

САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Елькина Анна Александровна

Выпускная квалификационная работа бакалавра

**Определение размеров тела человека по
изображениям с использованием нейронных сетей**

Направление 01.03.02 «Прикладная математика и информатика»

Основная образовательная программа СВ.5005.2017 «Прикладная математика,
фундаментальная информатика и программирование»

Профиль «Исследование и проектирование систем управления и обработки
сигналов»

Научный руководитель:
кандидат физ.-мат. наук,
доцент
Корхов В.В.

Рецензент:
исследователь,
ООО «Роберт Бош»
Белошапко А.Г.

Санкт-Петербург

2021

Содержание

Введение	3
Постановка задачи	4
Глава 1. Обзор и анализ существующих решений	5
1.1. Построение 3D-моделей тела человека	5
1.2. Приложение «Sizer».....	9
1.3. Итоги	10
Глава 2. Разработка алгоритма определения размеров тела человека ..	11
2.1. Краткое описание выбранного решения	11
2.2. Требования к входным изображениям	11
2.3. Сегментация листа бумаги.....	12
2.4. Поиск ключевых точек на теле человека	13
2.5. Вычисление размеров тела человека	17
Глава 3. Реализация решения.....	19
3.1. Используемые технологии и средства разработки	19
3.2. Программная реализация	20
Глава 4. Тестирование и анализ полученного решения	22
Выводы	28
Заключение	29
Список литературы	30

Введение

В современном мире покупки в Интернете стали неотъемлемой частью образа жизни большинства людей. Поставщикам одежды бросили вызов в предоставлении индивидуализированных товаров, которые бы точно соответствовали размеру тела определенного потребителя.

Отсюда и возникла проблема, связанная с получением и определением размеров человеческого тела напрямую через Интернет.

Обычно размеры человеческого тела можно измерять вручную или автоматически. Благодаря преимуществам интуитивно понятных и удобных инструментов ручное измерение с помощью лент уже много лет применяется в качестве традиционного метода сбора данных о человеческом теле. Однако, поскольку метод во многом зависит от опыта и суждений замерщиков, точность полученных данных ненадежна, что может легко привести к проблеме плохой посадки одежды. Кроме того, на это уходит много времени.

По сравнению с ручным измерением, эффективность и точность были значительно улучшены благодаря технологии 3D-сканирования человеческого тела. Однако, на сегодняшний день в контексте продаж через Интернет нереально использовать оба вышеупомянутых метода измерения.

Таким образом, возникает необходимость создания более простого способа решения данной задачи.

В связи с быстрым развитием направления искусственного интеллекта в настоящее время, определение размеров человеческого тела с помощью нейронных сетей, а не их физическое измерение, привлекает все больше и больше внимания в индустрии одежды. Соответственно, создание алгоритма, способного определить размеры человеческого тела по изображениям с использованием нейронных сетей, нашло бы высокий спрос и широкое применение во многих прикладных областях.

Постановка задачи

Целью данной научно-исследовательской работы является разработка и реализация алгоритма определения размеров тела человека по изображениям с использованием нейронных сетей.

Для достижения поставленной цели были сформулированы следующие задачи:

- Ознакомиться с существующими решениями определения размеров тела человека и проанализировать их;
- Выбрать модель, позволяющую определить размеры тела человека с наиболее высокой практической пользой и простотой;
- Реализовать предварительную обработку входного изображения, на основе которого будут производиться вычисления;
- Выбрать архитектуру и реализовать программу на языке Python, позволяющую определить размеры тела человека;
- Провести тестирование и анализ реализованного решения.

Глава 1. Обзор и анализ существующих решений

В первую очередь, рассмотрим и проанализируем существующие на данный момент решения, используемые для определения размеров тела человека по изображениям.

1.1. Построение 3D-моделей тела человека

Большинство существующих подходов к решению задачи основаны на построении 3D-модели тела человека. В некоторых современных литературных работах делается попытка представить эффективный и полностью автоматизированный метод определения размеров тела, но большинство из них изначально запрашивают большой объем информации, который необходимо определять специалистом [1; 2].

Рассмотрим более подробно метод, представленный в статье [2]. Предлагаемое в данной работе решение представляет собой автоматическое извлечение признаков человеческого тела и измерение размеров путем регрессионного анализа геодезических расстояний. Геодезическое расстояние - это кратчайший путь между двумя точками на трехмерной поверхности, который аналогичен прямой линии между двумя точками на плоскости.

Прежде всего для получения 3D-модели человеческого тела используется камера глубины. Затем для извлечения предопределенных характерных точек и линий используется метод автоматического извлечения фокальных элементов на трехмерном теле человека с помощью регрессионного анализа геодезических расстояний случайного леса. В конечном итоге, индивидуальный размер человеческого тела рассчитывается в соответствии с этими характерными точками и линиями. Пример работы данного метода представлен на Рисунке 1.

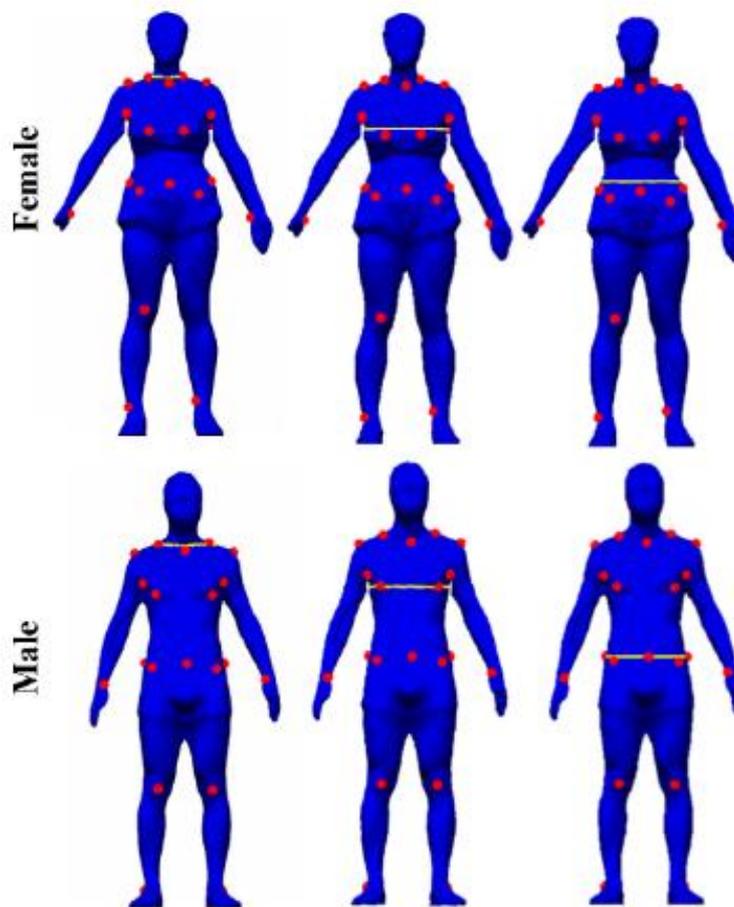


Рисунок 1 - Пример работы метода, представленного в статье [2]

Таким образом, полученный метод является нечувствительным к позам и различным формам трехмерного человеческого тела. Эти основные преимущества приводят к надежному и точному извлечению признаков и измерению размеров трехмерных человеческих тел в различных положениях и формах.

Тем не менее, данный метод всё же имеет существенные недостатки, к примеру, такие как эффективность вычислений, что делает его непригодным к работе в условиях реального времени.

Далее рассмотрим работу [3], где авторы решили ограничиться построением 3D-модели тела человека по одной стандартной позе. Это проявляется как в самом методе реконструкции, так и в проведенных ими проверочных экспериментах. Пример работы алгоритма представлен на Рисунке 2.

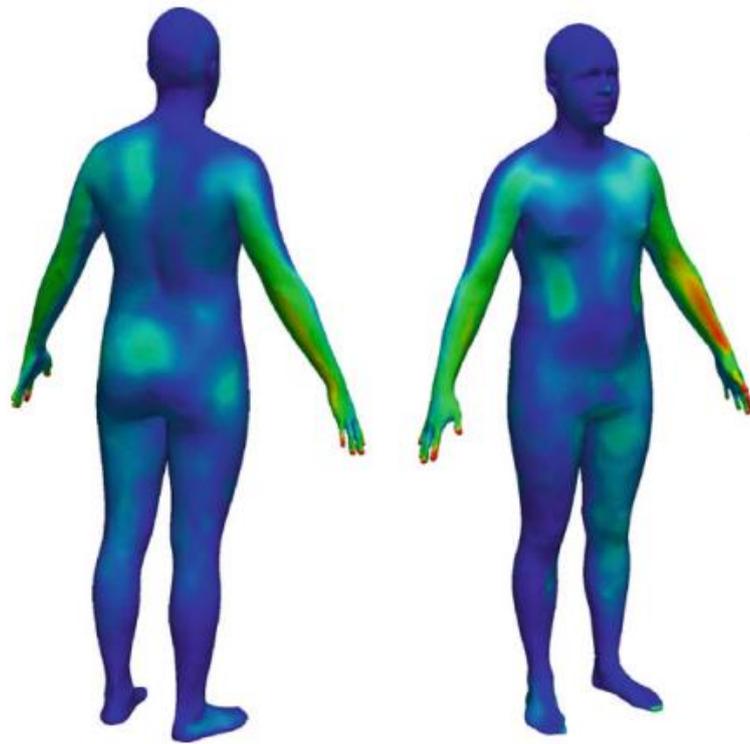


Рисунок 2 - Пример работы метода, приведенного в статье [3]

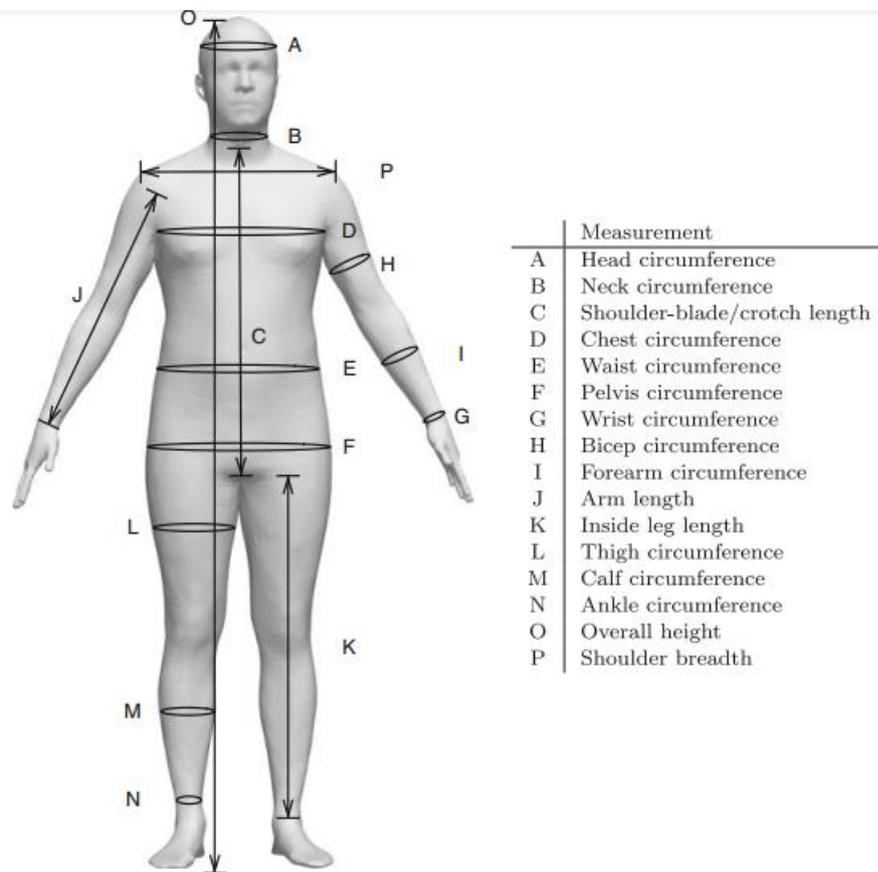


Рисунок 3 - Трехмерные измерения, используемые в процессе валидации

Решение строить 3D-модель тела человека только по одной возможной позе уменьшило количество параметров, которые необходимо было учитывать, что сделало эксперименты удобными. Однако этот выбор привёл к ошибкам в реконструкции. Несмотря на то, что испытуемых просили принять стандартное положение, всегда были некоторые вариации в их позах. Это означает, что в модель изменчивости формы интегрирована некоторая изменчивость, что снижает ее предсказательную силу.

В статьях [4; 5; 6] авторы предлагают оценивать форму тела под одеждой, подгоняя трехмерную модель тела к трехмерным реконструкциям людей. Эти методы не могут быть хорошо распространены на многослойную или свободную(оверсайз) одежду без дополнительного предварительного контроля [7]. В [8; 9; 10; 11; 12; 13] авторы представляют к изучению модели для оценки формы тела по 2D-изображениям людей в одежде, но точность получаемой формы ограничена из-за неоднозначности глубины. К тому же, все данные методы выполняют только реконструкцию 3D-модели человека и не предусматривают вычисление размеров тела. Примеры построения 3D-моделей тела человека представлены на Рисунке 4.



Рисунок 4 - Примеры построения 3D-моделей тела человека

1.2. Приложение «Sizer»

На сегодняшний день одним из самых известных на рынке приложений для определения размеров тела человека является приложение «Sizer» [14].

Метод, использованный в данном приложении, предлагает подход к прогнозированию измерений тела с использованием цифровой обработки изображений, сверточной нейронной сети dense human pose estimation (DensePose) [15] и методов машинного обучения для выполнения сегментации и, наконец, проведения измерений.

Авторы сосредоточены на прогнозировании размеров человеческого тела на 2D-изображениях и создании концепции сегментации тела с использованием выходных функций DensePose в качестве входных данных для классификаторов, выполняющих сегментацию тела.

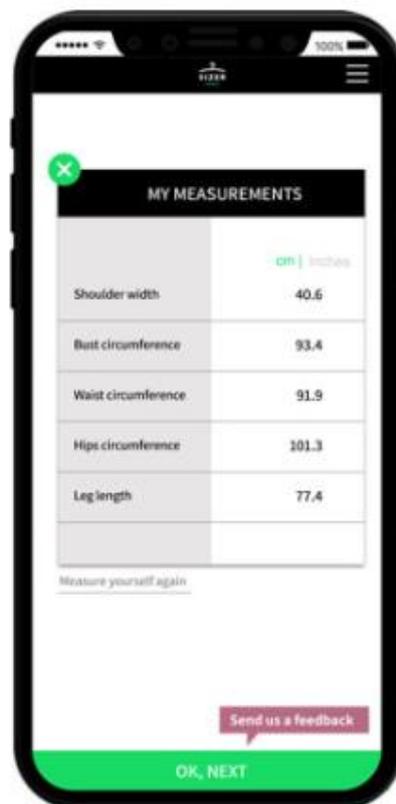
Сверточная нейронная сеть оценки позы человека запускает процесс инициализации данного подхода, вычисляя размеры человеческого тела при помощи изображений, полученных с различных перспектив. После этого, метод слияния соединяет четыре изображения тела и производится измерение ширины плеч, объема груди, объема талии, объема бедер и длины ног.

В конечном итоге, данный метод был адаптирован и интегрирован в мобильное приложение под названием «Sizer». Пользователю предлагается повторить в соответствии с инструкцией несколько простых поз во время взаимодействия с приложением, в течение которых собирается серия изображений пользователя с разных перспектив, на основе которых алгоритм поучает контуры тела и производит вычисления.

Пример работы приложения «Sizer» представлен на Рисунке 5.



GETTING MEASURED



RESULTS

Рисунок 5 - Результаты работы приложения «Sizer»

К недостаткам данного решения можно отнести долгое время вычислений и довольно большие погрешности при сканировании человека в свободной одежде.

1.3. Итоги

Исходя из исследования существующих решений, было выявлено, что автоматическое определение размеров тела человека может быть затруднено по нескольким причинам: во-первых, необходимые для вычислений ключевые точки на теле человека не всегда располагаются на видном месте; во-вторых, гарантировать надежность вычислений достаточно проблематично, когда позы людей или их типы телосложения различаются; в-третьих, самым существенным недостатком является то, что большинство существующих алгоритмов не приспособлены к быстрой работе в реальном времени.

Глава 2. Разработка алгоритма определения размеров тела человека

2.1. Краткое описание выбранного решения

В ходе анализа преимуществ и недостатков существующих подходов, было принято решение предложить собственный метод определения размеров тела человека с использованием нейронных сетей, который был бы приспособлен к работе в реальном времени без потери вычислительной эффективности.

Суть алгоритма заключается в следующем: на вход программа получает изображение человека рядом с листом белой бумаги и ширину этого листа в сантиметрах. Затем при помощи методов сегментации выделяем границы листа, вычисляем его высоту, и благодаря этому получаем соотношение пиксель/см (какому количеству сантиметров соответствует 1 пиксель) в длину и ширину. Далее при помощи модели OpenPose [16] и нейронных сетей выделяем на теле человека необходимые ключевые точки, после чего вычисляем нужные нам значения/расстояния между найденными точками в пикселях и переводим их в сантиметры. На основе полученных значений и таблиц размеров выводим рекомендованный для данного человека размер одежды.

2.2. Требования к входным изображениям

Качественное входное изображение человеческого тела очень важно для методологии измерения, описанной в данной работе. Для успешной работы алгоритма необходимо сделать фотографию в перспективе спереди человека, стоящего рядом с листом белой бумаги на одинаковом расстоянии от камеры. Телефон необходимо расположить на уровне бёдер перпендикулярно поверхности, на которой он находится. Образец нужного изображения человеческого тела представлен на Рисунке 6.



Рисунок 6 - Образец входного изображения

Также необходимо сообщить ширину используемого листа, чтобы выяснить корреляцию между шкалой пиксель / сантиметр (пиксель на сантиметр) (Стандартная ширина листа A4 = 21см).

Окружающая среда должна содержать качественное естественное или искусственное освещение, а также иметь однородный цветной фон. Одежда человека должна быть как можно более облегающей/открывающей участки кожи, чтобы она не мешала вычислению ключевых точек на теле. Человек должен смотреть прямо перед собой.

2.3. Сегментация листа бумаги

После получения изображения следующая задача - сегментировать лист бумаги. Для этого мы используем методы детектирования границ из библиотеки OpenCV [17]. На вход алгоритма подается полутоновое изображение, для его получения был использован метод `cvtColor` библиотеки OpenCV. В качестве результата возвращаются границы сегментированного листа бумаги, а также его ширина и высота. Пример результата сегментирования показан на Рисунке 7.



Рисунок 7 - Результат сегментирования листа бумаги

2.4. Поиск ключевых точек на теле человека

После сегментации и получения контура листа бумаги, перед вычислением размеров тела человека, нам необходимо получить ключевые точки его тела. Для решения этой задачи будем использовать модель OpenPose для оценки позы человеческого тела. Рассматривался модуль DNN (Deep neural networks) [18] библиотеки OpenCV.

Проект распознавания позы человека OpenPose - это библиотека с открытым исходным кодом, разработанная Университетом Карнеги-Меллона на основе сверточной нейронной сети. Это первое в мире приложение для оценки позы человека в реальном времени, основанное на глубоком обучении. В нашей задаче будем использовать предварительно обученную модель Caffe, которая выиграла конкурс ключевых точек COCO в 2016 году.

До недавнего времени прогресс в оценке позы человека на изображениях был незначительным из-за отсутствия высококачественных наборов данных. Но за последние несколько лет были выпущены несколько объемных наборов данных, которые упростили исследователям задачу сбора и разметки необходимых изображений.

Один из таких наборов данных: COCO Keypoints challenge.

Датасет COCO [19] - очень известный и “свежий” датасет. Последняя его версия, предназначенная для решения задач Computer Vision, была опубликована в 2019 году. Она содержит 80 различных категорий данных для задач распознавания и детекции объектов и 91 категорию для задач семантической сегментации. Суммарно датасет содержит 330 тысяч различных изображений.

Наборы для обучения, проверки и тестирования COCO, содержащие более 200 000 изображений и 250 000 человек, помеченных ключевыми точками (большинство людей в COCO в средних и крупных масштабах), доступны для загрузки. Аннотации по обучению и проверке (более 150 000 человек и 1,7 миллиона помеченных ключевых точек) также общедоступны.

Архитектура модели OpenPose представлена на Рисунке 8:

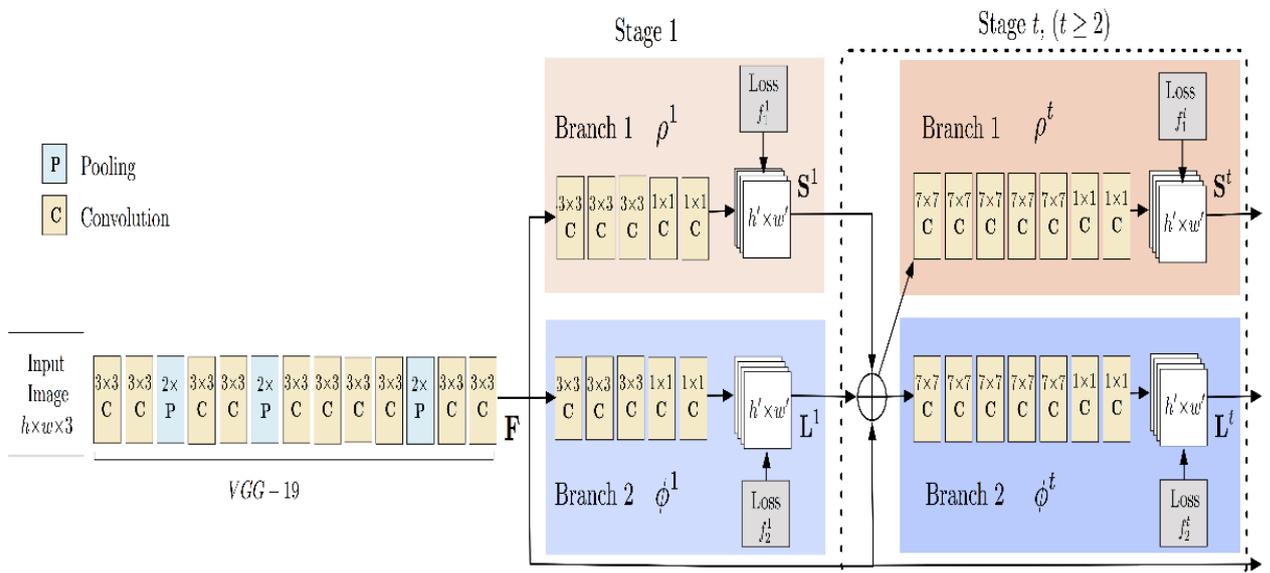


Рисунок 8 - Архитектура модели OpenPose

Модель принимает в качестве входных данных цветное изображение размером $w \times h$ и представляет в качестве выходных данных двухмерные местоположения ключевых точек тела человека на изображении.

Обнаружение ключевых точек тела человека на изображении происходит в три этапа:

1. Используем первые 10 слоев VGGNet для создания карт функций для входного изображения. VGG – это сверточная нейронная сеть с глубиной 19 слоев. Она обучена с использованием более одного миллиона изображений из базы данных ImageNet. Сеть обучалась на цветных изображениях размером 224x224 пикселей.

2. Используем многоступенчатую сверточную нейронную сеть с двумя ветвями, где первая ветвь предсказывает набор двумерных карт достоверности местоположений частей тела (например, локтя, колена и т.д.). Каждая карта достоверности представляет определенную часть схемы скелета позы человека. На Рисунке 9 приведен пример карты достоверности для одной из ключевых точек - левого плеча.

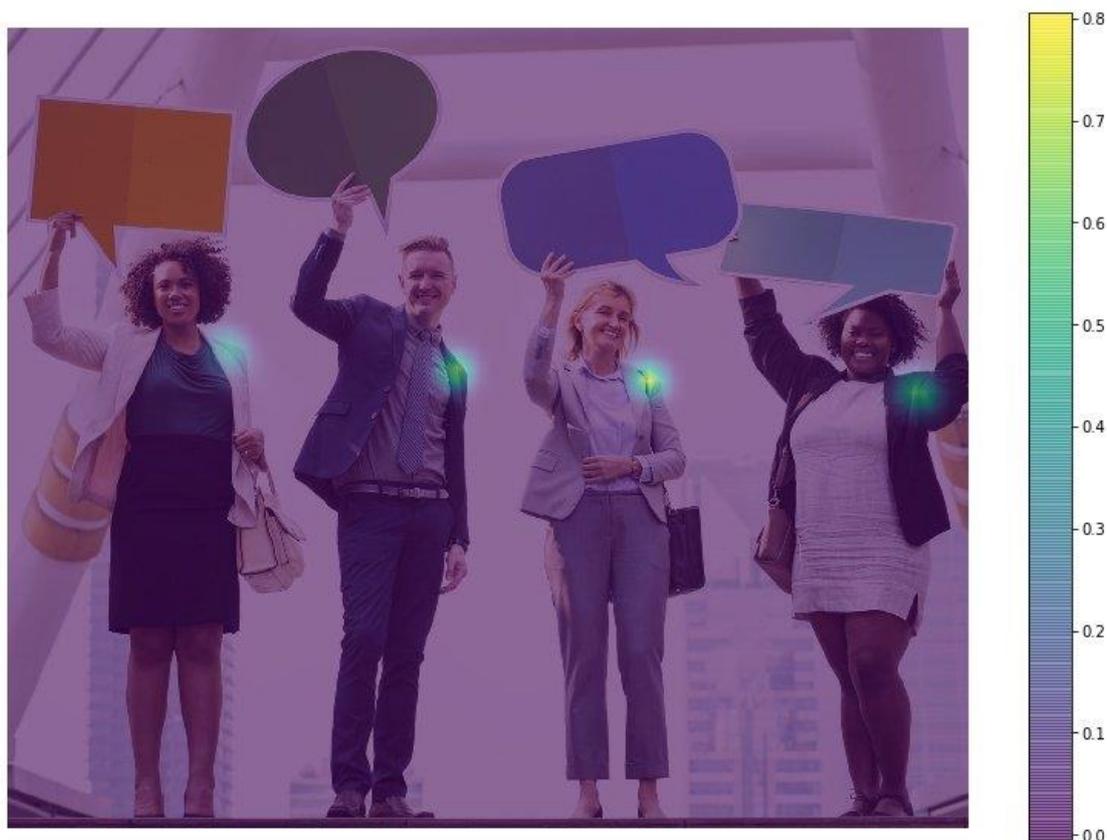


Рисунок 9 - Карта достоверности для левого плеча

Вторая ветвь предсказывает набор двумерных векторных полей сродства частей, которые кодируют степень ассоциации между частями. На Рисунке 10 показано сродство части шеи и левого плеча.

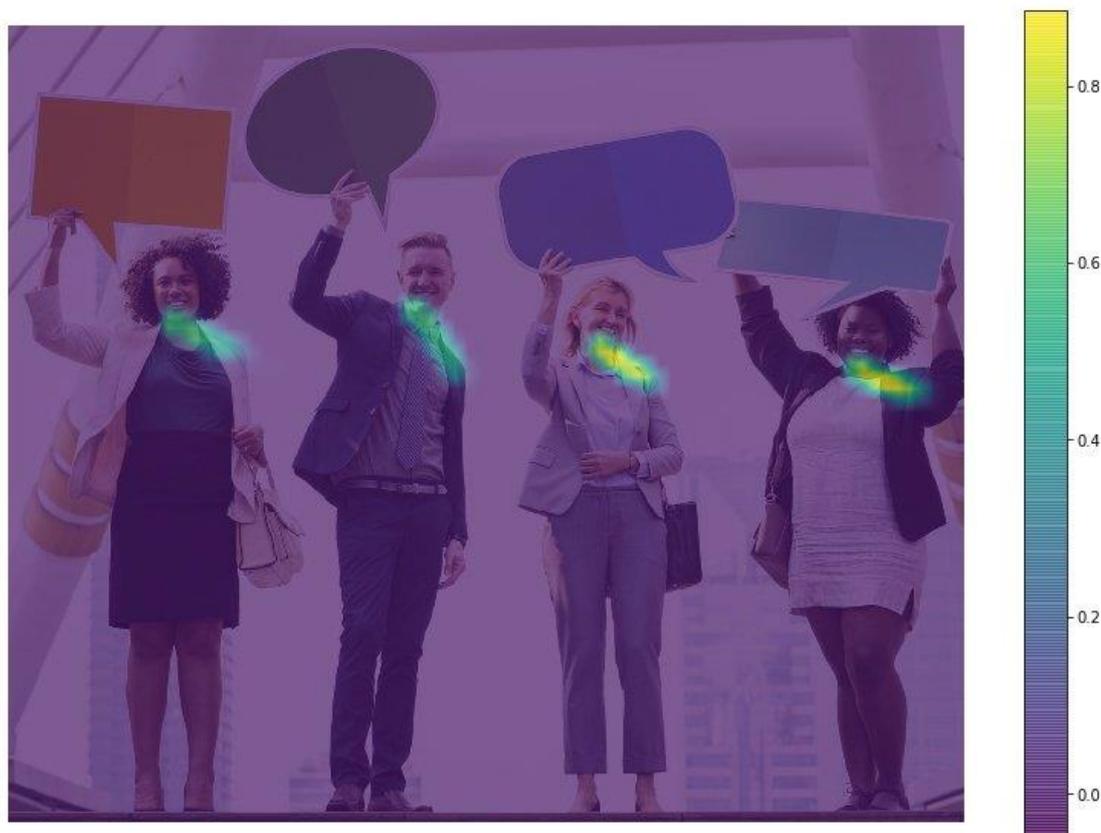


Рисунок 10 - Карта сродства левого плеча и части шеи

3. Карты достоверности и сродства анализируются жадными алгоритмами, чтобы получить двумерные ключевые точки для человека на изображении.

На выходе мы получаем 18 ключевых точек.

Результаты работы модели, нанесенные на человека, представлены на Рисунке 11.



Рисунок 11 - Результаты работы СОСО

Формат вывода точек: нос - 0, шея - 1, правое плечо - 2, правый локоть - 3, правое запястье - 4, левое плечо - 5, левый локоть - 6, левое запястье - 7, правое бедро - 8, правое колено - 9, правая лодыжка - 10, левое бедро - 11, левое колено - 12, лодыжка - 13, правый глаз - 14, левый глаз - 15, правое ухо - 16, левое ухо - 17, задний план - 18

Для получения необходимых в данной работе результатов нам понадобятся только точки 1-13.

2.5. Вычисление размеров тела человека

Далее мы будем использовать полученные ключевые точки для сопоставления изображений вместе с шириной листа бумаги в пикселях для определения масштаба пиксель / сантиметр.

Искомые значения (в сантиметрах) длины рук, длины туловища и длины ног были получены исходя из Формулы 1:

$$scale_height_{px}^{cm} = \frac{height_paper^{cm}}{height_paper^{px}} * dist(point_i, point_j)^{px} \quad (1).$$

Значение (в сантиметрах) ширины плеч по Формуле 2:

$$scale_width_{px}^{cm} = \frac{width_paper^{cm}}{width_paper^{px}} * dist(point_i, point_j)^{px} \quad (2),$$

где $width_paper^{px/cm}$ - ширина листа бумаги в пикселях/сантиметрах;
 $height_paper^{px/cm}$ - высота листа бумаги в пикселях/сантиметрах;
 $dist(point_i, point_j)^{px}$ - расстояние между двумя нужными для вычисления ключевыми точками в пикселях.

Далее, исходя из полученных алгоритмом значений и таблиц размеров одежды, представленных на Рисунках 12 и 13, выводим рекомендованный для данного человека размер футболки и штанов.



Рисунок 12 - Размеры футболки



Рисунок 13 - Размеры штанов

Глава 3. Реализация решения

3.1. Используемые технологии и средства разработки

В качестве инструмента решения задачи будет использован высокоуровневый язык программирования общего назначения Python. Данный язык славится большим количеством библиотек, которые легко устанавливаются. Благодаря им, проводить различного рода исследования на этом языке значительно проще.

В данной задаче были использованы следующие библиотеки:

- OpenCV - библиотека для работы с изображением и видео. Весьма гибкая библиотека, предоставляющая удобные инструменты для работы в области Computer Vision. Имеет множество простых алгоритмов для работы с изображениями.
- Библиотека NumPy предоставляет общие математические и числовые операции, базовые методы для обработки больших массивов и матриц.
- SciPy - включает средства для обработки числовых последовательностей, лежащих в основе моделей машинного обучения: интеграции, экстраполяции, оптимизации и т.д.
- Библиотека argparse - для обработки аргументов (параметров, ключей) командной строки.
- imutils – библиотека с рядом удобных функций для упрощения основных функций обработки изображений, таких как перевод, вращение, изменение размера, скелетонизация, отображение изображений Matplotlib, сортировка контуров, обнаружение краев и т.д.
- math – один из важнейших модулей в Python, который предоставляет пользователю обширный функционал для работы с числами.

3.2. Программная реализация

С исходным кодом работы можно ознакомиться по следующей ссылке:

github.com/alunet/diploma_bodysize

Перед началом работы необходимо загрузить файлы кода и модели.

Используем файл `getModels.sh`, чтобы загрузить веса модели в папку, где расположен код. Далее необходимо проверить эту папку и убедиться, что все файлы модели (файлы `.caffemodel`) были загружены.

В работе используются модели, обученные в Caffe Deep Learning Framework.

У моделей Caffe есть 2 файла:

- Файл `.prototxt`, который определяет архитектуру нейронной сети - расположение различных слоев и так далее;
- Файл `.caffemodel`, в котором хранятся веса обученной модели.

Воспользуемся обоими файлами для загрузки сети в память.

Входной кадр, который мы читаем с помощью OpenCV, должен быть преобразован во входной BLOB-объект (например, Caffe), чтобы его можно было передать в сеть. Это делается при помощи функции `blobFromImage`, которая преобразует изображение из формата OpenCV в формат blob Caffe.

Как только изображение передается в модель, можно выполнять поиск и выделение ключевых точек. Метод пересылки для класса DNN в OpenCV выполняет прямой проход по сети, что является еще одним способом сказать, что он делает прогноз.

На выходе получаем 4-х мерную матрицу:

- Первое измерение - это идентификатор изображения (в случае, если в сеть передается более одного изображения).

- Второе измерение определяет индекс ключевой точки. Модель создает карты достоверности и сопоставления. Для модели СОСО она состоит из 57 частей - 18 карт достоверности ключевых точек + 1 фон + 19 * 2 карт соответствия частей. Мы будем использовать только первые несколько точек, необходимые для вычисления в исходной задаче.
- Третье измерение - это высота выходной карты.
- Четвертое измерение - это ширина выходной карты.

Далее идёт проверка, присутствует ли каждая ключевая точка на изображении или нет.

После этого местоположение каждой ключевой точки можно найти, посмотрев максимальное значение карты достоверности, воспользовавшись пороговым значением, чтобы уменьшить количество ложных срабатываний.

Как только ключевые точки обнаружены, мы просто наносим их на изображение, после этого производим все необходимые вычисления расстояний и выводим результат.

Пример запуска скрипта программы в командной строке:

```
Python full_code.py --image [image_name] --width [width]
```

Параметр `--image` принимает путь до директории в которой хранится изображение. Изображения имеют расширение «`jrg`».

Параметр `--width` – ширина (в сантиметрах) листа бумаги, использованного на изображении.

Глава 4. Тестирование и анализ полученного решения

На Рисунках 14, 15, 16 и 17 приведено несколько примеров работы алгоритма:



Рисунок 14 - Результаты работы алгоритма на модели 1

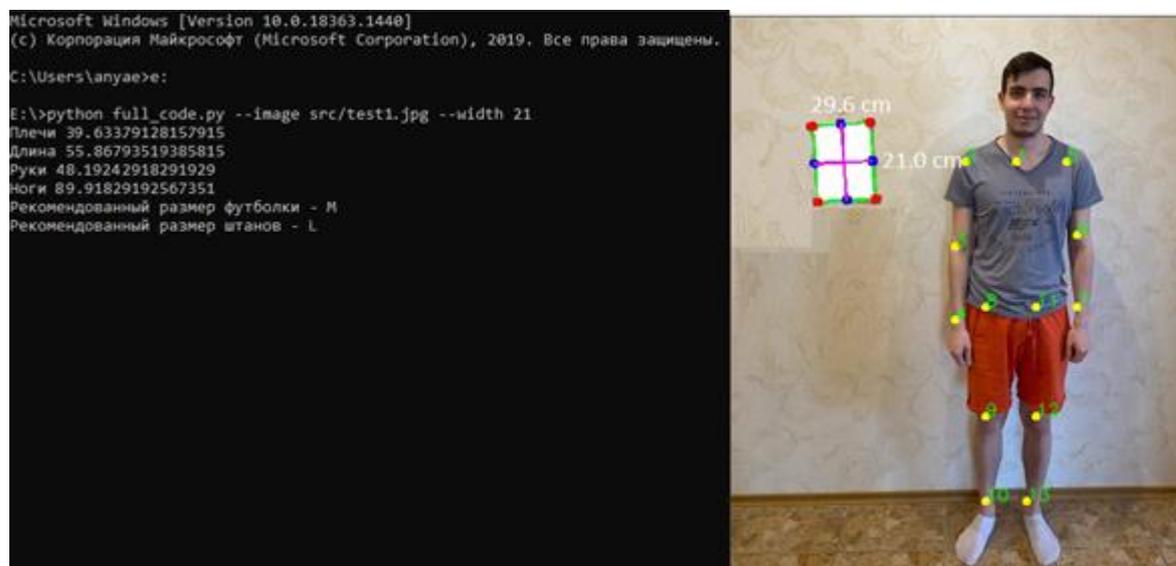


Рисунок 15 - Результаты работы алгоритма на модели 2

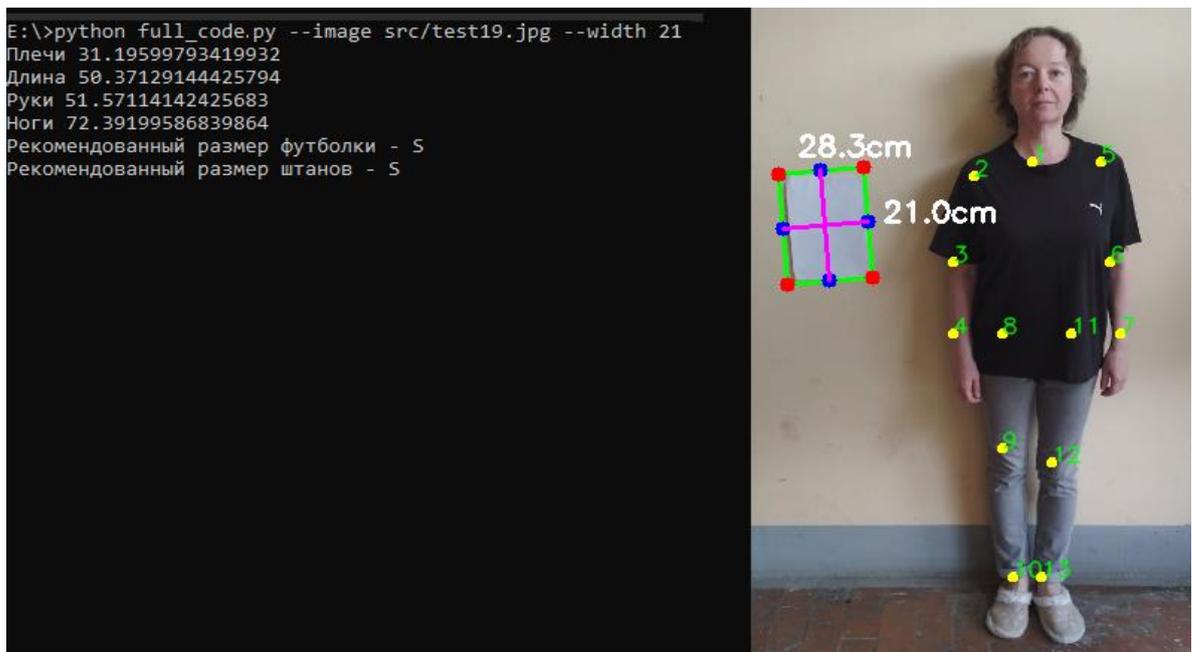


Рисунок 16 - Результаты работы алгоритма на модели 3



Рисунок 17 - Результаты работы алгоритма на модели 4

В Таблице 1 показано сравнение результатов работы алгоритма с реальными значениями:

Таблица 1 – Сравнение результатов работы алгоритма

Номер модели	Название параметра	Результат работы алгоритма (см)	Измеренные вручную значения (см)
Модель 1	Ширина плеч	31,84	32
	Длина туловища	50,87	51
	Длина рук	46,59	48
	Длина ног	79,99	81
Модель 2	Ширина плеч	39,63	40
	Длина туловища	55,87	55
	Длина рук	48,19	50
	Длина ног	89,91	90
Модель 3	Ширина плеч	31,2	31
	Длина туловища	50,37	49
	Длина рук	51,57	51
	Длина ног	72,39	74
Модель 4	Ширина плеч	32,96	34
	Длина туловища	52,29	54
	Длина рук	54,68	56
	Длина ног	83,31	85

Как мы видим, погрешность полученных в результате работы алгоритма вычислений варьируется в пределах 0,1-1,5 сантиметров.

Далее проанализируем разработанный алгоритм, проведя практическое сравнение с другим решением, а также выявим его основные достоинства и недостатки.

Для практического сравнения с предложенным в данной работе решением было выбрано приложение для определения размеров тела человека «Sizer».

Чтобы получить необходимые для сравнения данные из приложения, нужно выполнить следующий порядок действий: зайти на сайт sizer.me/, скачать и установить приложение, пройти регистрацию, прослушать объяснения и рекомендации перед началом работы, провести сканирование тела при помощи камеры мобильного телефона, и только потом получить нужные параметры.

На Рисунках 18 и 19 приведены результаты работы приложения «Sizer» на модели 1 и модели 4 из примеров выше.

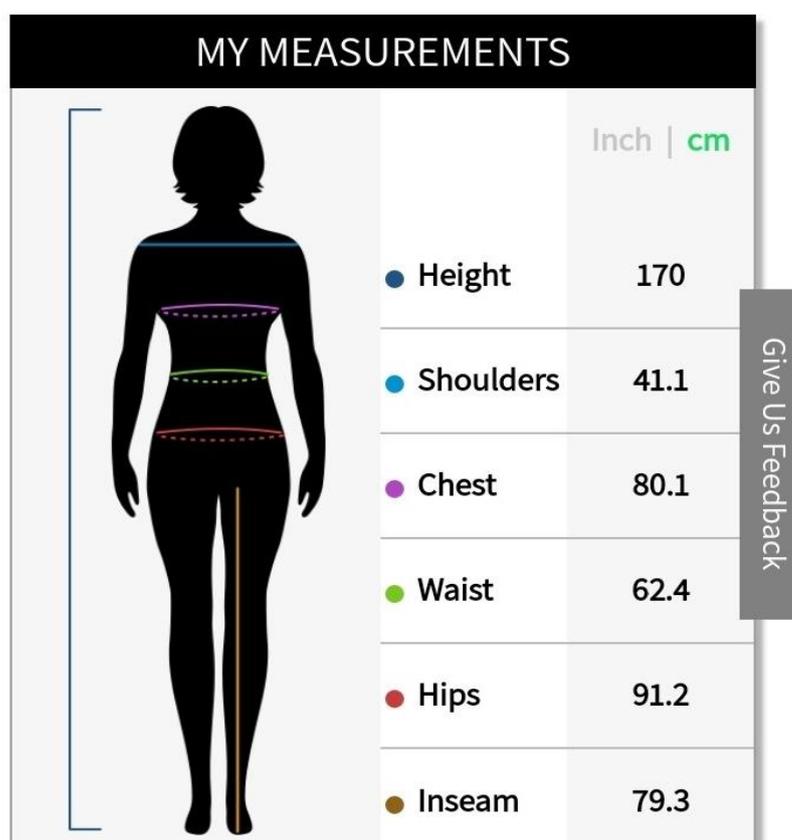


Рисунок 18 - Результат работы приложения «Sizer» на модели 1

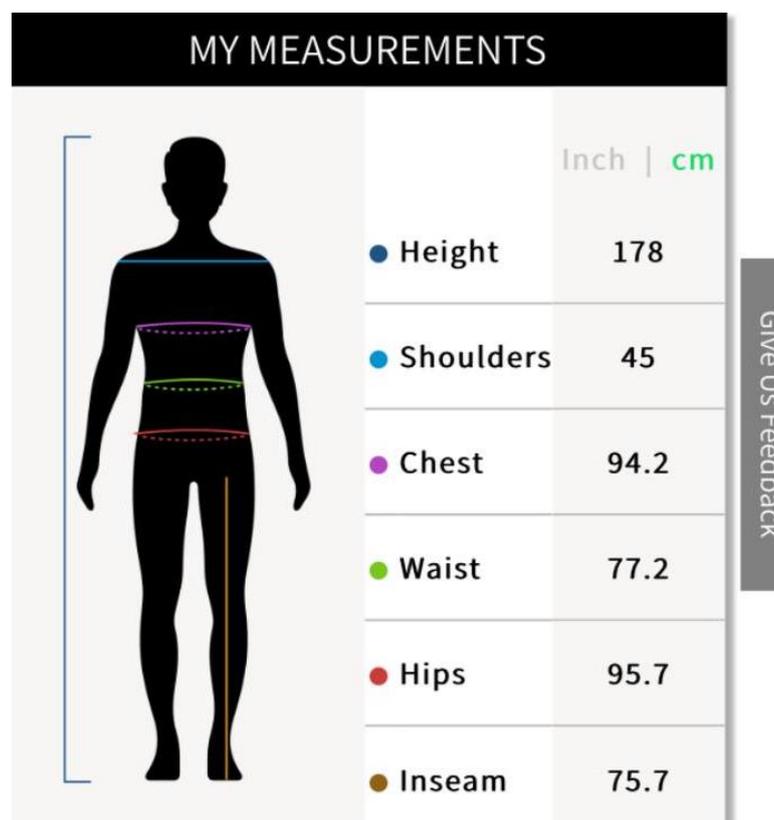


Рисунок 19 - Результат работы приложения «Sizer» на модели 4

Измерим с модели 1 и модели 4 вручную значения, как показано на Рисунках 18 и 19, и представим полученные результаты в Таблицах 2 и 3.

Таблица 2 - Сравнение результатов работы приложения на модели 1

Название параметра	Результат работы приложения (см)	Измеренные вручную значения (см)
Ширина плеч	41,1	40
Объем груди	80,1	84
Объем талии	62,4	60
Объем бедер	91,2	94
Длина ног	79,3	74

Таблица 3 - Сравнение результатов работы приложения на модели 4

Название параметра	Результат работы приложения (см)	Измеренные вручную значения (см)
Ширина плеч	45	44
Объем груди	94,2	95
Объем талии	77,2	78
Объем бедер	95,7	95
Длина ног	75,7	76

Исходя из данных в Таблицах 2 и 3, погрешность значений, полученных в ходе использования приложения, составляет около 0,3-5 сантиметров.

При проведении тестирования приложения «Sizer» были выявлены следующие основные недостатки:

- Приложение требует больших подготовительных процессов для выполнения вычислений. Учитывая человеческий фактор, риск совершения ошибок повышается. Созданное в ходе данной работы решение является более простым и понятным в использовании, для определения размеров тела человека достаточно только изображения.
- Получение необходимых значений в приложении «Sizer» является достаточно долгим по времени процессом. На скачивание, установку, регистрацию, прослушивание инструкций и сканирование тела пользователь может потратить от 10 минут. Далее – приложению нужно около 2-х минут на то, чтобы произвести сами вычисления и выдать результат. Предложенное же в данной работе решение выдает результат намного быстрее, примерно за 3-10 секунд (зависит от технических характеристик компьютера, на котором запускается программа).

Выводы

Таким образом, исходя из проведенного тестирования и анализа полученного решения, можно сделать вывод о том, что разработанный в ходе данной работы алгоритм обладает достаточной эффективностью и скоростью для удовлетворения требования работы в реальном времени. Погрешность предложенного алгоритма составляет около 0,1-1,5 сантиметров.

Для дальнейшего улучшения программы необходимо добавить рассмотрение тела человека с других ракурсов (сбоку/сзади), провести исследование того, как расстояние от камеры до человека/листа бумаги влияет на результат и уменьшить зависимость точности алгоритма от качества освещения/фона на изображении.

Заключение

В рамках данной работы были выполнены следующие задачи:

- Проведен анализ существующих методов определения размеров тела человека по изображениям;
- Предложено собственное решение исходной задачи;
- Разработан программный код, реализующий: получение и обработку исходного изображения; сегментацию листа бумаги; выделение ключевых точек на теле человека; вычисление размеров (в сантиметрах) тела человека и вывод рекомендованного размера одежды;
- Код размещен в репозитории на гитхаб по ссылке github.com/alunet/diploma_bodysize ;

Решение было протестировано и проанализировано, и также были предложены возможные пути развития для дальнейшего улучшения.

Список литературы

1. C. Liu, X. Fan, Z. Guo, Z. Mo, E. I.-C. Chang, and Y. Xu, “Wound area measurement with 3d transformation and smartphone images” *BMC Bioinformatics*, vol. 20, no. 724, 2019.
2. T. Xiaohui, P. Xiaoyu, L. Liwen, and X. Qing, “Automatic human body feature extraction and personal size measurement” *Journal of Visual Languages & Computing*, vol. 47, pp. 9 – 18, 2018.
3. J. Boisvert, C. Shu, S. Wuhrer, and P. Xi, “Three-dimensional human shape inference from silhouettes: Reconstruction and validation” *Machine Vision and Applications*, vol. 24, no. 1, pp. 145–157, January 2013.
4. Balan, A.O., Black, M.J.: The naked truth: Estimating body shape under clothing. In: *European Conf. on Computer Vision*. pp. 15–29. Springer (2008)
5. Zhang, C., Pujades, S., Black, M., Pons-Moll, G.: Detailed, accurate, human shape estimation from clothed 3D scan sequences. In: *IEEE CVPR* (2017)
6. Yang, J., Franco, J.S., Hetroy-Wheeler, F., Wuhrer, S.: Analyzing clothing layer deformation statistics of 3d human motions. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) *Computer Vision – ECCV 2018*. pp. 245–261. Springer International Publishing, Cham (2018)
7. Chen, X., Pang, A., Zhu, Y., Li, Y., Luo, X., Zhang, G., Wang, P., Zhang, Y., Li, S., Yu, J.: Towards 3d human shape recovery under clothing. *CoRR abs/1904.02601* (2019)
8. Kanazawa, A., Black, M.J., Jacobs, D.W., Malik, J.: End-to-end recovery of human shape and pose. In: *Computer Vision and Pattern Recognition (CVPR)* (2018)
9. Omran, M., Lassner, C., Pons-Moll, G., Gehler, P., Schiele, B.: Neural body fitting: Unifying deep learning and model based human pose and shape estimation. In: *International Conf. on 3D Vision* (2018)

10. Xu, Y., Zhu, S.C., Tung, T.: Denserac: Joint 3d pose and shape estimation by dense render and compare. In: International Conference on Computer Vision (2019)
11. Kolotouros, N., Pavlakos, G., Daniilidis, K.: Convolutional mesh regression for single-image human shape reconstruction. In: CVPR (2019)
12. Kolotouros, N., Pavlakos, G., Black, M.J., Daniilidis, K.: Learning to reconstruct 3D human pose and shape via model-fitting in the loop. In: International Conference on Computer Vision (Oct 2019)
13. Xiang, D., Joo, H., Sheikh, Y.: Monocular total capture: Posing face, body, and hands in the wild. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 10965–10974 (2019)
14. Приложение "Sizer" [Электронный ресурс]: URL: <https://sizer.me/> (дата обращения: 15.05.2021).
15. Документация DensePose [Электронный ресурс]: URL: <http://densepose.org/> (дата обращения: 15.05.2021).
16. Репозиторий OpenPose [Электронный ресурс]: URL: <https://github.com/CMU-Perceptual-Computing-Lab/openpose> (дата обращения: 16.02.2021).
17. Техническая документация библиотеки OpenCV [Электронный ресурс]: URL: <https://opencv.org/> (дата обращения: 16.02.2021).
18. Документация DNN [Электронный ресурс]: URL: https://docs.opencv.org/master/d6/d0f/group__dnn.html (дата обращения: 16.02.2021).
19. Датасет COCO [Электронный ресурс]: URL: <https://cocodataset.org/#keypoints-2018> (дата обращения: 1.02.2021).