava, Vlad Olaru, Cristian Sminchisescu // *IEEE Transactions on Pattern Analysis and Machine Intelligence*. — 2014. — jul. — Vol. 36, no. 7. — Pp. 1325–1339.
- [2] Deep 3D human pose estimation: A review / Jinbao Wang, Shujie Tan, Xiantong Zhen et al. // *Computer Vision and Image Understanding*. — 2021. — Vol. 210. — P. 103225. <https://www.sciencedirect.com/science/article/pii/S1077314221000692>.
- [3] Tome, Denis. Lifting from the Deep: Convolutional 3D Pose Estimation from a Single Image / Denis Tome, Chris Russell, Lourdes Agapito // 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). — 2017. — Pp. 5689–5698.
- [4] Detailed, Accurate, Human Shape Estimation From Clothed 3D Scan Sequences / Chao Zhang, Sergi Pujades, Michael J. Black, Gerard Pons-Moll // The IEEE Conference on Computer Vision and Pattern Recognition (CVPR). — 2017. — July.
- [5] Learning from Synthetic Humans / Gül Varol, Javier Romero, Xavier Martin et al. // CVPR. — 2017.
- [6] Whittle, Michael W. Clinical gait analysis: A review / Michael W. Whittle // *Human Movement Science*. — 1996. — Vol. 15, no. 3. — Pp. 369–387. <https://www.sciencedirect.com/science/article/pii/0167945796000061>.
- [7] google.github.io. Pose Classification. — https://google.github.io/mediapipe/solutions/pose_classification.html.

- [8] OpenPose: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields / Z. Cao, G. Hidalgo Martinez, T. Simon et al. // *IEEE Transactions on Pattern Analysis and Machine Intelligence*. — 2019.
- [9] Kocabas, Muhammed. MultiPoseNet: Fast Multi-Person Pose Estimation using Pose Residual Network. — 2018. <https://arxiv.org/abs/1807.04067>.
- [10] RMPE: Regional Multi-person Pose Estimation / Hao-Shu Fang, Shuqin Xie, Yu-Wing Tai, Cewu Lu // ICCV. — 2017.
- [11] Prediction of Human Activities Based on a New Structure of Skeleton Features and Deep Learning Model / Neziha Jaouedi, Francisco J. Perales, José Maria Buades et al. // *Sensors*. — 2020. — Vol. 20, no. 17. <https://www.mdpi.com/1424-8220/20/17/4944>.
- [12] Tsung-Yi Lin Matteo Ruggero Ronchi, Alexander Kirillov. COCO 2020 Keypoint Detection Task. — <https://cocodataset.org/#keypoints-2020>.
- [13] Lin, Tsung-Yi. Microsoft COCO: Common Objects in Context. — 2014. <https://arxiv.org/abs/1405.0312>.
- [14] Bazarevsky, Valentin. BlazeFace: Sub-millisecond Neural Face Detection on Mobile GPUs. — 2019. <https://arxiv.org/abs/1907.05047>.
- [15] Zhang, Fan. MediaPipe Hands: On-device Real-time Hand Tracking. — 2020. <https://arxiv.org/abs/2006.10214>.
- [16] Bazarevsky, Valentin. BlazePose: On-device Real-time Body Pose tracking. — 2020. <https://arxiv.org/abs/2006.10204>.
- [17] Ivan Grishchenko, Valentin Bazarevsky. MediaPipe Holistic — Simultaneous Face, Hand and Pose Prediction, on

- Device. — <https://ai.googleblog.com/2020/12/mediapipe-holistic-simultaneous-face.html>.
- [18] PaStaNet: Toward Human Activity Knowledge Engine / Yong-Lu Li, Liang Xu, Xinpeng Liu et al. // CVPR. — 2020.
- [19] Osokin, Daniil. Real-time 3D Multi-person Pose Estimation Demo. — <https://github.com/Daniil-Osokin/lightweight-human-pose-estimation-3d-demo.pytorch>.
- [20] Saksena, Saumya Kumaar. Human Action Classification. — <https://github.com/dronefreak/human-action-classification>.
- [21] google.github.io. MediaPipe.Home. — <https://google.github.io/mediapipe/>.
- [22] Votel, Ronny. Next-Generation Pose Detection with MoveNet and TensorFlow.js. — <https://blog.tensorflow.org/2021/05/next-generation-pose-detection-with-movenet-and-tensorflowjs.html>.
- [23] Sandler, Mark. MobileNetV2: Inverted Residuals and Linear Bottlenecks. — 2019.
- [24] Lin, Tsung-Yi. Feature Pyramid Networks for Object Detection. — 2016. <https://arxiv.org/abs/1612.03144>.
- [25] Zhou, Xingyi. Objects as Points. — 2019. <https://arxiv.org/abs/1904.07850>.
- [26] Cao, Zhe. Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields. — 2016. <https://arxiv.org/abs/1611.08050>.
- [27] Contributors, MMPose. OpenMMLab Pose Estimation Toolbox and Benchmark. — <https://github.com/open-mmlab/mmpose>. — 2020.

- [28] *Yan, Sijie.* Spatial Temporal Graph Convolutional Networks for Skeleton-Based Action Recognition. — 2018. <https://arxiv.org/abs/1801.07455>.
- [29] *Toshev, Alexander.* DeepPose: Human Pose Estimation via Deep Neural Networks / Alexander Toshev, Christian Szegedy // 2014 IEEE Conference on Computer Vision and Pattern Recognition. — 2014. — Pp. 1653–1660.
- [30] *google.github.io.* Pose Classification. — https://google.github.io/mediapipe/solutions/pose_classification.html.
- [31] *MAKOS, MICHAL.* Human Pose Classification - BEng Thesis. — <https://github.com/mmakos/HPC>.
- [32] 2D Human Pose Estimation: New Benchmark and State of the Art Analysis / Mykhaylo Andriluka, Leonid Pishchulin, Peter Gehler, Bernt Schiele // IEEE Conference on Computer Vision and Pattern Recognition (CVPR). — 2014. — June.
- [33] *Johnson, Sam.* Clustered Pose and Nonlinear Appearance Models for Human Pose Estimation / Sam Johnson, Mark Everingham // Proceedings of the British Machine Vision Conference. — 2010. — doi:10.5244/C.24.12.
- [34] *Johnson, Sam.* Learning Effective Human Pose Estimation from Inaccurate Annotation / Sam Johnson, Mark Everingham // Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. — 2011.
- [35] *Haoshu, Fang.* Halpe Full-Body Human Keypoints and HOI-Det dataset. — <https://github.com/Fang-Haoshu/Halpe-FullBody>.

- [36] Human action recognition by learning bases of action attributes and parts / Bangpeng Yao, Xiaoye Jiang, Aditya Khosla et al. — 2011. — 11. — Pp. 1331–1338.
- [37] Yoga-82: A New Dataset for Fine-grained Classification of Human Poses / Manisha Verma, Sudhakar Kumawat, Yuta Nakashima, Shanmuganathan Raman // IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). — 2020. — Pp. 4472–4479.

A Приложение

Визуальные результаты работы моделей



(a)

(b)



(c)



(d)

Рис. 19: Пример результатов работы модели BlazePose.



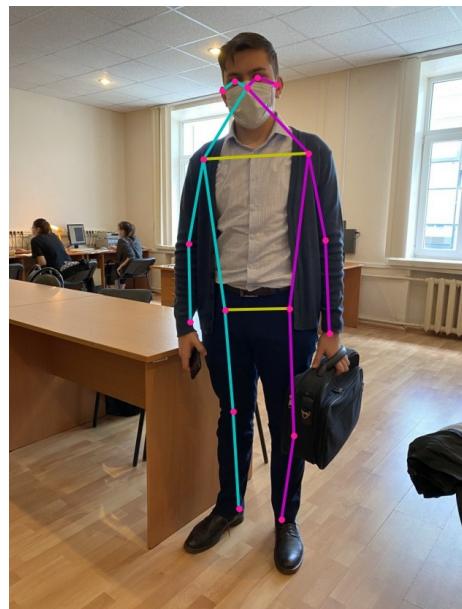
(a)



(b)



(c)



(d)

Рис. 20: Пример результатов работы модели MoveNet.SinglePose.



Рис. 21: Пример результатов работы модели OpenPose.

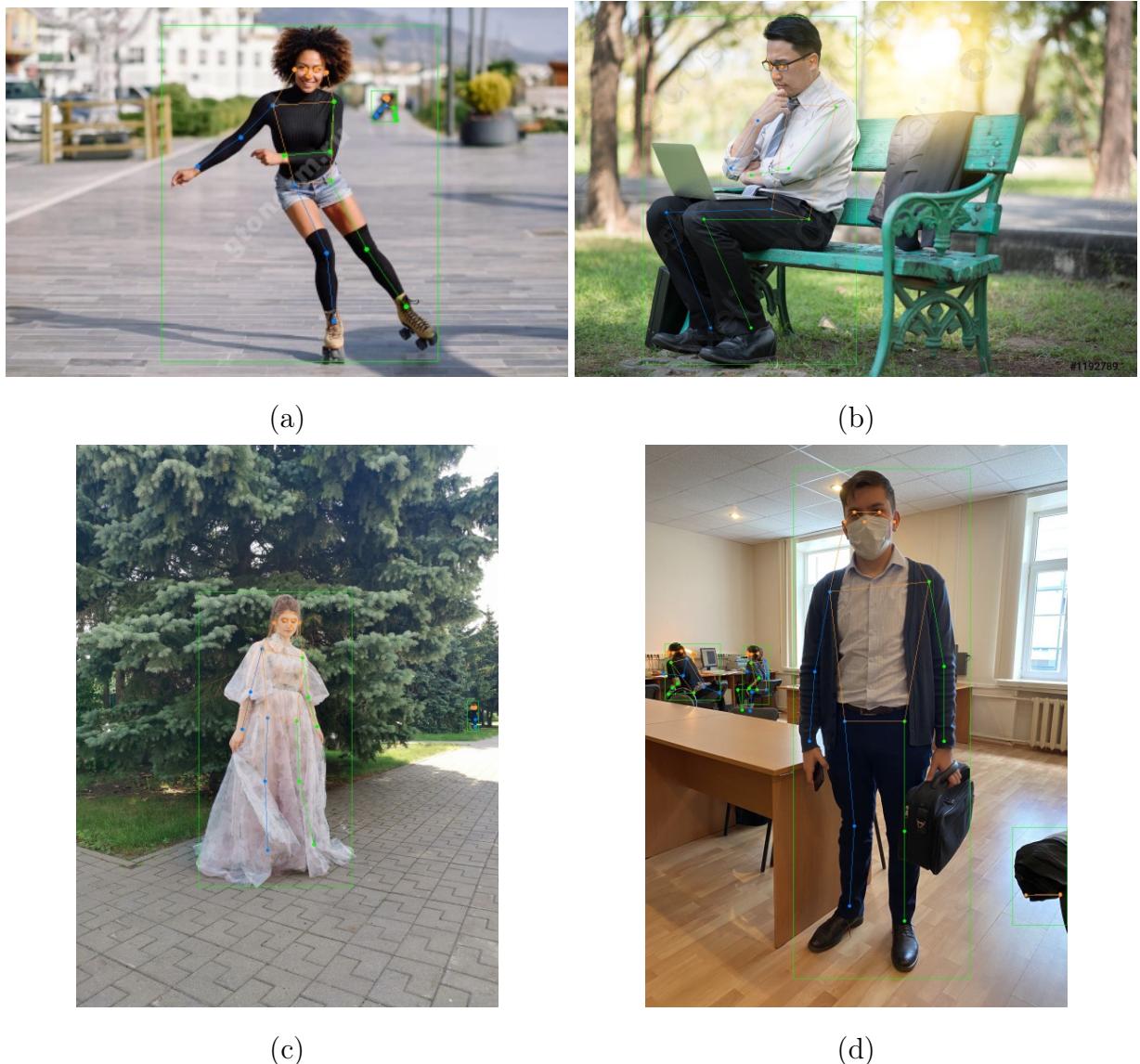


Рис. 22: Пример результатов работы модели MMPose.