МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ФАКУЛЬТЕТ ПРИКЛАДНОЙ МАТЕМАТИКИ И ИНФОРМАТИКИ
Кафедра дискретной математики и алгоритмики

КОВАЛЁВ Павел Алексеевич

РАЗРАБОТКА НЕЙРОСЕТЕВЫХ МОДЕЛЕЙ ДЛЯ ОЦЕНКИ ПОЗЫ ЧЕЛОВЕКА ПО МОНОКУЛЯРНОЙ КАМЕРЕ

Магистерская диссертация

специальность 1-31 81 09 «Алгоритмы и системы обработки больших объемов информации»

Научный руководитель
Белоцерковский Алексей Маратович
кандидат технических наук

Допущена к защите
«___» _____ 2020 г.
Зав. кафедрой дискретной математики и алгоритмики
В.М. Котов
доктор физико-математических наук, профессор

Минск, 2020

ОГЛАВЛЕНИЕ

ПЕРЕЧЕНЬ УСЛОВНЫХ ОБОЗНАЧЕНИЙ И СОКРАЩЕНИЙ

MSE – среднеквадратичная ошибка (mean square error)
MAE – средняя абсолютная ошибка (mean absolute error)
CCE – категориальная кросс-энтропия (categorical crossentropy)
BCE – бинарная кросс-энтропия (binary crossentropy)
LR – градиентный шаг (learning rate)
SPPE – оценка позы одного человека (single person pose estimation)
MPPE – оценка позы нескольких человек (multi person pose estimation)

ВВЕДЕНИЕ

Задача оценки позы человека по фото или видео с монокулярной камеры в последнее время привлекает все большее внимание научного сообщества. Одной из причин такого тренда является рост количества возможных применений: взаимодействие с компьютером, роботами; использование в кино и видеоиграх; анализ движений и производительности спортсменов; видеонаблюдение, контроль безопасности и др. Другой причиной является успешное применение глубокого обучения в задачах компьютерного зрения.

Целью исследования является развитие применения глубоких сверточных нейронных сетей для задачи предсказания взаимного плотного соответствия пикселей на изображениях с людьми и UV координат на фрагментах развертки эталонной трехмерной модели.

В данной работе описываются задачи связанные с оценкой позы человека, производится обзор релевантной литературы и уже существующих решений, на основе идей из которых предлагается новая нейросетевая архитектура для совместного решения задач бинарной сегментации людей на изображениях, семантической сегментации частей тел, локализации точек скелета, регрессии UV координат, разделяющей объекты сегментации. Эта архитектура представляет собой каскадную полносверточную нейронную сеть, в которой предварительно полученные предсказания для более простых задач используются для упрощения обучения следующих частей каскада.

Для обучения сети использовался язык программирования Python и фреймворк Keras с бэкендом Tensorflow. Исходный код проекта опубликован в репозитории на github (*https://github.com/PvtKowalski/human-pose*).

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Магистерская диссертация, 41 с., 25 источников, 37 иллюстраций, 1 таблица, 1 приложение.

ОЦЕНКА ПОЗЫ ЧЕЛОВЕКА, МАШИННОЕ ОБУЧЕНИЕ, СВЕРТОЧНЫЕ НЕЙРОННЫЕ СЕТИ, КОМПЬЮТЕРНОЕ ЗРЕНИЕ.

Цель: развитие применения глубоких сверточных нейронных сетей для задачи предсказания плотного соответствия на изображениях с людьми.

Задачи: изучить литературу по теме, обучить нейронную сеть для предсказания плотного соответствия совместно с другими задачами, в т. ч. задачей разделяющей сегментацией объектов.

Объект исследования: глубокие нейросетевые алгоритмы, в контексте задач компьютерного зрения.

Предмет исследования: применение нейронных сетей для анализа изображений с людьми.

В ходе работы получены следующие результаты:

Проведен обзор релевантной литературы, на основе которой предложена многозадачная архитектура для анализа изображений с людьми.

Создан конвейер для обучения нейронной сети на популярном фреймворке для глубокого обучения.

Проанализированы полученные результаты.

Новизну представляет совмещение в одной полносверточной нейросетевой архитектуре решения задач предсказания плотного соответствия и разделяющей сегментации объектов.

Работа состоит из трех глав. В первой приводится обзор литературы по теме, во второй описан предлагаемый подход для обучения нейронной сети, в третьей главе представлены результаты проведенных экспериментов.

GENERAL CHARACTERISTICS OF THE WORK

Master's thesis, 41 pages, 25 sources, 37 figures, 1 table, 1 appendix.

HUMAN POSE ESTIMATION, MACHINE LEARNING, CONVOLUTIONAL NEURAL NETWORKS, COMPUTER VISION.

Goal: development of application of deep convolutional neural networks for the dense human pose estimation task.

Tasks: to study relevant literature, to train a neural network for dense human pose estimation jointly with other tasks, including instance segmentation.

Object of research: deep neural networks applied to computer vision tasks.

Subject of research: application of neural networks for visual human understanding.

The following results were obtained:

An overview of relevant literature was carried out, based on which a multitask architecture for visual human analysis was proposed.

A pipeline for neural network training was implemented using popular deep learning framework.

Obtained results were analyzed.

Novel aspect of this work is that single fully-convolutional architecture is used for joint solving of dense human pose estimation and instance segmentation.

The thesis consists of three chapters. First one provides the overview of relevant literature. The second one describes the proposed approach for neural network training. The third chapter presents the results of conducted experiments.

АГУЛЬНАЯ ХАРАКТАРЫСТЫКА РАБОТЫ

Магістарская дысертацыя, 41 с., 25 крыніц, 37 малюнкаў, 1 табліца, 1 дадатак.

АЦЭНКА ПОЗЫ ЧАЛАВЕКА, МАШЫННАЕ НАВУЧАННЕ, СВЕРТКАВЫЯ НЕЙРОННЫЯ СЕТКІ, КАМП’ЮТАРНЫ ЗРОК.

Мэта: развіцце выкарыстоўвання глыбокіх сверткавых нейронных сетак для задачы прадказання шчыльнай адпаведнасці на малюнках з людзьмі.

Задачы: вывучыць літаратуру па тэме, навучыць нейронную сетку для прадказання шчыльнай адпаведнасці сумесна з іншымі задачамі, у т. л. сегментцыйным падзяленнем аб'ектаў.

Аб'ект даследавання: глыбокія нейрасеткавыя алгарытмы, у кантэксце задач камп'ютарнага зроку.

Прадмет даследавання: выкарыстанне нейроных сетак для аналізу малюнкаў з людзьмі.

У рабоце атрыманы наступныя вынікі:

Праведзены агляд рэлевантнай літаратуры, на аснове якой прапанавана шматзадачная архітэктура для аналізу малюнкаў з людзьмі.

Створаны канвеер для навучання нейроннай сеткі на папулярным фрэймворке для глыбокага навучання.

Прааналізаваны атрыманыя вынікі.

Навізну ўяўляе сумяшчэнне ў адной полнасверткавай нейрасеткавай архітэктуры рашэння задач прадказання шчыльнай адпаведнасці і сегментацыйнага падзялення аб'ектаў.

Праца складаецца з трох глаў. У першай прыводзіцца агляд літаратуры па тэме, у другой апісаны прапанаваны падыход для навучання нейроннай сеткі, у трэцяй главе прадстаўлены вынікі праведзеных эксперыментаў.

ГЛАВА 1.
ОБЗОР ЛИТЕРАТУРЫ И ОСНОВНЫЕ ПОНЯТИЯ

1.1. Задача оценки двумерной позы человека

Самая простая постановка задачи определения позы человека – предсказание двумерных координат суставов для одного человека на изображении. Для случая, когда людей на изображении много, существует два основных подхода к решению: «top-down», «bottom-up». Решения в стиле «bottom-up» подразумевают локализацию суставов всех людей на изображении, после чего производится «сборка» скелетов людей из обнаруженных суставов. «Top-down» подходы сначала используют детектор, с помощью которого из изображения выделяются прямоугольные регионы с единственным человеком, после чего для каждого из этих регионов производится локализация суставов.

1.1.1. DeepPose: Human Pose Estimation via Deep Neural Networks

Одной из первых работ по предсказанию позы человека на изображении с помощью глубоких нейронных сетей является DeepPose [1]. Авторы рассматривали эту задачу как задачу регрессии.

В DeepPose решалась задача определения позы одного человека на изображении (single person pose estimation). По размеченным суставам из изображений вырезались регионы с одним человеком, координаты суставов в пикселах пересчитывались относительно центра вырезанного региона и нормализовались.

Для предсказания координат суставов использовалась глубокая сверточная нейронная сеть, имеющая в общей сложности 5 сверточных и 3 полносвязных слоя. Суммарно модель имела примерно 40 миллионов обучаемых параметров. На вход подавалось изображение размера 220x220. Функция ошибки — сумма квадратов отклонений координат.

Сеть обучалась стохастическим градиентным спуском с размером батча 128 и LR 0.0005. Поскольку авторами использовался относительно небольшой датасет, а размер модели был достаточно большим, данные были аугментированы с помощью случайно вырезанных из исходного изображения прямоугольников, отзеркаливаний; также на полносвязных слоях использовалась регуляризация dropout с параметром 0.6.

Кроме того, для улучшения предсказаний использовался каскад нейронных сетей, где каждая последующая сеть уточняет предсказания предыдущей.

Рисунок 1.1

На рисунке 1.1 изображена общая схема подхода, описанного в DeepPose.

Из исходного изображения вырезались регионы вокруг предсказаний предыдущей сети для каждого сустава. Следующие сети (с такой же архитектурой, как и основная) учились на этих изображениях предсказывать корректирующее предыдущие координаты смещение. Для уточняющих стадий каскада использовалась аугментация, сэмплирующая немного смещенные регионы вокруг изначально предсказанного сустава.

Рисунок 1.2

Данные (примеры на рисунке 1.2), которыми пользовались авторы статьи (Frames Labeled in Cinema [2], Leeds Sports Dataset и его расширение [3]), представляют собой изображения, на которых размечены суставы людей. FLIC содержит 4000 изображений в обучающей и 1000 изображений в тестовой выборке. Датасет содержит кадры из голливудских фильмов с разнообразными позами и одеждой. Для каждого человека размечено 10 суставов верхней части туловища. LSD — 11000 изображений для обучения, 1000 для тестирования. Датасет содержит изображения людей, занимающихся спортом, большинство людей ростом в 150 пикселов, для каждого размечено 14 суставов всего тела.

В статье вводились следующие метрики:

Процент правильных частей (Percentage of Correct Parts, PCP). Часть (конечность) считалась найденной, если оба ее предсказанных сустава смещены не более чем на 50% истинной длины части.

Процент найденных суставов (Percent of Detected Joints, PDJ). Сустав считается найденным, если расстояние между предсказанным суставом и истинным менее некоторой доли диаметра торса (доля варьируется для построения кривой точности).

Таким образом, в данной статье мы видим первое применение глубоких нейронных сетей к задаче оценки позы людей. Авторами были исследованы идеи использования каскада регрессоров, уточняющих предсказания. Данный подход позволил им превзойти по качеству существующие на тот момент результаты.

1.1.2. Stacked Hourglass Networks for Human Pose Estimation

Идеи каскадной архитектуры с несколькими стадиями улучшения предсказаний развивают авторы статьи Stacked Hourglass Networks for Human Pose Estimation [4]. В этой статье также решается задача SPPE по одному изображению с монокулярной камеры.

Рассмотрим ключевые особенности архитектуры предложенной нейронной сети (общая схема на рисунке 1.3).

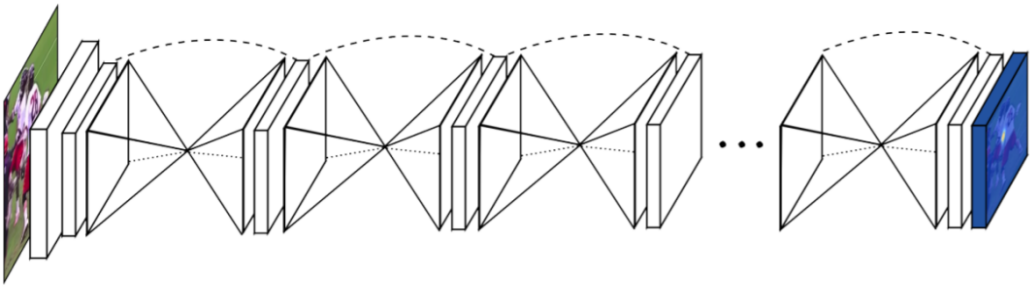


Рисунок 1.3

Используются «hourglass modules» (рисунок 1.4), которые представляют собой полносверточную сеть, сочетающую в себе идею U-Net [5] и блоки из ResNet [6]. В данных модулях сверточные слои и слои субдискретизации применяются для трансформации признаков на уровень с низким пространственным разрешением (4x4). После каждой операции субдискретизации сеть создает ответвление с дополнительными свертками, но без изменения оригинального разрешения. С помощью слоев, повышающих дискретизацию, признаки низкого разрешения возвращаются к исходному размеру, объединяясь в процессе с признаками всех масштабов.

Для получения предсказаний к признакам последних слоев применяется дважды свертка 1x1. Выходом сети является набор тепловых карт, по одной карте на сустав скелета.

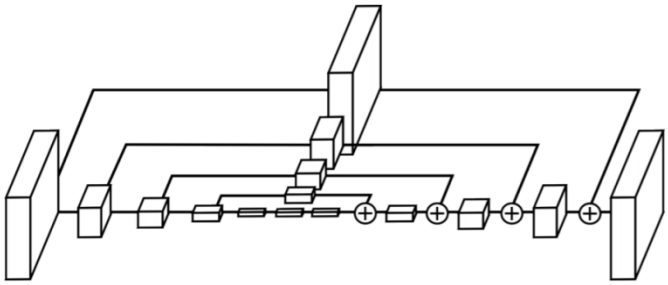


Рисунок 1.4

В архитектуре повсеместно используются блоки (residual modules, рисунок 1.5) из ResNet. Нигде не используются ядра сверток размером более 3x3. Благодаря этому уменьшается потребление памяти графического процессора. Также с этой целью начальное разрешение «hourglass» модулей было снижено с 256x256 до 64x64 с помощью свертки 7x7 с шагом 2 и residual блока с последующей субдискретизацией. Все блоки «hourglass» модуля возвращают тензоры с 256 каналами.

Применяется техника intermediate supervision (рисунок 1.5). Архитектура состоит из стадий, после каждой из которых получают промежуточные предсказания позы. Эти предсказания объединяются с другими признаками и используются для следующих стадий каскада с целью исправления ошибок. В финальной версии сети используется каскад из восьми частей.

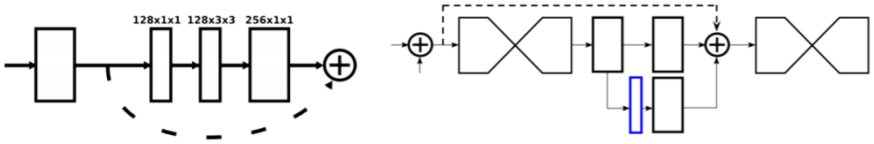


Рисунок 1.5

Как было отмечено выше, В данной архитектуре предсказываются не координаты напрямую как в DeepPose, а тепловые карты для каждого сустава. В разметке в точку с суставом помещается гауссиана (со среднеквадратичным отклонением в 1 пиксель). Сеть обучалась предсказывать только человека в середине изображения (при наличии нескольких человек в кадре). Данные аугментировались поворотами на 30 градусов, и вариацией масштаба 0.75–1.25.

При обучении используется оптимизатор RMSprop с LR 0.00025. Обучение сети занимало 3 дня на графическом процессоре NVIDIA TITAN X. После того как валидационная ошибка выходила на плато, LR уменьшался в 5 раз. Авторы использовали нормализацию батчей [7]. Получение предсказаний

занимало 75 мс. Использовалась функция ошибки MSE. Для улучшения предсказаний (при конвертации тепловых карт в значения координат) точка максимального пикселя смещалась на четверть пикселя в сторону наибольшего соседа (по вертикали и горизонтали).

Авторами статьи использовались датасеты FLIC и MPII [8]. MPII Human Pose Dataset содержит 28000 изображений людей в обучающей и 11000 в тестовой выборке. Для валидации использовалось 3000 изображений.

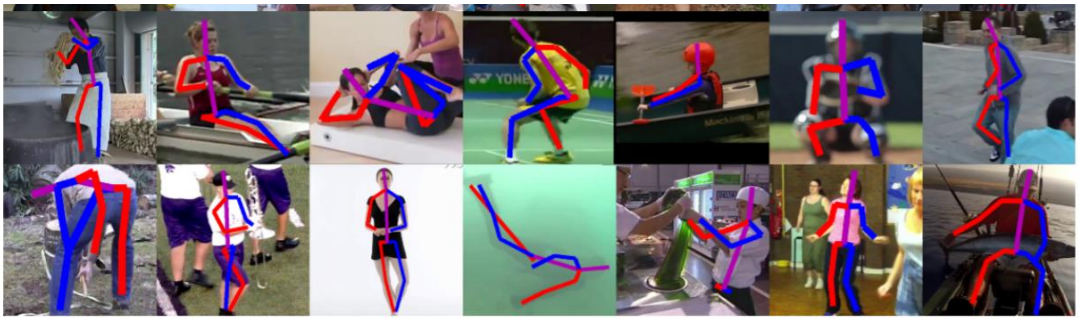


Рисунок 1.6

Авторами была продемонстрирована эффективность каскада «hourglass» модулей для предсказания позы человека (примеры предсказаний на рисунке 1.6). Были исследовано влияние параметров каскада (количество модулей, параметров intermediate supervision) на качество предсказаний.

1.1.3. Cascaded Pyramid Network for Multi-Person Pose Estimation

В статье Cascaded Pyramid Network for Multi-Person Pose Estimation [9] авторы решали задачу определения позы для нескольких людей на изображении. Они использовали top-down подход. Для детекции людей использовалась FPN [10], с операцией ROI-Align из Mask R-CNN [11]. Авторы обучали детектор на всех категориях из датасета MS COCO [12]. Общая схема подхода изображена на рисунке 1.7.

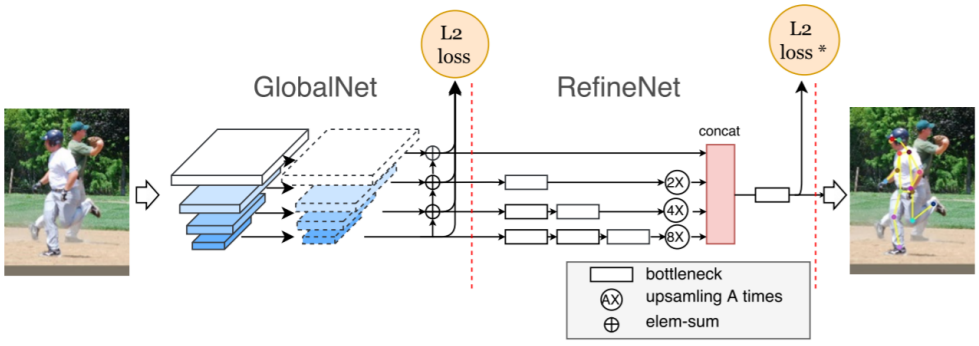


Рисунок 1.7

После стадии с детекцией людей для каждого человека детектируются суставы. GlobalNet основан на архитектуре ResNet. Тепловые карты суставов генерируются из каждого уровня признаков ResNet с помощью свертки 3x3. При этом более глубокие признаки после upsampling и свертки 1x1 складываются с признаками более высокого уровня.

Первую сеть дополняет RefineNet, который предназначен для улучшения качества определения сложных суставов. Признаки с каждого уровня GlobalNet конкатенируются, и на их основе делается еще одно предсказание суставов. Однако градиенты для обновления весов считаются исключительно по суставам, при определении которых сеть ошибается больше всего.

Модели обучались только на MS COCO (57K изображений, 150K различных людей), валидация проводилась на 5000 изображений из MS COCO minival датасета.

При обучении использовались аугментации: отражение по горизонтали, повороты (±45°) и случайный масштаб (0.7–1.35). Изначальные веса ResNet были предобучены на классификации датасета Imagenet [13].

1.1.4. Simple Baselines for Human Pose Estimation and Tracking

В статье Simple Baselines for Human Pose Estimation and Tracking [14], отмечается успех архитектур Stacked Hourglass и CPN, но авторы задаются вопросом, насколько хорошей может быть простая архитектура для определения позы.

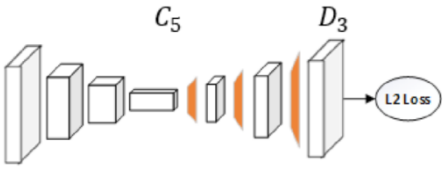


Рисунок 1.8

Авторами предлагается очень простая, по сравнению с описанными ранее, архитектура, состоящая из «головы» архитектуры ResNet и трех upsample блоков (рисунок 1.8).

Несмотря на простоту архитектуры, авторам удалось немного превзойти результаты Stacked Hourglass и CPN на MS COCO val2017.

Для детекции людей авторами статьи использовался Faster R-CNN [15]. При обучении вырезанные регионы с людьми расширялись до соотношения сторон 4:3. Из этого изображения вырезались части и приводились к фиксированному разрешению (например 256:192). Аугментации: масштаб

(±30%), вращения(±40 градусов) и горизонтальное отражение. ResNet был инициализирован весами с Imagenet [13].

1.2. Задача сегментации объектов

Самый простой тип задачи сегментации объектов — семантическая сегментация (semantic segmentation). Суть этой задачи в том, чтобы классифицировать каждый пиксель исходного изображения. Для решения этой задачи, как правило, используют полносверточные нейронные сети.

При решении задачи семантической сегментации нет необходимости различать пиксели, принадлежащие разным объектам одного класса. Задача, в которой разделение объектов принципиально, является намного более сложной. Для сегментации, разделяющей объекты, (instance segmentation) в основном используют два основных подхода: с использованием детектора (выполняющие поиск ограничивающей рамки для каждого объекта) и без (bounding-box free / anchor free методы, в которых, как правило, каждому пикселю строится некоторое векторное представление, позволяющее, например, кластеризовать пиксели для разделения объектов).

1.2.1. U-Net: Convolutional Networks for Biomedical Image Segmentation

Для решения задач семантической сегментации в настоящее время очень часто применяют нейросетевые архитектуры похожие на U-Net [5]. Авторы этой архитектуры развили подход, описанный в статье FCN [16], применяя его к задаче сегментации биомедицинских данных.

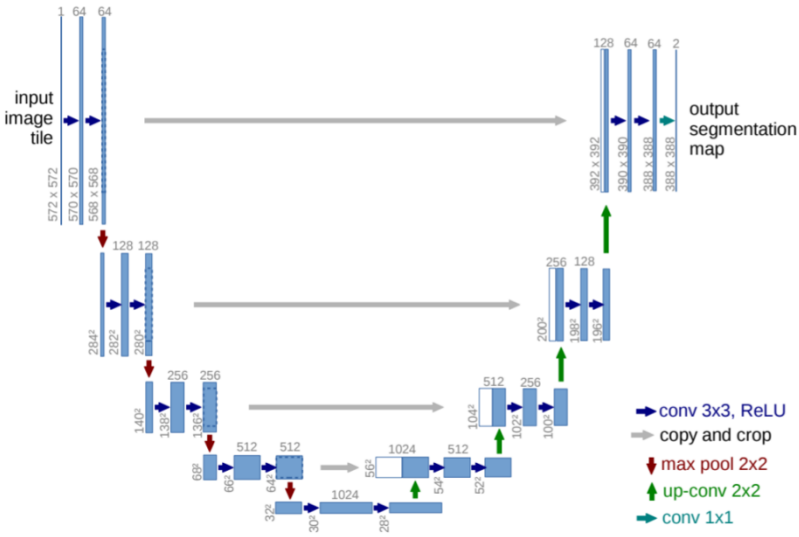


Рисунок 1.9

U-Net (оригинальная схема на рисунке 1.9) состоит из части кодирующей изображение, части декодирующей и переносов карт активаций между ними. Кодирующая сеть состоит из некоторого количества блоков (похожих на блоки архитектуры VGG [17]). Каждый такой блок состоит из двух сверток с фильтром 3x3 (с применением функции активации ReLU [18] после каждой свертки) и слоя субдискретизации с размером окна 2x2 и с шагом 2. На каждом следующем шаге вниз количество каналов удваивается.

Декодирующая сеть состоит из такого же количества блоков. Каждый из этих блоков состоит из слоя повышающей дискретизации (upsampling), свертки с размером фильтра 2x2, вдвое сокращающей количество каналов, далее идет конкатенация с соответствующей по размеру картой признаков из кодирующей сети (skip connection) и двух сверток с фильтром 3x3 (с применением активации ReLU после каждой свертки). На последнем слое используется свертка 1x1 и активация softmax для получения семантической карты для каждого класса.

Используя данную архитектуру, авторы добились хороших результатов в поставленных задачах по сегментации биомедицинских данных; что послужило причиной роста популярности подхода и успешного применения U-Net для семантической сегментации на данных других доменов.

1.2.2. Mask R-CNN

Относительно простой, гибкий фреймворк для разделяющей сегментации объектов предлагается авторами статьи Mask R-CNN [11]. Их подход совмещает детекцию объектов (с помощью ограничивающих рамок) вместе с качественными предсказаниями сегментационных масок. Свой метод авторы базируют на подходе, описанном в статье Faster R-CNN [15], но добавляют параллельную к предсказаниям рамок ветвь для предсказания сегментационных масок объектов.

Faster R-CNN имеет два выхода для каждого объекта-кандидата: метка класса и корректирующие смещения для рамки. К этим выходам добавляется третий выход, предсказывающий маску. Однако этот дополнительный выход требует извлечения пространственных признаков объекта без повреждения деталей и утери информации.

Сеть Faster R-CNN состоит из двух стадий. Первая часть, детектор регионов интереса (Region Proposal Network), предсказывает ограничивающие рамки объектов-кандидатов. Вторая часть извлекает признаки с использованием операции RoIPool и производит классификацию и уточняющую регрессию параметров рамок. При этом обе стадии переиспользуют предварительно полученные с помощью базовой сети (backbone) признаки.

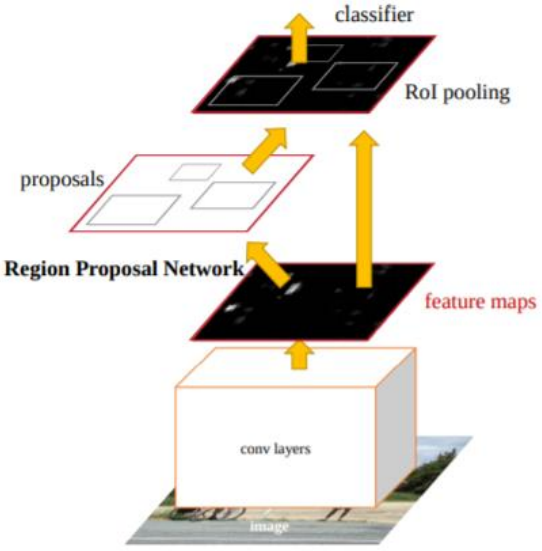


Рисунок 1.10

На рисунке 1.10 схематично изображен подход, описанный в Faster R-CNN.

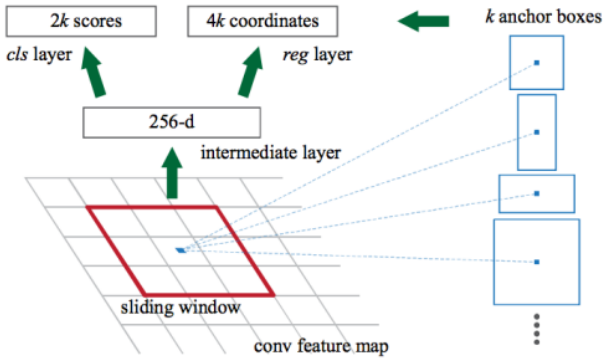


Рисунок 1.11

В Region Proposal Network (рисунок 1.11) по полученным базовой сетью признакам скользят окном 3x3. Каждое окно передают в небольшую сеть. Полученные значения передаются в два параллельных полносвязных слоя: box-regression layer и box-classification layer. Для подсчета выходов этих слоев используются якоря: k рамок для каждого положения скользящего окна, имеющих разные размеры и соотношения сторон. Слой регрессии рамок, для каждого такого якоря выдаёт 4 значения, корректирующие положение охватывающей рамки; слой классификации выдаёт два числа, которые соответствуют тому, что рамка содержит некоторый объект или нет.

Mask R-CNN (общая схема на рисунке 1.12) отличается от Faster R-CNN дополнительной веткой, которая предсказывает положение маски, покрывающей найденный объект, и решает уже задачу разделяющей сегментации объектов (instance segmentation).

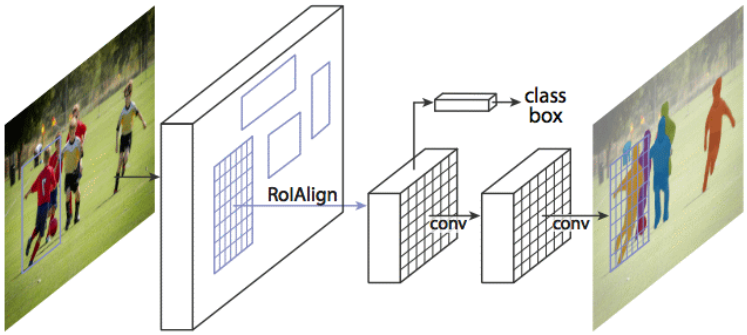


Рисунок 1.12

Маски объектов по признакам из RoIAlign предсказываются отдельно для каждого класса, далее выбирается маска класса с наибольшим предсказанием в классификаторе.

Поскольку для вычисления маски требуются более детализированные признаки, операция RoIPool, вычисляющая матрицу признаков для региона интереса, заменена на RoIAlign. В RoIPool выделение фрагмента карты признаков решалось округлением дробных значений параметров рамки.

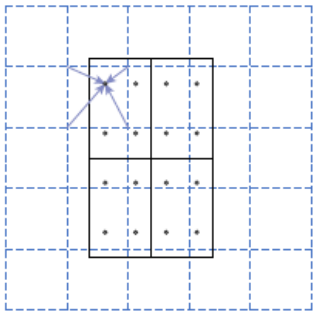


Рисунок 1.13

RoIAlign отличается тем, что использует билинейную интерполяцию признаков (рисунок 1.13).

1.2.3. Безрамочная паноптическая сегментация

Для того чтобы разделять предсказания для людей на изображении вместо подхода с использованием детектора вроде Mask R-CNN, возможен подход без детекции прямоугольниками — т. н. box-free instance segmentation. Решение задачи с помощью одной полносверточной сети значительно упростило бы обучение и программную реализацию и потенциально ускорило бы получение предсказаний.

В статье BBFNet [19] решалась задача паноптической сегментации (panoptic segmentation), одновременной сегментации отдельных объектов и неисчислимого «вещества» (например классы небо, лес, река и т. д.). Для

решения задачи авторы использовали совместное обучение (рисунок 1.14), решая 4 подзадачи: семантическую сегментацию, Hsough voting (регрессию векторов в центр объекта из каждого пикселя объекта, с помощью которых можно было бы кластеризовать пиксели, тем самым разделяя объекты), водораздел (классификация пикселей объекта по степени удаленности от его границы), triplet loss (получение эмбеддингов пикселей с помощью данной функции ошибки позволяет сблизить пиксели одного объекта и раздвинуть для разных).

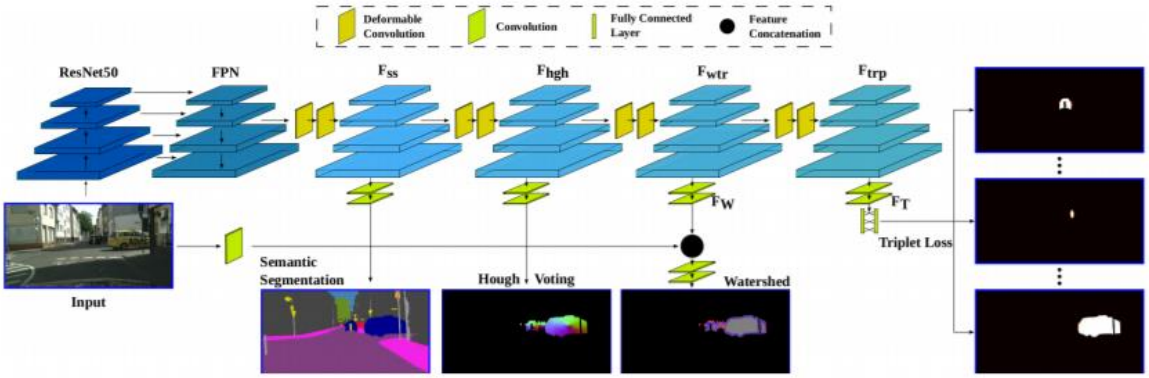


Рисунок 1.14

Ветвь семантической сегментации оптимизировалась с помощью кросс-энтропийной функции ошибки. Ветвь Hough voting оптимизировалась с помощью модифицированного квадратичного отклонения. Ветвь для предсказания поверхности водораздела (сегментация на 4 класса: фон, 5 пикселей от границы, 15 пикселей от границы объекта, остальной объект) также использовалась кросс-энтропийная функция ошибки. Четвертая ветвь вместо классического triplet loss использует кросс-энтропийную функцию ошибки для классификации пар пикселей на принадлежность одному объекту. Поскольку классифицировать каждую пару пикселей было бы очень вычислительно дорого, то для каждого объекта выбиралось фиксированное количество пикселей «якорей», для которых сэмплировались с помощью эвристик наиболее сложные пиксели принадлежащие тому же или различным объектам.

1.3. Задача предсказания плотного соответствия

В статье DensePose: Dense Human Pose Estimation In The Wild [20] представляется совершенно новая версия задачи определения позы человека. Авторы не просто предсказывают двумерные координаты суставов, а предлагают способ построения отображения пикселей человека на точки поверхности эталонной трехмерной модели.

1.3.1. DenseReg: Fully Convolutional Dense Shape Regression In-the-Wild

Одной из первых работ по предсказанию плотного соответствия пикселей и точек UV пространства непосредственно с помощью глубоких нейронных сетей является статья DenseReg [21]. Для получения данных для обучения использовались датасеты с естественными изображениями лиц с разметкой в виде двумерных ключевых точек, по которым можно было, разместив 3D шаблон, получить необходимую разметку.

Задача поставлена таким образом, что для каждого пикселя на изображении с лицом человека необходимо предсказать, принадлежит ли он лицу, а также два значения UV координат на поверхности шаблона. Авторы пытались подойти к проблеме напрямую, но из-за сложности задачи простые подходы с использованием сверточных нейронных сетей не давали хороших результатов. Поэтому авторы воспользовались техникой квантизации регрессии (рисунок 1.15), они разбили искомое UV пространство на сетку; решая задачу классификации, определяли дискретный регион, в котором находится пиксель, и далее регрессировали оставшеюся ошибку. К такому гибридному подходу авторы пришли, анализируя результаты в областях семантической сегментации и локализации ключевых точек, где нейронные сети лучше справляются именно с классификационной задачей.

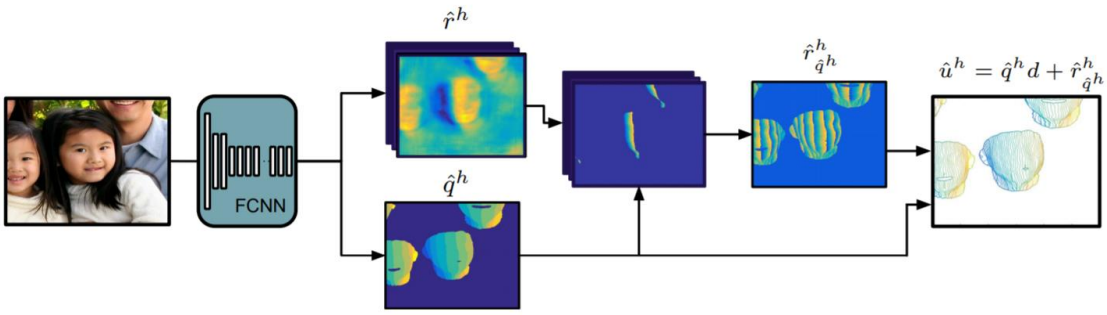


Рисунок 1.15

Таким образом, в данной статье была описана задача регрессии плотного соответствия с использованием разметки, полученной с помощью подбора 3D шаблона лица. Была предложена полносверточная архитектура, использующая идеи из задач семантической сегментации и плотной регрессии. Авторами показано, что их схема с квантизацией регрессии позволяет превзойти более простые подходы. Также предсказание плотного соответствия позволяет решать смежные задачи, такие как локализация ключевых точек и семантическая сегментация, путем переноса аннотаций с единого шаблона. Авторами была продемонстрирована обобщаемость метода на другие домены деформирующихся поверхностей, таких как тела людей и человеческие уши.

1.3.2. DensePose: Dense Human Pose Estimation In The Wild

В данной статье рассматривается задача предсказания соответствия пикселей и точек на поверхности эталонной трехмерной модели человека.

Простейшая архитектура нейронной сети для такой задачи состоит в том, чтобы использовать полносверточную сеть, которая одновременно выполняет классификацию и регрессию. Одна часть сети классифицирует пиксель, принадлежит он фону или какому-либо из 24 фрагментов на развертке. Это делается с помощью стандартного сегментационного подхода с кросс-энтропийной функцией ошибки. Вторая часть уточняет координаты каждого пикселя внутри своего фрагмента, выполняя регрессию двух UV координат используя функцию ошибки MAE.

Данный подход просто реализовать, однако это нагружает одну сеть разными задачами, что создает трудности при обучении.

В статье описывается несколько вариантов архитектур помимо простой полносверточной сети. Авторы статьи адаптируют к своей задаче архитектуру Mask-RCNN [11] с основой из ResNet50 [6] и применением пирамиды признаков (Feature Pyramid Networks [10]). После слоя RoIAlign добавляется полносверточная сеть для классификации и регрессии, состоящая для простоты из 8 чередующихся слоев 512 канальной свертки с размером фильтра 3x3 и функцией активации ReLU. Таким образом, для каждого обнаруженного детектором человека предсказывалась сегментация фрагментов тела и регрессия UV координат (рисунок 1.16).

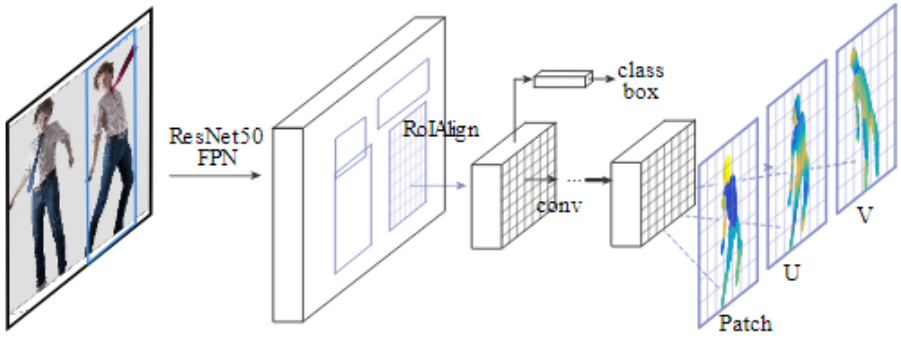


Рисунок 1.16

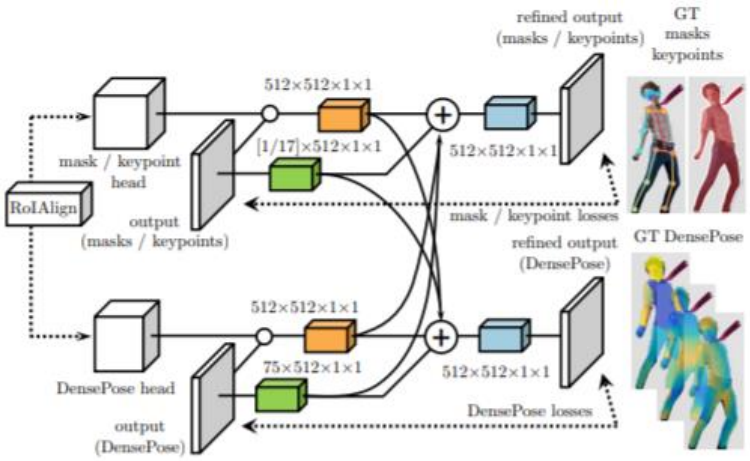


Рисунок 1.17

Для улучшения качества предсказаний авторы дополняют Mask-RCNN каскадной архитектурой (рисунок 1.17) с дополнительным обучающим сигналом. Сеть учится предсказывать суставы и общую сегментационную маску. Признаки, которые она выучивает для этих задач, объединяются с признаками для UV регрессии и промежуточными предсказаниями.

1.4. Выводы

В данной главе был представлен обзор релевантной литературы. Сформулированы задачи определения позы человека в постановках локализации точек суставов и предсказания плотного соответствия, а также семантической сегментации и сегментации отдельных объектов. Были рассмотрены различные подходы и техники, применяемые для решения поставленных задач.

ГЛАВА 2.
ОПИСАНИЕ ПРЕДЛАГАЕМОГО РЕШЕНИЯ

2.1. Используемые данные

В данном разделе описываются использованные в экспериментах данные, процедуры фильтрации датасета, предварительной обработки и аугментации.

2.1.1. Датасет MS COCO

Датасет MS COCO предназначен для исследования проблем в задаче анализа изображений, таких как: детекция объектов, анализ контекстной связи между объектами. Основным приоритетом в создании датасета являлся сбор изображений естественных сцен (in the wild), поскольку редкие ракурсы, загражденные и тесно расположенные объекты, встречающиеся в повседневных сценах, представляли трудность для систем распознавания. Авторы датасета экспериментально установили, что модели, обученные на их данных, показывали лучшие результаты, чем на датасетах, существовавших ранее.

Авторами уделялось особое внимание сбору изображений с большим количеством объектов, а также с объектами, распознавание которых требует анализа контекста (так как сам объект может быть малого размера, размыт, заслонен и т. д.). Пример изображения можно видеть на рисунке 2.1.

Объекты на изображениях в датасете размечены ограничивающими рамками, а также сегментационными масками (а на изображениях с людьми также размечались суставы скелета). Для того чтобы разметить большое количество изображений использовалась платформа Amazon Mechanical Turk.

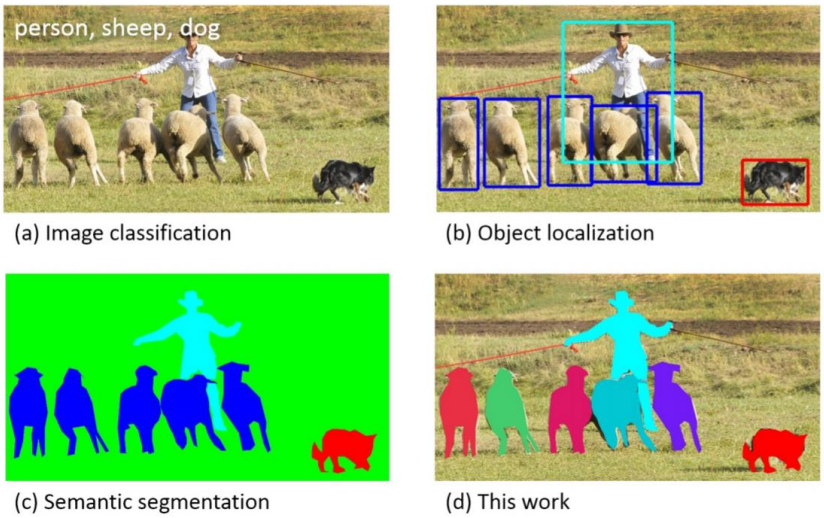


Рисунок 2.1

Датасет MS COCO содержит объекты 91 класса, для 82 из которых размечено более пяти тысяч объектов. В целом, датасет содержит около 2,500,000 размеченных объектов на 328,000 изображениях. Авторы отмечают преимущества своего датасета перед предшествующими ImageNet, PASCAL VOC и SUN, основными из которых являются: разнообразие, богатая разметка, большое количество объектов для каждого класса, а также большее количество объектов на каждом изображении.

Версия датасета, опубликованная в 2014 году, содержит 82,783 изображений в обучающей выборке, 40,504 в валидационной, и 40,775 в тестовой. В датасете около 270,000 людей с сегментационной разметкой и всего около 886,000 отсегментированных объектов (в обучающей и валидационной выборках).

2.1.2. Датасет COCO-DensePose

Поскольку ранее не существовало датасетов для задачи предсказания плотного соответствия для изображений людей, авторы статьи DensePose занялись созданием нового датасета на основе MS COCO. Ими было размечено 50,000 различных людей, при этом вручную было размечено более пяти миллионов соответствий (пикселей и точек на поверхности эталонной трехмерной модели). Схема процесса разметки на рисунке 2.2.

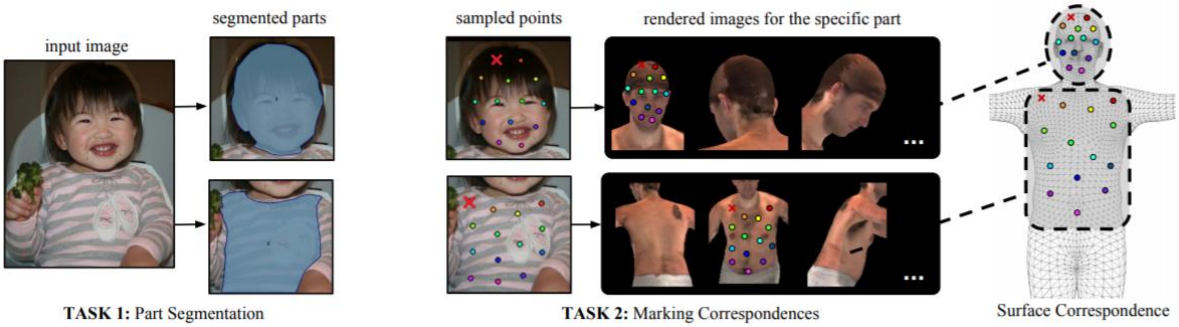


Рисунок 2.2

Для этого датасет Microsoft COCO был дополнен разметкой, включающей в себя: семантическую разметку 14 частей тела, набор точек (в среднем 100 точек на человека) с индексами соответствующего фрагмента развертки эталонной трехмерной модели человека SMPL [22] с локальными UV координатами для этого фрагмента (всего используется 24 фрагмента).

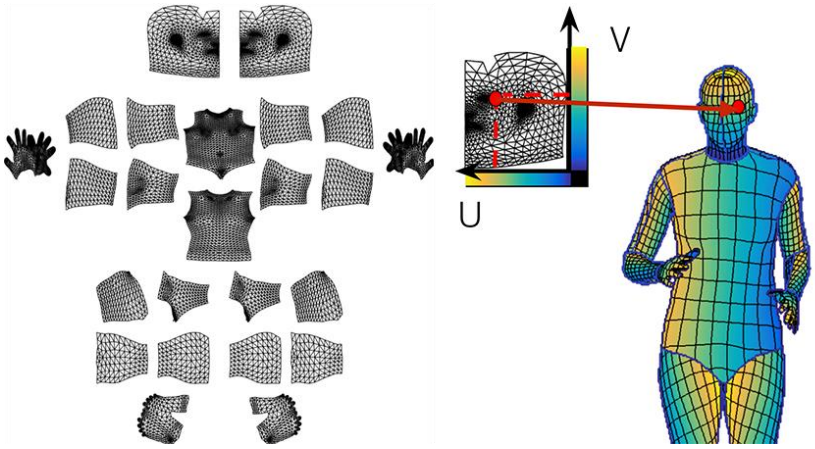


Рисунок 2.3

На рисунке 2.3 можно видеть все 24 фрагмента развертки и их структуру.

Получение такой разметки выполнялось в несколько стадий. Сначала, размечались сегментационные маски, а затем, сгенерированные на них точки предлагалось поставить в соответствие точками на эталонной модели.

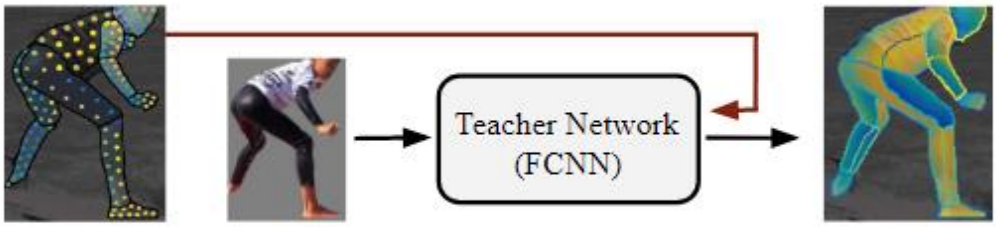


Рисунок 2.4

Чтобы получить разметку каждого пикселя, авторами была обучена нейросеть для интерполяции разметки (рисунок 2.4). Однако в опубликованной версии проекта, авторы вместо интерполирующей сети используют модифицированную функцию ошибки, позволяющую обучаться на точечной неплотной разметке.

2.1.3. Фильтрация, предобработка и аугментация данных

Обучение проводилось только на изображениях из DensePose COCO, на которых есть люди с разметкой DensePose, размер рамки которых минимум 90 пикселей в высоту или ширину, а также присутствует разметка минимум 5 суставов скелета.

Рамка изображения расширялась до соотношения сторон 1:1, само изображение дополнялось с краев до новой рамки (значениями яркости равными краевым). Затем изображение вместе с разметкой (сегментационные

маски, точки соответствия поверхностей и пр.) приводилось к размеру 256x256.

Аугментация данных (data augmentation) — это методика увеличения количества обучающих данных путем модификации имеющихся данных. Для достижения хороших результатов глубокие сети должны обучаться на больших разнообразных обучающих выборках. Поэтому, если исходный обучающий набор содержит небольшое количество изображений, необходимо выполнить аугментацию, чтобы улучшить способность модели к обобщению.

В своих экспериментах мы использовали следующие аугментации данных: вариации масштаба (80%–140%), смещения (±20%), повороты (±35 градусов), сдвиг (±4 градуса), а также аддитивные изменения яркости и гауссово размытие.

2.2. Архитектура нейронной сети

Изначально проводились эксперименты с обучением многозадачной нейронной сети, в этих экспериментах возникали проблемы со сходимостью, предположительно из-за того, что один общий участок сети использовался разными задачами, и не мог предоставить достаточно информативные признаки, из которых ответвления из нескольких слоев (специализирующиеся на конкретных задачах) получали бы предсказания. Также некоторую трудность составляла балансировка составной функции ошибки.

Для решения проблем с обучением многозадачной архитектуры мы воспользовались каскадным подходом, который упрощает сходимость, позволяет постепенно получить решения для разных задач путем добавления и дообучения дополнительных стадий каскада на основе ранее полученных признаков.

Поскольку также хотелось получить не очень большую и не очень медленную сеть, в качестве основного источника признаков изображения была взята архитектура MobileNetV2 [23], предобученная на задаче классификации на датасете Imagenet. Признаки входного изображения разных пространственных размеров конкатенировались, и объединялись с помощью свертки 1x1 (далее основные признаки будем обозначать *F*).

Каждая стадия каскада представляла собой архитектуру U-Net, которая получала на вход основные признаки *F* и предсказания, полученные ранними частями каскада.

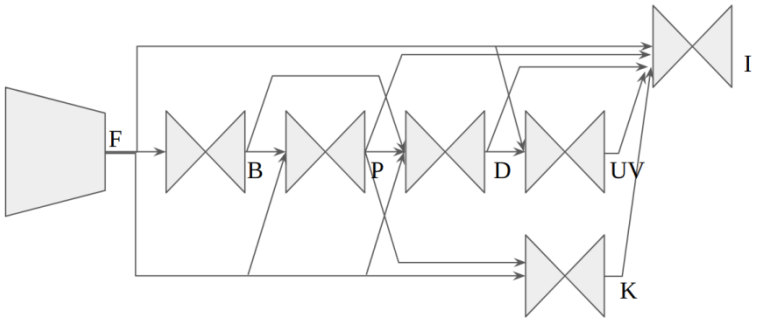


Рисунок 2.5

На рисунке 2.5 изображена общая схема предложенной архитектуры.

Архитектура сети U-Net, была несколько модифицирована для того чтобы увеличить рецептивное поле нейронов последних слоев. Для этого у сверточных слоев уровня с наименьшим разрешением последовательно экспоненциально увеличивался параметр дилатации, и полученные активационные карты этих слоев суммировались (рисунок 2.6).

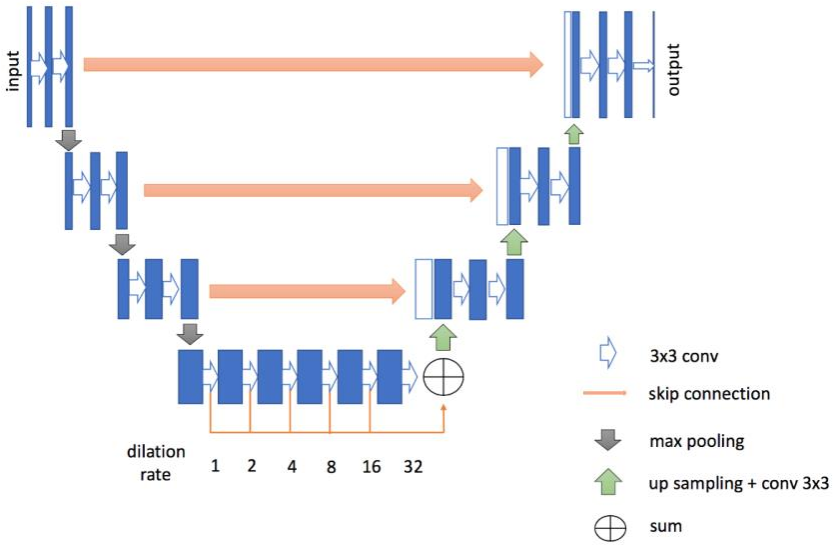


Рисунок 2.6

2.3. Процесс обучения

2.3.1. Используемые средства

При работе использовался язык программирования Python 3, различные пакеты для машинного обучения, компьютерного зрения, визуализации (Numpy, Scipy, Matplotlib, Keras, OpenCV, Scikit-Image и др.). В качестве бэкенда для фреймворка Keras использовался Tensorflow. Использовался менеджер пакетов Conda. Для аугментации данных использовался пакет ImgAug.

2.3.2. Стадия задач сегментации

Изначально обучались сети, отвечающие за предсказания бинарных масок и сегментации 14 частей тела (рисунок 2.7, обучаемые части выделены рамкой). В качестве функции ошибки использовалась сумма BCE и CCE для бинарной сегментации и сегментации 14 частей тела соответственно.

$$L = L_{bin} + L_{parts}$$

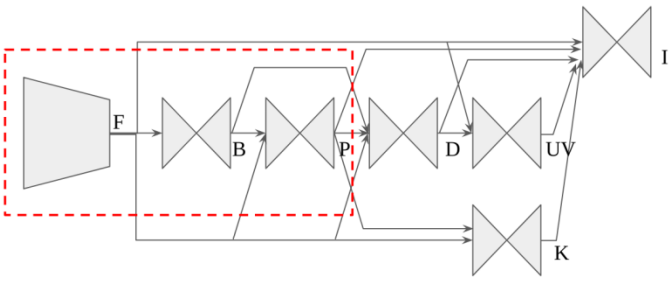


Рисунок 2.7

2.3.3. Стадия задачи предсказания попиксельного соответствия

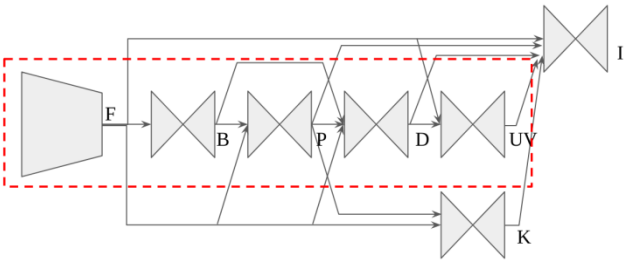


Рисунок 2.8

На рисунке 2.8 выделены части архитектуры, которые обучаются на второй стадии. В качестве функции ошибки для предсказаний 24 фрагментов развертки использовалась CCE, для регрессии UV координат использовался

логарифм гиперболического косинуса *log(cosh(x))*, функция ошибки, которая для небольших значений *x* приблизительно равна $(x^2)/2$ и равна примерно *abs(x) – log(2)* для больших *x* (такая функция ошибки ведет себя как MSE, но на нее будут меньше влиять редкие сильные ошибочные предсказания).

$$L = 2L_{bin} + 2L_{parts} + 20L_{patches} + 1.5 \times 10^4 L_U + 1.5 \times 10^4 L_V$$

2.3.4. Стадия задачи разделяющей сегментации объектов

На последней, третьей, стадии обучается вся архитектура целиком (рисунок 2.9).

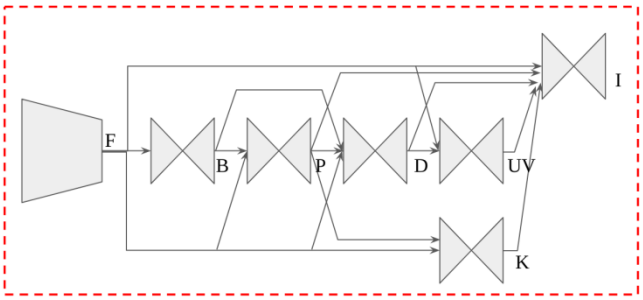


Рисунок 2.9

В функцию ошибки включены слагаемые для всех задач. Тепловые карты для ключевых точек суставов (где в позициях с суставами была размещена гауссиана с отклонением в 6 пикселей и высотой пика равной 1) предсказываются с помощью функции ошибки BCE, а векторные представления объектов с помощью логарифма гиперболического косинуса.

Для разделения объектов используется несколько модифицированный подход с Hough voting — для каждого человека на изображении, в его пикселях предсказывается вектор из центра изображения в точку головы. При подготовке данных создается маска, благодаря которой ошибка не считается с пикселей людей, для которых нет разметки точек на голове.

$$L = 2L_{bin} + 2L_{parts} + 20L_{patches} + 1.5 \times 10^4 L_U + 1.5 \times 10^4 L_V + 0.5L_{keypoints} + 0.01L_{instance}$$

2.4. Выводы

В этой главе были описаны особенности трехстадийного процесса обучения нейронной сети, функции ошибки, используемые при программной реализации средства.

ГЛАВА 3.
РЕЗУЛЬТАТЫ ЭКСПЕРИМЕНТОВ

3.1. Сегментация

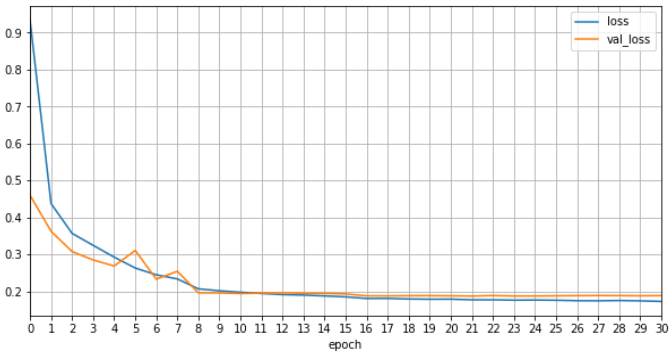


Рисунок 3.1

На рисунке 3.1 изображены графики функции ошибки на обучающей и валидационной выборках для первой стадии обучения. На рисунке 3.2 изображен график расписания learning rate.

Для обучения использовался оптимизатор RMSprop. Поскольку часть сети была предобучена на Imagenet, эти слои были заморожены в течение первых 4 эпох. Размер батча 16.

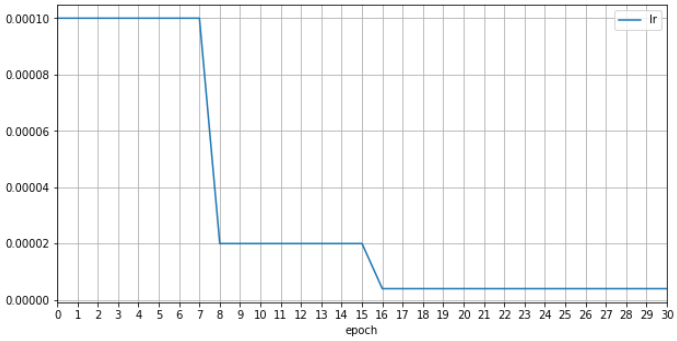


Рисунок 3.2

Далее, в таблице 3.1, приведены метрики IOU для сегментации людей целиком и частей тела (поскольку результаты были недостаточно хороши для надежного разделения объектов, то валидация проводилась на 600 изображениях, на которых присутствовал единственный человек с разметкой).

Таблица 3.1

	I	II	III
Binary	0.832	0.841	0.841
Torso	0.564	0.592	0.599
Right Hand	0.218	0.267	0.276
Left Hand	0.194	0.239	0.253
Left Foot	0.070	0.093	0.099
Right Foot	0.071	0.096	0.103
Upper Leg Right	0.230	0.263	0.266
Upper Leg Left	0.237	0.262	0.273
Lower Leg Right	0.150	0.183	0.190
Lower Leg Left	0.157	0.187	0.196
Upper Arm Left	0.299	0.335	0.342
Upper Arm Right	0.322	0.362	0.370
Lower Arm Left	0.232	0.281	0.294
Lower Arm Right	0.252	0.300	0.312
Head	0.684	0.702	0.704

На рисунке 3.3 изображен пример входных данных, разметки и предсказания сети.

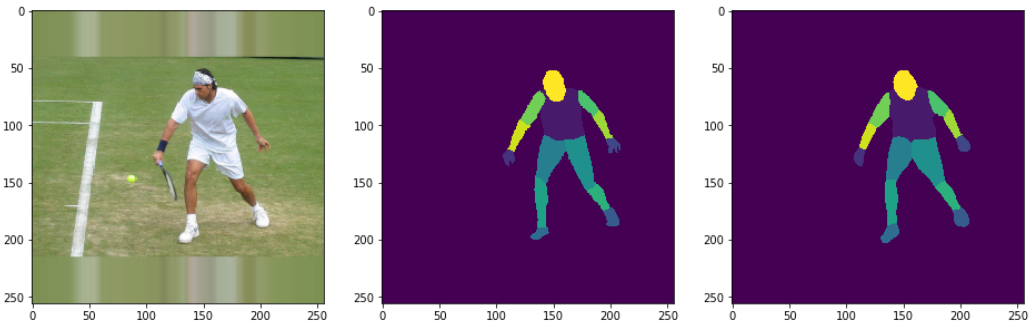


Рисунок 3.3

3.2. Предсказание плотного соответствия

На рисунке 3.4 изображены графики функции ошибки на обучающей и валидационной выборках для второй стадии обучения.

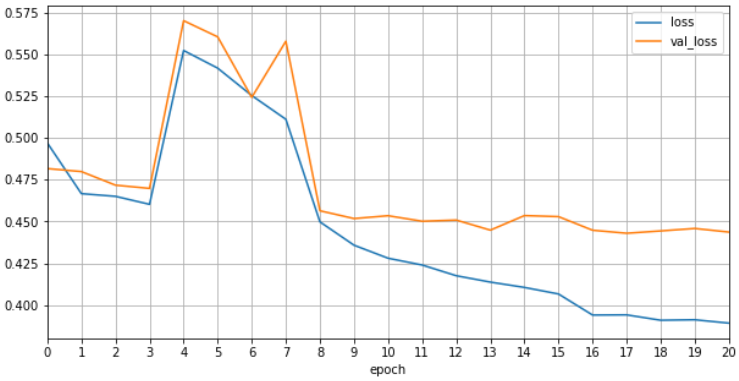


Рисунок 3.4

На рисунке 3.5 изображен график расписания learning rate.

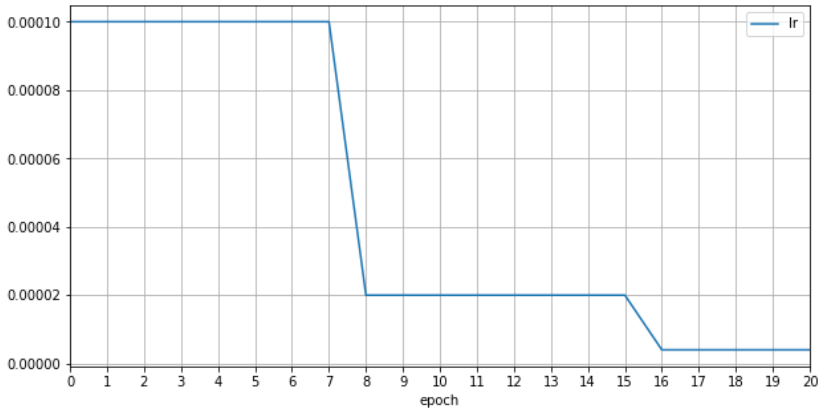


Рисунок 3.5

На рисунке 3.6 приведен пример переноса текстуры по полученным предсказаниям UV координат.

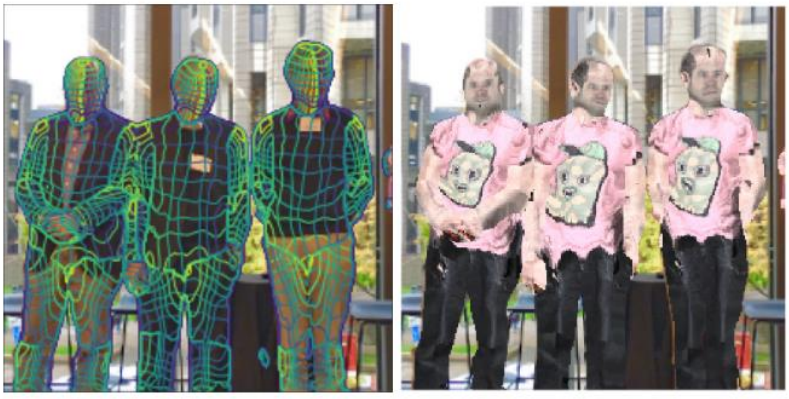


Рисунок 3.6

3.3. Разделяющая сегментация

На рисунке 3.7 изображены графики функции ошибки на обучающей и валидационной выборках для третьей стадии обучения.

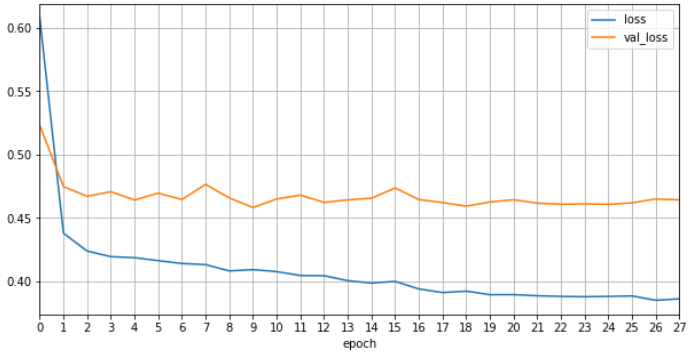


Рисунок 3.7

На рисунке 3.8 изображен график расписания learning rate.

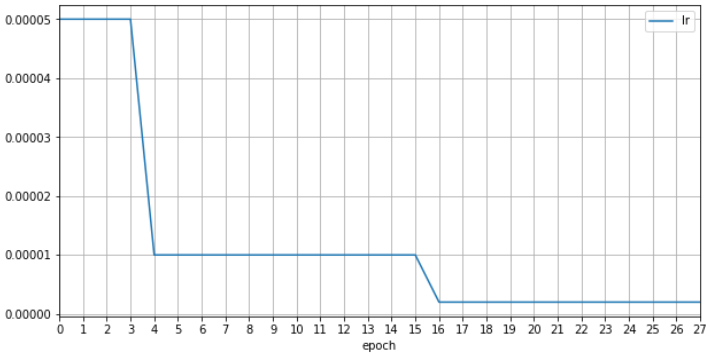


Рисунок 3.8

Эксперименты показали, что предложенный способ для разделения объектов, работает хуже, когда объекты на изображении расположены плотно или пересекаются. Эти вопросы требуют дополнительных исследований.

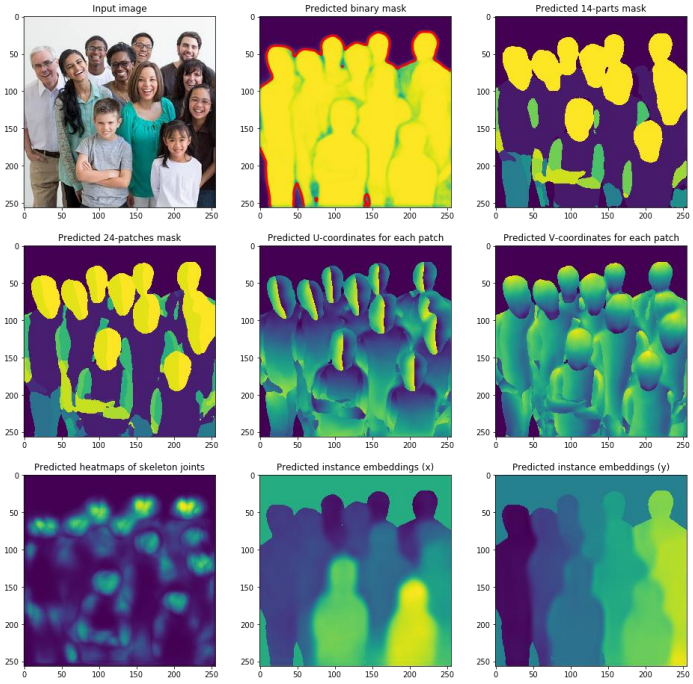


Рисунок 3.9

На рисунке 3.9 можно видеть все предсказания, полученные с помощью предложенной архитектуры. На рисунке 3.10 более подробно представлены результаты разделения объектов с помощью алгоритма кластеризации DBSCAN [24, 25]. Данный алгоритм находит сгустки точек высокой плотности, после чего наращивает из них кластеры. Данный алгоритм хорошо подходит для поиска кластеров (количество которых заранее неизвестно) схожей плотности. Также отмечают быстродействие данного алгоритма для задач кластеризации с большим количеством точек. Но стоит также отметить, что недостатком использования этого алгоритма является необходимость настраивать его параметры (*eps* и *min_samples*), отвечающие за инициализацию кластеров и их распространение. Для примера на рисунке использовались значения *eps* = 0.01 и *min_samples* = 100.

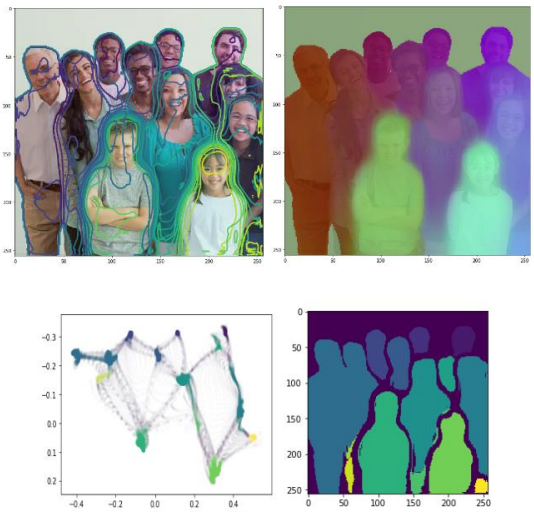


Рисунок 3.10

На рисунке 3.11 изображен пример, на котором кластеры разделены лучше, из-за того, что люди стоят не так плотно и не заслоняют друг друга.

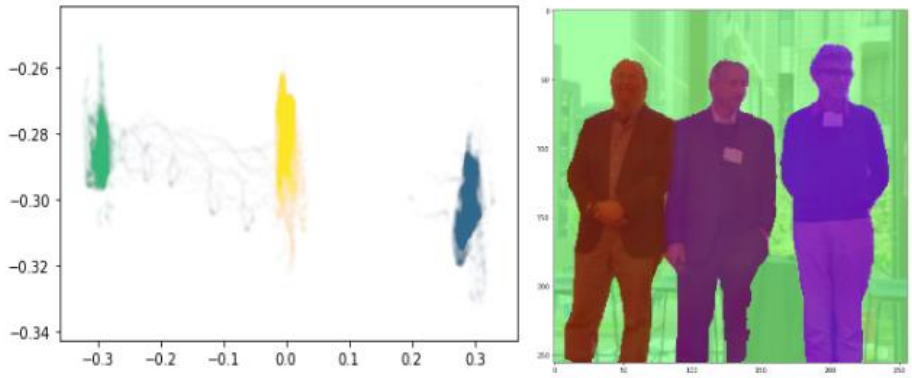


Рисунок 3.11

3.4. Выводы

В ходе работы была обучена нейронная сеть, выполняющая совместное решение задач сегментации людей, предсказания плотного соответствия, разделяющей сегментации объектов. Был проведен анализ результатов.

ЗАКЛЮЧЕНИЕ

Область визуального анализа изображений с людьми активно развивается в настоящее время и является актуальным направлением для исследований. В работе был проведен обзор релевантной литературы, на основе которого была разработана полносверточная каскадная нейросетевая архитектура, сочетающая предсказание плотного соответствия, разделяющей сегментации объектов и других задач. Использование каскадной архитектуры и постадийного обучения было вызвано проблемами со схождением модели, в которой задачи решались одновременно и параллельно, а не последовательно. В дальнейших исследованиях приоритетом будет являться разработка архитектуры с параллельными ветвями, что позволило бы уменьшить размер итоговой модели, отбросив ветви второстепенных задач после завершения обучения (поскольку предсказание плотного соответствия содержит в себе информацию о сегментации и видимом скелете). Но обучение дополнительным задачам, увеличивая обучающий сигнал, благотворно сказывается на качестве предсказаний, что видно по результатам для сегментации, а также неоднократно упоминается в литературе. Одним из ключевых моментов для запуска стабильного обучения являлся подбор коэффициентов, взвешивающих слагаемые ошибок для разных задач; а также генерация масок, обнуляющих ошибку в областях изображения в которых отсутствует разметка, или конфликтуют различные типы разметки (что позволило предсказывать плотное соответствие пикселей без необходимости предварительного обучения сети, интерполирующей разметку).

Также предложен новый простой способ для получения эмбеддингов пикселей для разделяющей сегментации объектов в виде предсказания вектора положения головы из каждого пикселя объекта. При наличии нужной разметки (ключевые точки и маски объектов) такой способ просто реализовать, однако он требует доработки для применения в случае анализа сложных сцен, где объекты расположены плотно, заслоняют друг друга, или, в особенности, где кодирующие объект целевые точки расположены близко. Дополнительными вопросами, заслуживающими рассмотрения, в данном случае могут являться способы повышения четкости границ между объектами, использование triplet loss, получение дополнительных предсказаний, позволяющих применение сегментации методом водораздела, увеличение количества целевых точек.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Toshev, A. DeepPose: Human Pose Estimation via Deep Neural Networks / A. Toshev, C. Szegedy // 2014 IEEE Conf. Comput. Vis. Pattern Recognit. – 2014. – С. 1653-1660.

2. Benjamin Sapp. MODEC: Multimodal Decomposable Models for Human Pose Estimation / Benjamin Sapp, B. Taskar // CVPR. – 2013.

3. Johnson, S. Clustered Pose and Nonlinear Appearance Models for Human Pose Estimation. / S. Johnson, M. Everingham // bmvc / Citeseer. – 2010. – Т. 2. – С. 5.

4. Newell, A. Stacked hourglass networks for human pose estimation / A. Newell, K. Yang, J. Deng // European conference on computer vision / Springer. – 2016. – С. 483–499.

5. Ronneberger, O. U-net: Convolutional networks for biomedical image segmentation / O. Ronneberger, P. Fischer, T. Brox // International Conference on Medical image computing and computer-assisted intervention / Springer. – 2015. – С. 234–241.

6. Deep residual learning for image recognition / K. He [и др.] // Proceedings of the IEEE conference on computer vision and pattern recognition. – 2016. – С. 770–778.

7. Ioffe, S. Batch normalization: Accelerating deep network training by reducing internal covariate shift / S. Ioffe, C. Szegedy // ArXiv Prepr. ArXiv150203167. – 2015.

8. 2d human pose estimation: New benchmark and state of the art analysis / M. Andriluka [и др.] // Proceedings of the IEEE Conference on computer Vision and Pattern Recognition. – 2014. – С. 3686–3693.

9. Cascaded pyramid network for multi-person pose estimation / Y. Chen [и др.] // Proceedings of the IEEE conference on computer vision and pattern recognition. – 2018. – С. 7103–7112.

10. Feature pyramid networks for object detection / T.-Y. Lin [и др.] // Proceedings of the IEEE conference on computer vision and pattern recognition. – 2017. – С. 2117–2125.

11. Mask r-cnn / K. He [и др.] // Proceedings of the IEEE international conference on computer vision. – 2017. – С. 2961–2969.

12. Microsoft coco: Common objects in context / T.-Y. Lin [и др.] // European conference on computer vision / Springer. – 2014. – С. 740–755.

13. Imagenet: A large-scale hierarchical image database / J. Deng [и др.] // Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on / IEEE. – 2009. – С. 248–255.

14. Xiao, B. Simple baselines for human pose estimation and tracking / B. Xiao, H. Wu, Y. Wei // Proceedings of the European conference on computer vision (ECCV). – 2018. – С. 466–481.

15. Faster r-cnn: Towards real-time object detection with region proposal networks / S. Ren [и др.] // Advances in neural information processing systems. – 2015. – С. 91–99.

16. Long, J. Fully convolutional networks for semantic segmentation / J. Long, E. Shelhamer, T. Darrell // Proceedings of the IEEE conference on computer vision and pattern recognition. – 2015. – С. 3431–3440.

17. Simonyan, K. Very deep convolutional networks for large-scale image recognition / K. Simonyan, A. Zisserman // ArXiv Prepr. ArXiv14091556. – 2014.

18. Agarap, A.F. Deep learning using rectified linear units (relu) / A.F. Agarap // ArXiv Prepr. ArXiv180308375. – 2018.

19. Bonde, U. Towards Bounding-Box Free Panoptic Segmentation / U. Bonde, P.F. Alcantarilla, S. Leutenegger // ArXiv Prepr. ArXiv200207705. – 2020.

20. Alp Güler, R. Densepose: Dense human pose estimation in the wild / R. Alp Güler, N. Neverova, I. Kokkinos // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. – 2018. – С. 7297–7306.

21. Densereg: Fully convolutional dense shape regression in-the-wild / R. Alp Guler [и др.] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. – 2017. – С. 6799–6808.

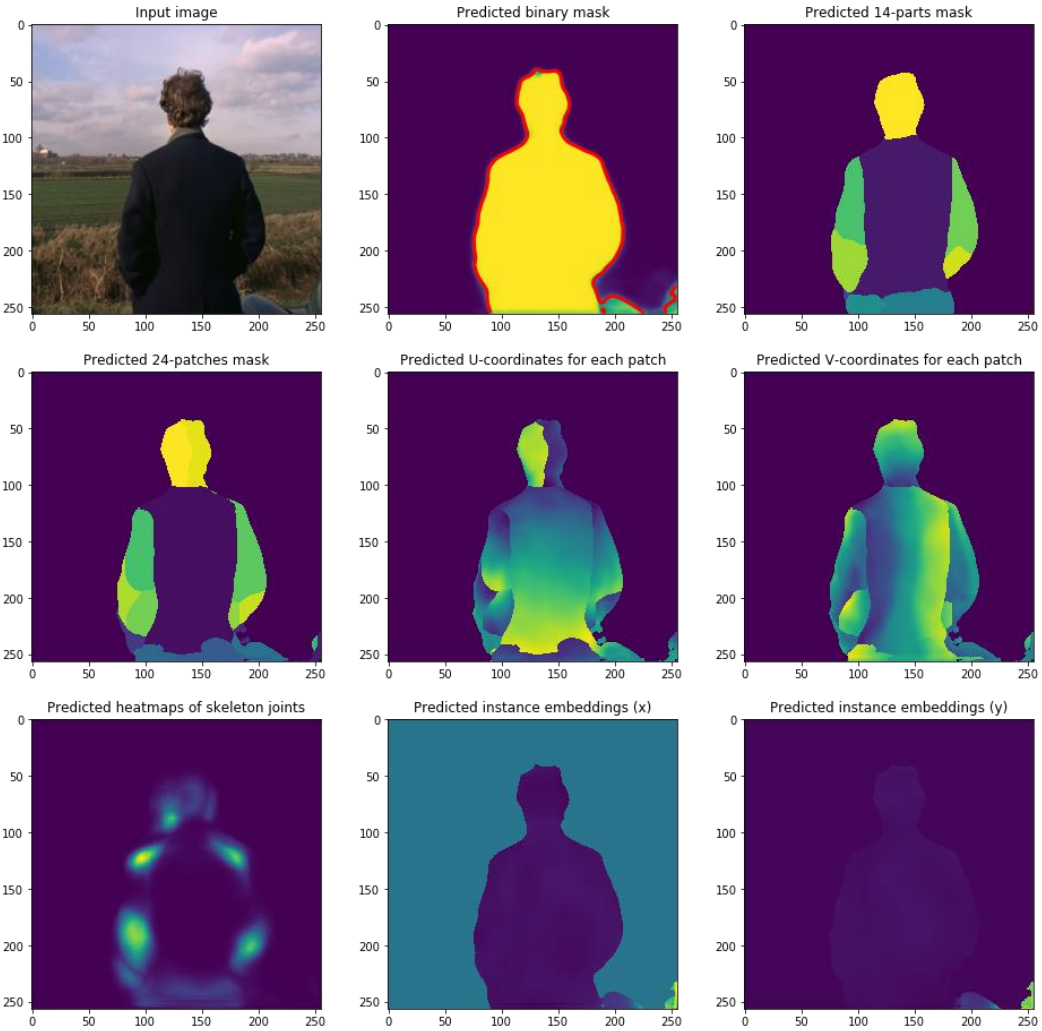
22. SMPL: A skinned multi-person linear model / M. Loper [и др.] // ACM Trans. Graph. TOG. – 2015. – Т. 34, № 6. – С. 1–16.

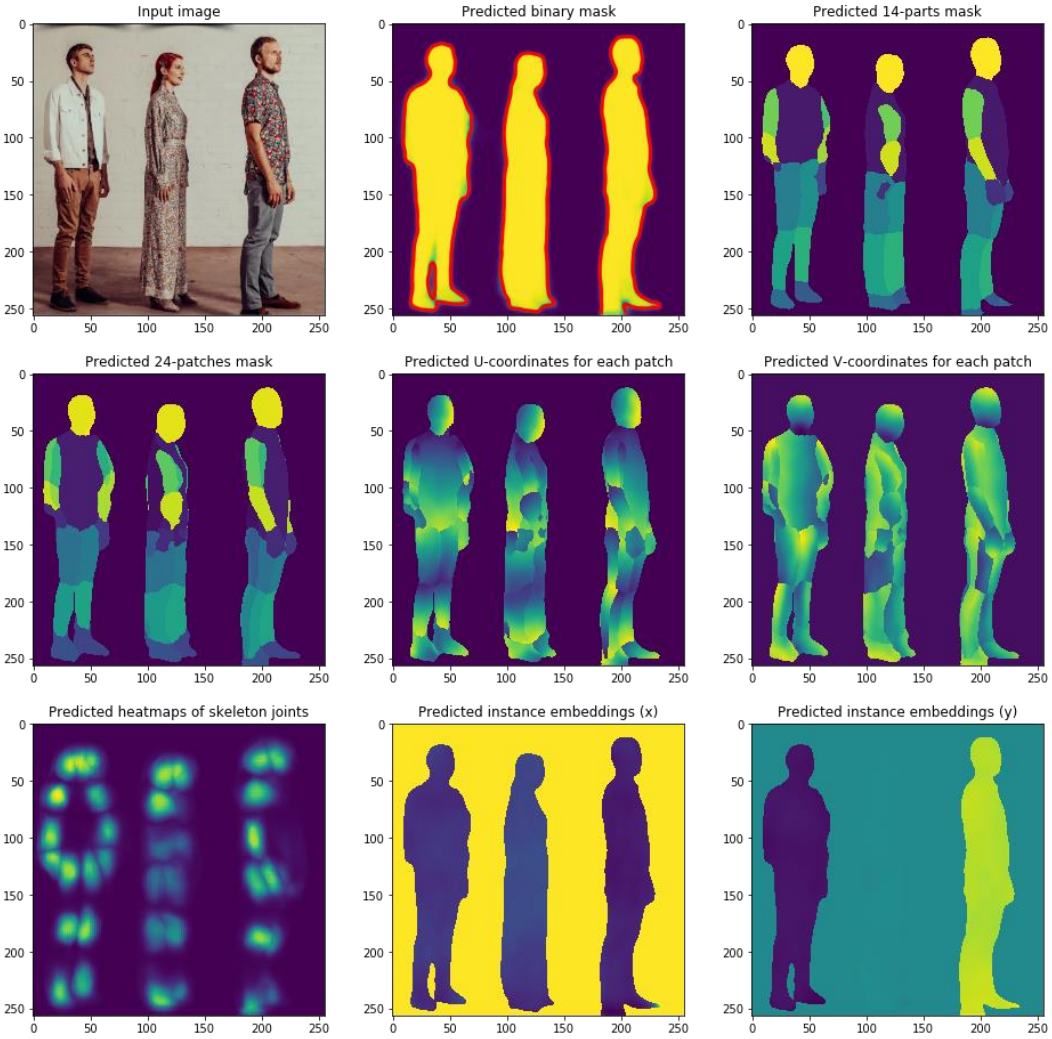
23. Mobilenetv2: Inverted residuals and linear bottlenecks / M. Sandler [и др.] // Proceedings of the IEEE conference on computer vision and pattern recognition. – 2018. – С. 4510–4520.

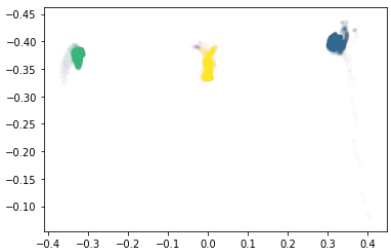
24. A density-based algorithm for discovering clusters in large spatial databases with noise. / M. Ester [и др.] // Kdd. – 1996. – Т. 96. – С. 226–231.

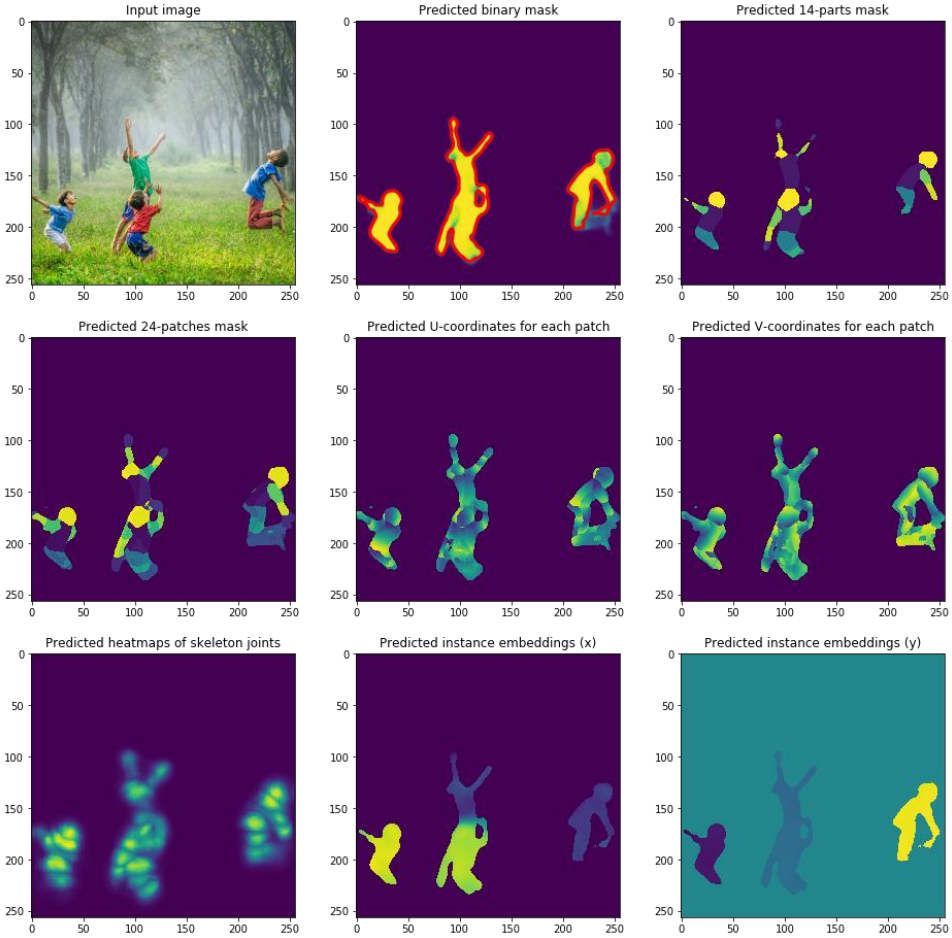
25. DBSCAN revisited, revisited: why and how you should (still) use DBSCAN / E. Schubert [и др.] // ACM Trans. Database Syst. TODS. – 2017. – Т. 42, № 3. – С. 1–21.

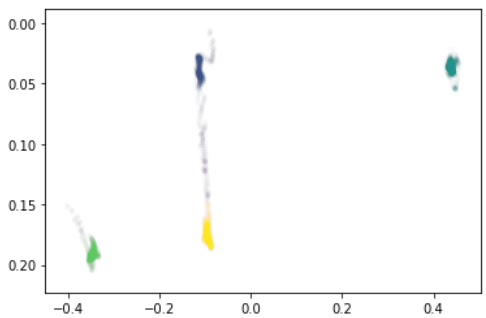
ПРИЛОЖЕНИЕ

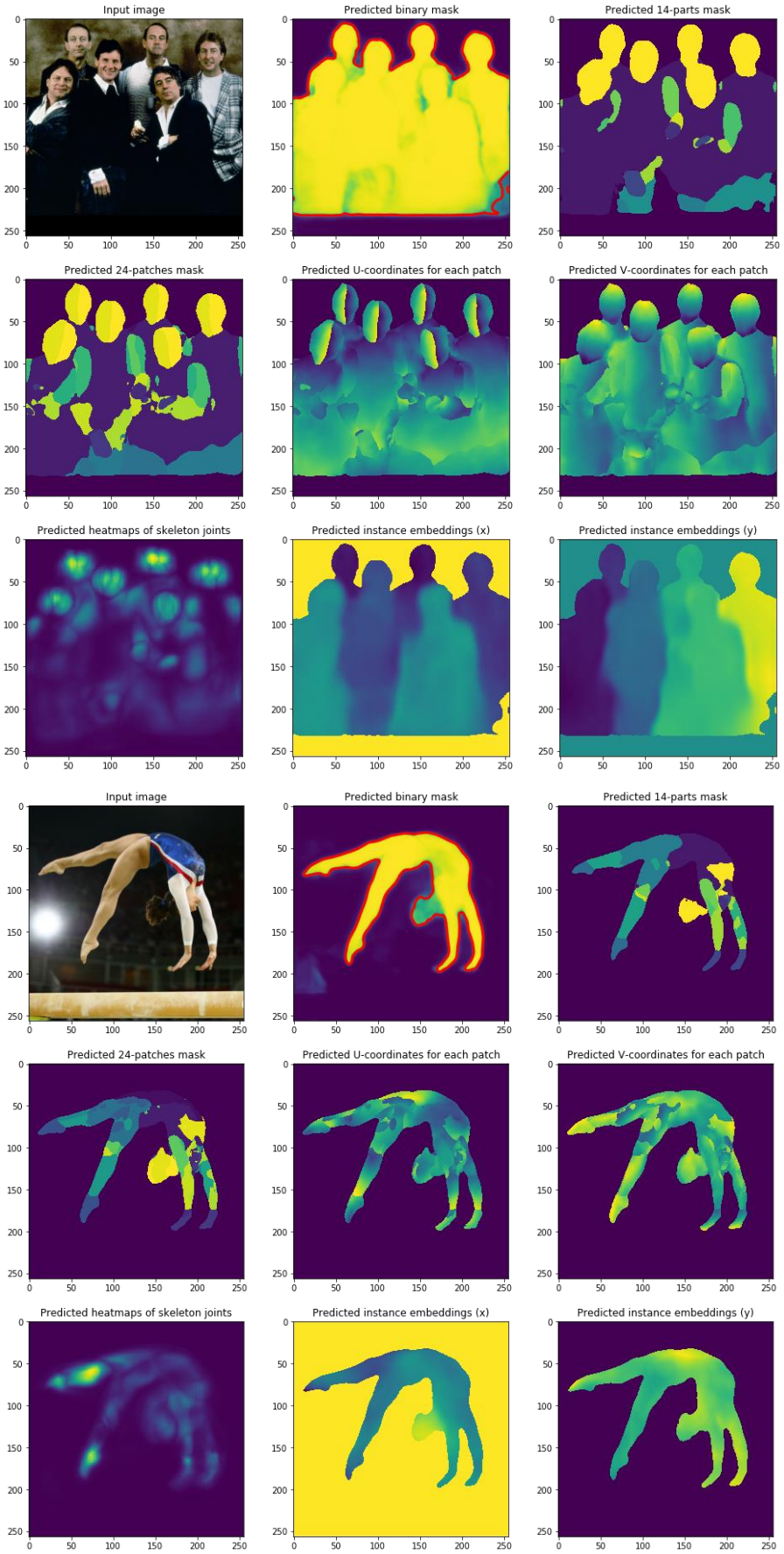

Input image
Predicted binary mask
Predicted 14-parts mask
Predicted 24-patches mask
Predicted U-coordinates for each patch
Predicted V-coordinates for each patch
Predicted heatmaps of skeleton joints
Predicted instance embeddings (x)
Predicted instance embeddings (y)


Input image
Predicted binary mask
Predicted 14-parts mask
Predicted 24-patches mask
Predicted U-coordinates for each patch
Predicted V-coordinates for each patch
Predicted heatmaps of skeleton joints
Predicted instance embeddings (x)
Predicted instance embeddings (y)




Input image
Predicted binary mask
Predicted 14-parts mask
Predicted 24-patches mask
Predicted U-coordinates for each patch
Predicted V-coordinates for each patch
Predicted heatmaps of skeleton joints
Predicted instance embeddings (x)
Predicted instance embeddings (y)


0.00
0.05
0.10
0.15
0.20
−0.4
−0.2
0.0
0.2
0.4


Input image
Predicted binary mask
Predicted 14-parts mask
Predicted 24-patches mask
Predicted U-coordinates for each patch
Predicted V-coordinates for each patch
Predicted heatmaps of skeleton joints
Predicted instance embeddings (x)
Predicted instance embeddings (y)
Input image
Predicted binary mask
Predicted 14-parts mask
Predicted 24-patches mask
Predicted U-coordinates for each patch
Predicted V-coordinates for each patch
Predicted heatmaps of skeleton joints
Predicted instance embeddings (x)
Predicted instance embeddings (y)