**Аскаров Е.А., Бектемесов А.Т.**
*Университет Туран, Казахстан*

# РАСПОЗНАВАНИЕ ЖЕСТОВ РУК НА ОСНОВЕ НЕЙРОННЫХ СЕТЕЙ

Аннотация.В современном обществе технологии искусственного интеллекта играют все более важную роль. Хорошо известно, что мы можем использовать глубокое обучение для распознавания человеческих жестов, что очень полезно для нас. Например, если наша система обнаружит студента, который играет в свой мобильный телефон в классе, система даст ему очень низкий балл за эту лекцию. Другой пример: если некоторые учащиеся внимательно делают записи, наша система выставит им очень высокие баллы.

Ключевые слова**:** компьютерное зрение, оптическое распознавание, Python 3, OpenCV, OpenPose, машинное обучение.

Введение

Оценка двухмерных поз в реальном времени с использованием полей сродства частей - очень актуальная тема. Суть этого проекта состоит в том, чтобы предложить алгоритм оценки позы человеческого тела, который использует восходящий алгоритм Part Affinity Fields (PAFs) (получение позиции ключевой точки для получения скелета) вместо традиционного нисходящего алгоритма (сначала обнаружение людей и затем вернитесь к ключевой точке). Мы узнали об OpenPose, который представляет собой обнаружение ключевых точек CMU с открытым исходным кодом для нескольких человек в реальном времени, включая ключевые точки человека, ключевые точки рук, обнаружение ключевых точек лица и оценку позы. Мы решили использовать OpenPose для решения проблемы распознавания жестов в облачной среде класса. Получив ключевые точки человеческого тела, мы смогли получить некоторые ключевые углы, вычислив позиционное соотношение между ключевыми точками, и использовать эти углы и положения для оценки жеста ученика на видео. Система могла определять шесть наиболее распространенных жестов в классе: слушание, создание заметок, игра в мобильный телефон, сон, поднятие руки и стояние. Хотя код может работать очень быстро на сервере, он работает медленнее на моем ноутбуке из-за плохой конфигурации моего ноутбука, и на моем ноутбуке нет возможности делать выводы в реальном времени. Мы попытались совместить Openpose с YOLO, чтобы система работала быстрее, но до сих пор нет способа решить проблему низкой скорости на нашем ноутбуке, а среднее время расчета одного кадра составляло более 1 секунды. Чтобы добиться лучших результатов, мы изменили эту систему реального времени на систему, которая распознает каждые 3 секунды. Наконец, мы объединили часть распознавания выражения лица с частью распознавания жеста. Система может выполнять распознавание каждые 3 секунды и записывать результаты работы каждого ученика, а затем отправлять их учителю в режиме реального

времени. Учителям удобно адаптировать свой стиль преподавания к успеваемости учеников. Более того, система также генерирует отчет об успеваемости учащегося и оценку успеваемости в классе, чтобы реализовать интерактивное распознавание поведения.

Почему хороший набор данных нужен как воздух?

Общие объекты в контексте (COCO): мы используем набор данных COCO, чтобы иметь дело с частью распознавания жестов в системе облачного класса. Набор данных COCO используется для обнаружения ключевых точек OpenPose в проекте. Набор данных COCO, подготовленный Microsoft, представляет собой большой набор данных изображений, предназначенный для обнаружения объектов, сегментации, определения ключевых точек человека, семантической сегментации и создания заголовков. Целью этого набора данных является понимание сцены, которое в основном захватывается из сложных повседневных сцен. Цели на изображении отмечены точной сегментацией. Набор данных COCO включает большое количество изображений, которые более 200000, и 250000 экземпляров людей в этом наборе данных отмечены ключевыми точками. Более того, большинство людей в наборе данных COCO относятся к средним и крупным масштабам. Некоторые примеры набора данных COCO показаны на рисунке 1. [1]

Рисунок 1: Некоторые примеры набора данных COCO

Библиотека OpenPose

OpenPose - это хорошо известная библиотека с открытым исходным кодом, основанная на сверточной нейронной сети и контролируемом обучении. Он может отслеживать выражение лица человека, туловища, конечностей и даже пальцев. Этот алгоритм вполне подходит для обнаружения как одного, так и нескольких человек. Этот алгоритм взят из статьи CVPR 2017, написанной Цао Чжэ из Лаборатории перцепционных вычислений CMU «Оценка двухмерных поз в реальном времени с использованием полей сродства частей». [2]

В OpenPose мы должны ввести изображение, пропустить его через магистраль (такую как VGG, Res-Net, Mobile-Net), а затем пройти 6 этапов.

На каждом этапе есть две ветви: одна для обнаружения тепловой карты, а другая для обнаружения векторной карты. С помощью тепловой карты и векторной карты вы можете узнать все ключевые точки на картинке, а затем отметить точки для всех с помощью PAF. Платформа OpenPose показана на рисунке 2.

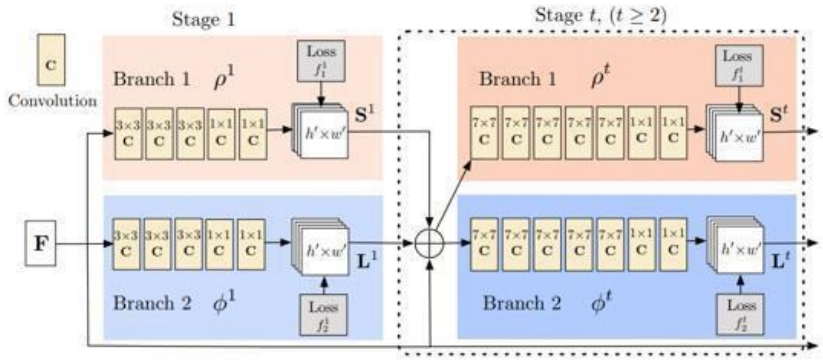


Рисунок 2: Фреймворк OpenPose

Нам нужно использовать OpenPose, чтобы реализовать распознавание ключевых точек человеческого тела, а затем оценить позу человека на картинке по позиционным отношениям между отмеченными ключевыми точками. Кроме того, система может отправлять обнаруженный жест учеников учителю и оценивать жесты учеников в это время, чтобы отразить успеваемость каждого ученика в классе. Это может помочь учителям иметь интуитивное представление об успеваемости учащихся.

OpenPose может оценивать позу одного человека на изображении, а также может обрабатывать оценку позы нескольких людей на изображении. Входными данными этой модели является изображение h × w × 3, и эта модель может выводить два массива, содержащих карты достоверности ключевых точек и тепловые карты сродства частей каждой пары ключевых точек. Верхние 10 слоев VGG19 используются для извлечения карт характеристик входного изображения. Затем используется двухуровневая многоступенчатая структура CNN. Сетевая структура OpenPose показана на рисунке 3.

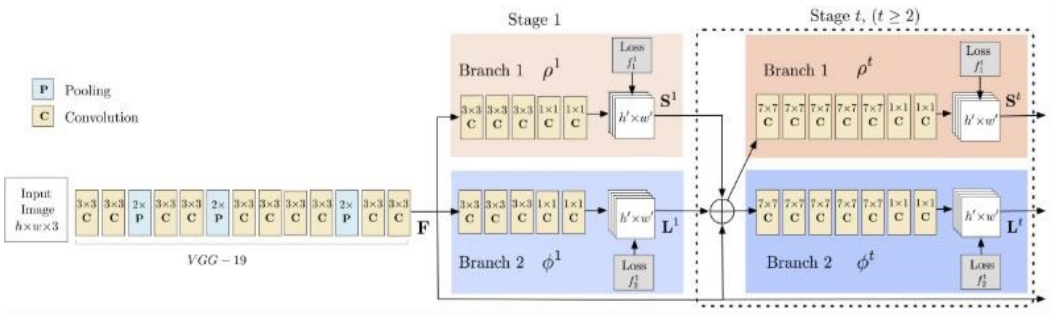


Рисунок 3: Сетевая структура OpenPose

После получения входного изображения мы использовали первые 10 слоев сети VGG19 для извлечения функций входного изображения. Одним из улучшений VGG по сравнению с AlexNet является использование нескольких

последовательных 3 × 3 ядра свертки вместо больших ядер свертки в AlexNet (11 × 11, 7 × 7, 5 × 5). Главная его цель - улучшить глубину сети и улучшить эффект нейронной сети до определенной степени при условии обеспечения того же поля восприятия. Например, слой суперпозицию трех ядер свертки 3 × 3 с шагом 1 можно рассматривать как рецептивное поле размера 7, что означает, что три непрерывных свертки 3 × 3 эквивалентны свертке 7 × 7. [3] Общее количество параметров трех непрерывных сверток 3 × 3 равно 3 × (3 × 3 × $C2$). Если 3 × 3 ядро свертки используется напрямую, количество общих параметров 7 × 7 × $C2$, где C относится к количество каналов. Очевидно, что 27 × $C2$ меньше 49 × $C2$, что означает, что три непрерывных 3 × 3 свертки могут уменьшить параметры; а ядро свертки 3 × 3 помогает лучше поддерживать свойства изображения. Более того, мы использовали 3 нелинейные функции вместо 1 свертки, что увеличило различающая способность функции. Структура VGG19 показана на рисунке 4. [4]

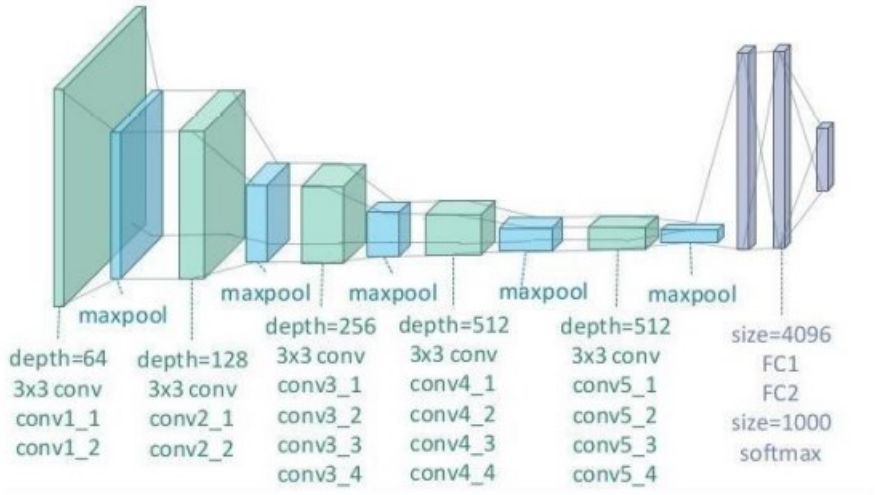


Рисунок 4: Каркас VGG19

После первых 10 уровней сети VGG19 мы можем получить функцию F. Функция F обрабатывается через непрерывную многоступенчатую сеть. Каждая фаза (t) сети содержит две ветви, и входными результатами являются S (t) (карта достоверности детали) и L (t) (карта соответствия детали). S (t) сообщает нам, где находится голова и где локоть; L (t) говорит нам, какие места определенно находятся на какой ноге. С помощью L (t) Координатные точки S (t) соединяются, образуя каркас позы человека. Входные данные, полученные на первом этапе сети, - это признак F. Признак F обрабатывается сеть, чтобы получить S (1) и L (1) соответственно. Начиная с фазы (2), вход в сеть фазы (t) состоит из трех частей: S (t − 1), L (t − 1) и признак F. Входы в сеть для каждой фазы:

$$St = \rho t\ (F, St - 1, Lt - 1), \forall t \geq 2$$

$$Lt = \varphi t\ (F, St - 1, Lt - 1), \forall t \geq 2$$

Чтобы определить, является ли сеть конвергентной, мы определяем функцию потерь сети:

$$\mathbf{f} = \sum fS\ \mathbf{N}\ t$$

$$t=1 + fL$$

$fS$ и $fL$ представляют условия ошибки двух выходных изображений соответственно. Их формулы:

$$fS\ t = \sum \sum W(P) \cdot Sj\ t\ (p) - Sj * (p)$$

$$fL\ t = \sum \sum W(P) \cdot Lc\ t\ (p) - Lc * (p)$$

t представляет стадию.

W представляет двоичную маску. Если W (p) = 0, это означает, что текущая точка p отсутствует (не видна или не на изображении) во избежание неправильного наказания во время тренировки. Наконец, мы можем соединить ключевые точки и торсы, чтобы завершить требуемую модель. Мы можем получить все части тела и туловища, а также два соседних торса должны иметь общие точки соединения. Объединив все туловища через суставные точки, мы можем получить скелет тела всех людей. Благодаря обучению работе с конечной сетью я получил модель: model.h5. Мне нужно прочитать нейронную сеть, затем отрегулируйте размер выходных данных, чтобы они были такими же, как входные, а затем проверьте карту достоверности ключевого момента. Мы должны сохранить координаты (x, y) и оценки вероятности для каждой ключевой точки, и выполнить обнаружение ключевых точек на входном изображении. А затем используйте Heatmap, чтобы найти допустимое соединение. пара. Наконец, объедините все ключевые точки, принадлежащие одному человеку, чтобы нарисовать каркасную карту. В части распознавания жеста нам нужно распознать шесть видов жестов: сидение, принятие примечание, играя в сотовый телефон, спит, поднимает руку и стоит. Чтобы реализовать распознавание этих жестов, мы можем записать координаты ключевых точек человеческого тела, а затем вычислить позиционные отношения между различными частями человеческого тела через координаты каждой клавиши точка. Нам нужно только знать координаты трех точек, чтобы узнать угол, образованный тремя сторон, а затем используйте диапазон значений этих углов, чтобы вывести позу человека на изображении.

Наконец, через OpenCV я могу использовать камеру, чтобы записывать выступления студентов перед экран в облачном классе, затем отправьте жест ученика учителю и запишите оценку студенческие жесты. Я использую OpenPose для распознавания человеческих ключевых точек. Благодаря этой модели мы можем получить координаты 18 ключевых точек человеческого тела. Мы сравнили наш подход с некоторыми из методы, предложенные предыдущими людьми, и результат показан в таблице 1. Путем сравнения Результаты экспериментов в таблице показывают, что используемый нами метод имеет очень значительную повышение точности распознавания по сравнению с другими методами.

| | Голова | Плечо | Локоть | Запясть | Тазобедренный сустав | Колено | Лодыжк | Карта |
|---|---|---|---|---|---|---|---|---|
| Deepcut | 73.4 | 71.8 | 57.9 | 39.9 | 56.7 | 44.0 | 32.0 | 54.1 |
| Iqbal et al. | 70.0 | 65.2 | 56.2 | 46.1 | 52.7 | 47.9 | 44.5 | 54.7 |
| DeeperCut | 87.9 | 84.0 | 71.9 | 63.9 | 68.8 | 63.8 | 58.1 | 71.2 |
| Наш метод | 94.1 | 91.2 | 81.5 | 72.7 | 78.1 | 72.9 | 67.9 | 80.2 |

Таблица 1: Сравнение некоторых различных методов

Наша последняя система должна распознавать только некторые жесты. Однако в наборе данных COCO нет данных, специально используемые для оценки жестов учащихся в классе, поэтому мы сделали 200 фотографий с помощью камеры чтобы проверить точность этих жестов.

Вывод

Мы решили вопрос с предварительную обработку набора данных. Изначально мы планировали использовать набор данных EgoHands, но обнаружили, что это не соответствует нашим потребностям, и позже переключился на набор данных COCO. А потом мы использовали OpenPose которая представляет собой библиотеку с открытым исходным кодом, основанную на сверточных нейронных сетях и контролируемом обучении и разработан с Caffe в качестве основы для определения ключевых точек человека. После получения ключевых моментов человеческое тело, мы смогли найти угол, образованный каждой частью человеческого тела, через координаты каждой ключевой точки, а затем сделайте вывод об позе человека на изображении. В конце концов, наша система может распознавать четыре человеческие позы с точностью до 75%.

Литература:


1) COCO dataset, (2015). COCO 2018 Keypoint Detection Task. [online] Available from: http://cocodataset.org/#overview [Accessed 5th April 2019].
2) Cao, Z., Simon, T., Wei, S. and Sheikh, Y. (2017). Realtime multi-person 2d pose estimation using part affinity fields. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 7291-7299.
3) Andrew, NG. (2018). CS229: Machine Learning. [Stanford University]. San Francisco.
4) Fang, H., Xie, S., Tai, Y. and Lu, C. (2017). RMPE: Regional multi-person pose estimation. 2017 IEEE International Conference on Computer Vision. pp. 2334-2343.