УДК 004.9

Алгоритм распознавания жестов русского языка на видео

Э. В. Куликова
E-mail: elina.kulikova.2018@inbox.ru
Ярославский государственный университет им. П.Г. Демидова

Аннотация

Распознавание жестов рук является важной темой в области компьютерного зрения и машинного обучения. Эта технология имеет широкий спектр применений, таких как управление умными устройствами, взаимодействие человека и робота, а также помощь людям с ограниченными возможностями.

В результате исследования был разработан алгоритм распознавания жестов рук на видео, который позволяет определять и классифицировать некоторые цифры и буквы русского жестового языка. Алгоритм был реализован на языке программирования Python с использованием библиотек OpenCV, MediPipe и TensorFlow. Кроме того, самостоятельно был собран корпус данных для русского жестового языка. Была проведена оценка качества алгоритма на основе анализа его производительности и посчитанных метрик на реальных видео.

Ключевые слова: Распознавание жестов на видео, классификация жестов, язык жестов, русский жестовый язык.

Введение

Сообщества с нарушениями слуха часто используют язык жестов, который представляет собой систему, использующую визуально-мануальную модальность для передачи смысла. Язык жестов зависит в основном от жестов рук, движений тела и выражения лица. Распознавание языка жестов (SLR) — сложная задача, особенно распознавание динамических знаков, зависящих от движения. Вот почему многие исследователи заинтересованы в разработке приложения SLR с целью уменьшения барьера между сообществом с нарушениями слуха и обществом [1].

Одной из ключевых задач сейчас является возможность точного отслеживания и интерпретации жестов рук в режиме реального времени, особенно на видео.

Существует несколько ключевых проблем связанных с распознаванием жестов рук в видеорядах: руки постоянно находятся в движении, меняют углы наклона и перекрывают друг друга, изменчивость формы и размеров рук. Всё это может затруднить создание универсального алгоритма.

Кроме того, жесты рук сильно зависят от контекста, а это означает, что один и тот же жест может иметь разное значение в зависимости от ситуации. Например, жест рукой, означающий «стоп» в одном контексте, может означать «подойти» в другом.

В этой статье проводятся исследования некоторых ключевых методов и алгоритмов, используемых для распознавания жестов рук на видео, а также будет рассмотрено их потенциальное применение в области русского жестового языка.

Анализ существующих технологий

Существует два типа распознавания жестов рук: на основе носимых перчаток и на основе машинного зрения. Минус первого метода в том, что он дорог и требует и требует ношения на руке специального устройства для распознавания жестов, а также нестабилен в некоторых средах. Второй метод основан на обработке изображений, где последовательность операций выполняется следующим образом: захват изображения с помощью веб–камеры, сегментация, извлечение признаков и классификация жестов [2].

Методики распознавания языка жестов на основе компьютерного зрения принято делить на две категории: статические и динамические. Статические признаки — это те, которые требуют обработки только одного изображения на входе классификатора, то есть его можно рассматривать как картинку формы руки. Динамические знаки можно рассматривать как видео, содержащее ряд последовательных кадров для построения знака. Как правило, в языке жестов знаки строятся из серии быстрых движений рук и выражений тела. Следовательно, статическое распознавание не является хорошим решением проблем языка жестов, поскольку оно не может справиться с вариациями знаков. Значит, динамическое решение является более эффективным и действенным [1].

Многие исследователи использовали методы извлечения признаков вручную с алгоритмами машинного обучения для классификации жестов рук, но в последнее время в большинстве разработок использовались методы глубокого обучения. Изначально для распозна-

вания жестов рук применялись свёрточные нейронные сети (CNN), но с помощью них было трудно распознать динамические жесты рук, содержащие пространственно-временную информацию. Некоторые исследователи использовали рекуррентные нейронные сети (RNN), которые в основном похожи на CNN, хотя свёрточные нейронные сети оказались более успешными. Недавно разработчики использовали долговременную кратковременную память (LSTM) для извлечения долговременной зависимости. Комбинация CNN и LSTM использовалась для достижения высокой точности распознавания жестов рук. Долгосрочная зависимость требует высокой сложности вычислений, что является основной проблемой LSTM. Нейронные сети, основанные на внимании, напротив, создают кратковременную зависимость, которая требует меньшей вычислительной сложности [2].

Таким образом, учитывая все особенности и сложности распознавания и классификации жестов рук на видео в реальном времени, было решено провести несколько экспериментов с разными моделями нейронных сетей для извлечения признаков и сравнить их результаты.

Сбор корпуса данных

Поскольку не существует общедоступного и общепринятого набора данных для цифр и букв русского жестового языка, то он был создан самостоятельно с помощью добровольцев. Инструкции о том, как показывать жесты были взяты из проекта «Словарь. Русский жестовый язык» (https://surdo.me).

Сбор корпуса данных для русского жестового языка включал в себя несколько этапов:

1. Цифры. На первом этапе записывались на видео цифры русского жестового языка. Участвовало 19 добровольцев, из которых 10 женщин и 9 мужчин в возрасте от 20 до 55 лет. Добровольцы показывали цифры от 1 до 10 на правой и левой руке с помощью жестов. Было записано 38 видео и взято одно видео с цифрами от 1 до 10 из проекта «Словарь. Русский жестовый язык».
2. Буквы. На втором этапе записывались на видео буквы русского жестового языка. Участвовало 11 добровольцев, из которых 7 женщин и 4 мужчин в возрасте от 20 до 55 лет. Добровольцы показывали буквы русского алфавита с помощью жестов. Было записано 10 видео и взято одно видео из проекта «Словарь. Русский жестовый язык».

3. На третьем этапе все видео были подвергнуты раскадровке и вручную отобраны лучшие фотографии, где чётко видно жест и он показан достоверно. Для последующей классификации жесты были разбиты на классы. Для цифр классы с наименованием от 1 до 10, которые соответствуют цифрам, и 25 классов для букв с их наименованием соответственно.

Дополнительно были сняты 2 видео для проверки качества распознавания и классификации жестов в реальных условиях эксплуатации.

Обработка корпусов данных

Оба корпуса данных с цифрами и буквами русского жестового языка (RSL) были переработаны решением «MediaPipe Hand Landmarker». Оно является частью проекта «MediaPipe» — это фреймворк с открытым исходным кодом, представленный Google, который помогает создавать мультимодальные конвейеры машинного обучения [3]. Задача «MediaPipe Hand Landmarker» позволяет обнаружить ориентиры рук на изображении. Его можно использовать для локализации ключевых точек рук и визуализации ориентиров [3].

Было решено разделить каждый набор данных на два. Один содержит документы в формате JSON (рис. 1), где каждая фотография набора представляет собой следующую информацию: сколько рук обнаружено на фото и массив данных длинной 43 элемента. Массив содержит ключевые точки обнаруженных рук в формате x, y и z: 21 ориентир для правой руки, 21 ориентир для левой руки и точка соотношения положений двух рук относительно друг друга. Если на фотографии обнаружена одна рука, то элементы с индексами после 21 будут нулевыми. Все координаты также были нормализованы от 0 до 1 относительно координат изображения.

```
{
  "num_hands": 2,
  "landmarks": [
    [
      1,
      0.9376944347895629,
      -1.6562921700824518e-7
    ],
    [
      0.8552957420378341,
      0.8466151352315685,
      -0.0012534725246950984
    ],
    [
      0.6261349278546047,
      0.7566050155486314,
      -0.0054916092194616795
    ],
    [
      0.4559098215631299,
      0.68231147092206,
      -0.009846813045442104
    ],
    [
      0.3920729425002743,
      0.5881747352426417,
      -0.013850336894392967
    ],
    ...
    [
      -0.29899818204940826,
      0.3388255268253013,
      -0.06701805721968412
    ]
  ]
}
```


Рис. 1: Пример экземпляра набора данных с точками

Второй набор — изображения, содержащие кисти рук, для каждого жеста размером 64 × 64.

Для цифр русского жестового языка теперь существует два корпуса данных: 10 классов по 300 документов в формате JSON для каждого жеста и 10 классов по 300 цветных фотографий для каждого жеста только обнаруженных рук разрешением 64 × 64. А для букв русского жестового языка - 25 классов по 8 документов в формате JSON для каждого жеста, которые содержат информацию о 50 кадрах, в течение которых, показывался жест, то есть массивы размером 50 × 43 ориентиров рук. И 25 классов по 8 папок для каждого человека с 50 цветными фотографиями для каждого жеста только обнаруженных рук разрешением 64 × 64. Если жест показывался меньше чем за 50 кадров, то в случае точек недостающие данные дополнялись методом интерполяции, а в случае изображений дополнялись копией предыдущего.

Алгоритм распознавания жестов

Для работы с видео используется библиотека с открытым кодом алгоритмов компьютерного зрения, обработки изображений и численных алгоритмов общего назначения — OpenCV. Она предоставляет очень простой интерфейс для захвата видео с камеры, в данном случае, встроенной веб–камеры на ноутбуке.

Для распознавания рук на видео используется решение «MediaPipe Hand Landmarker», которое локализует на кадре ключевые точки правой и/или левой руки и может визуализировать их (пример на рис. 2).

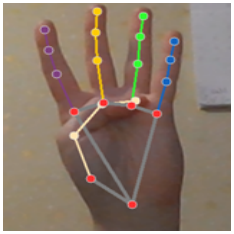

Рис. 2: Пример работы «MediaPipe Hand Landmarker»

Собранные корпуса данных для русского жестового языка будут использоваться для обучения моделей нейронных сетей, главной задачей которых является научиться правильно классифицировать цифры и буквы языка жеста. Были выбраны четыре модели: для распознавания жеста по ключевым точкам руки – многослойный персептрон (MLP), одномерная свёрточная (1D CNN) и сеть долгой краткосрочной памяти (LSTM), а также по фото – двумерная свёрточная сеть (2D CNN).

Алгоритм распознавания языка жестов включает в себя следующие шаги:

1. Подключение решения «MediaPipe Hand Landmarker» в режиме одиночного обнаружения (каждый кадр видео рассматриваем отдельно) и минимальным значением достоверности – 0.6 (то есть при оценке >= 0.6 обнаружение рук считается успешным);
2. Захват живого потока с веб–камеры;
3. Первичная обработка каждого кадра видео для распознавания руки:
    - Прочитать кадр;
    - Перевернуть кадр вокруг оси Y для правильного вывода рук;
    - Преобразовать изображение BGR в RGB и передать его на обработку в конвейер решения «MediaPipe Hand Landmarker», где кадр пройдёт через модель обнаружения ладони, а потом пройдёт через модель ориентира руки. В результате,

получим 21 ориентир для каждой обнаруженной руки на данном видеокадре;

4. Вторичная обработка каждого кадра видео для классификации жеста:
   - Если выбрана модель двумерной свёрточной сети. Кадр обрезается с учётом найденных на прошлом шаге ориентиров обнаруженных рук, размер изображения меняется на 64×64, кадр нормализуется и отправляется на вход обученной 2D CNN, где происходит предсказание с использованием обученной нейронной сети;
   - Если выбрана одна из моделей, обученная на ключевых точках рук (MLP/1D CNN/LSTM). Представить кадр как массив из 43 точек, где 21 ориентир для правой руки, 21 ориентир для левой руки, вычисленные на прошлом шаге, и посчитанная точка соотношения положений двух рук относительно друг друга с помощью алгоритма Евклида. Если на кадре была обнаружена одна рука, то элементы с индексами после 21 будут нулевыми. Далее координаты нормализуются от 0 до 1 и массив данных отправляется на вход обученной сети, где она по ключевым точкам рук пытается предсказать какой жест изображён на кадре;
5. На кадре визуализируются ориентиры обнаруженных рук и отображается символ, соответствующий жесту (пример на рис. 3);

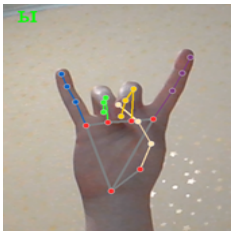

Рис. 3: Пример распознавания руки и классификации жеста

6. Шаги 3-5 повторяются до тех пор, пока пользователь не прекратит использовать режим распознавания;
7. Отпустить захват видеопотока и освободить ресурсы.

На основе данного алгоритма было создано приложение, распознающее жестовый язык. Входными данными для приложения яв-

ляется видеопоток, захваченный веб-камерой. В качестве выходных данных пользователь может получить видеоизображение с расшифровкой в виде текстового предсказания распознанного жеста. Приложение реализовано на языке программирования Python с использованием библиотек OpenCV, MediPipe, TensorFlow и разработанного алгоритма распознавания жестов на основе обученной ранее нейронной сети.

Экспериментальные исследования

Во время экспериментов проводились исследования по влиянию архитектуры и параметров обучения на качество работы нейронных сетей. Были протестированы различные комбинации архитектур и параметров, включая количество слоев, количество нейронов в каждом слое, функции активации, скорость обучения и размер пакета данных.

Нейронные сети обучались на корпусах данных, описанных в разделах «Сбор корпуса данных» и «Обработка корпусов данных». Каждый корпус данных был разделён на три выборки: обучающую (60%), валидационную (20%) и тестовую (20%). Обучение проводилось на обучающей выборке, а результаты оценивались на валидационной и тестовой выборках.

Эксперименты проводились в три этапа. Первый этап включал в себя создание таких моделей сетей, которые давали наилучший результат классификации цифр русского жестового языка, второй этап - букв, а третий одновремнно цифр и букв.

В итоге, было разработанно 12 оптимальных архитектур нейронных сетей по 4 на каждом этапе экспериментов. На рисунке 4 представлен пример архитектуры 1D CNN для классификации цифр.

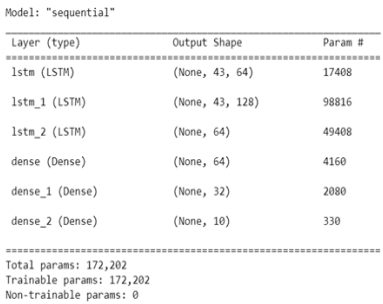

Рис. 4: Архитектура 1D CNN для классификации цифр

Поскольку корпус данных для русского жестового языка мал, то было решено также провести эксперименты с применением процесса аугментации и переобучить каждую модель на новых данных. Использование такого инструмента может улучшить устойчивость моделей к изменениям входных данных и улучшить их общую производительность.

Для решения задачи классификации жестов по ключевым точкам рук были взяты методы девиации по координатам x и y в пределах 5% и сдвиг точек на угол в 2, 5 и 10 градусов по формулам 1 – 2, где random — случайное число от 0 до 1, а α — угол сдвига в градусах.

$$\begin{aligned} x &= x + 0.05 * random - \frac{0.05}{2} \\ y &= y + 0.05 * random - \frac{0.05}{2} \end{aligned} \tag{1}$$

$$\begin{aligned} x &= x * \cos\alpha - y * \sin\alpha \\ y &= x * \sin\alpha + y * \cos\alpha \end{aligned} \tag{2}$$

А для классификации по изображениям — случайным образом изменения яркости и контраста, и переворачивание по горизонтали (слева направо) и вертикали (сверху вниз).

Таким образом, после применения аугментации данных каждому экземпляру данных соответствует 5 образцов, где один оригинальный, а четыре аугментированных. Следовательно, наборы данных для цифр и букв русского жестового языка увеличились в 4 раза.

Проверка качества обучения каждой нейронной сети проводилась на соответствующей тестовой выборке на каждом этапе экспериментов. В качестве метрик были выбраны метрики точности, полноты и F–меры, так как они часто приминяются в задачах классификации. Данная оценка модели показывает, насколько хорошо сеть обобщает данные и способна предсказывать результаты для новых данных. Кроме того, были произведены испытания на реальных видеоданных. Такое опробование является критически важным для обеспечения высокой производительности каждой модели в реальных условиях эксплуатации.

Ниже на таблицах 1 – 6 представлены вычисленные метрики качества до аугментации и после для каждой модели нейронной сети на каждом этапе исследования:

1. Классификация жестов цифр:

Модель	**До аугментации**			**После аугментации**		
	Точность	**Полнота**	**F-мера**	**Точность**	**Полнота**	**F-мера**
MLP	0.94	0.94	0.94	0.94	0.94	0.94
1D CNN	0.93	0.92	0.92	0.94	0.93	0.93
LSTM	0.61	0.60	0.59	0.83	0.80	0.79
2D CNN	0.99	0.99	0.99	0.99	0.99	0.99

Таблица 1: Оценка качества модели на тестовых данных цифр

Модель	**До аугментации**			**После аугментации**		
	Точность	**Полнота**	**F-мера**	**Точность**	**Полнота**	**F-мера**
MLP	0.81	0.79	0.78	0.84	0.86	0.84
1D CNN	0.78	0.78	0.77	0.80	0.82	0.80
LSTM	0.46	0.50	0.43	0.77	0.72	0.70
2D CNN	0.60	0.60	0.59	0.65	0.65	0.63

Таблица 2: Оценка классификации русских жестов цифр на видео

2. Классификация жестов букв:

Модель	**До аугментации**			**После аугментации**		
	Точность	**Полнота**	**F-мера**	**Точность**	**Полнота**	**F-мера**
MLP	0.99	0.99	0.99	1.00	0.99	1.00
1D CNN	0.96	0.95	0.94	0.98	0.97	0.97
LSTM	0.89	0.88	0.88	0.94	0.94	0.94
2D CNN	1.00	1.00	1.00	1.00	1.00	1.00

Таблица 3: Оценка качества моделей на тестовых данных букв

Модель	**До аугментации**			**После аугментации**		
	Точность	**Полнота**	**F-мера**	**Точность**	**Полнота**	**F-мера**
MLP	0.12	0.24	0.11	0.41	0.50	0.42
1D CNN	0.43	0.52	0.44	0.52	0.61	0.53
LSTM	0.00	0.07	0.01	0.09	0.19	0.09
2D CNN	0.04	0.09	0.05	0.17	0.24	0.17

Таблица 4: Оценка классификации русских жестов букв на видео

3. Классификация жестов цифр и букв:

Модель	**До аугментации**			**После аугментации**		
	Точность	**Полнота**	**F-мера**	**Точность**	**Полнота**	**F-мера**
MLP	0.96	0.96	0.96	0.97	0.97	0.97
1D CNN	0.94	0.92	0.92	0.95	0.94	0.94
LSTM	0.85	0.85	0.85	0.93	0.92	0.92
2D CNN	0.99	0.99	0.99	0.99	0.99	0.99

Таблица 5: Оценка качества моделей на тестовых данных цифр и букв

Модель	**До аугментации**			**После аугментации**		
	Точность	**Полнота**	**F-мера**	**Точность**	**Полнота**	**F-мера**
MLP	0.27	0.26	0.25	0.29	0.36	0.31
1D CNN	0.24	0.30	0.23	0.22	0.28	0.23
LSTM	0.12	0.14	0.11	0.15	0.14	0.13
2D CNN	0.17	0.19	0.16	0.26	0.20	0.17

Таблица 6: Оценка классификации русских жестов цифр и букв на видео

Заключение

В статье были рассмотрены методы и подходы к решению задачи распознавания жестов рук на видео, включая методы машинного обучения и методы обработки изображений.

Был самостоятельно создан корпус данных русского жестового языка и разработан алгоритм с использованием четырёх созданных моделей нейронных сетей: MLP, 1D CNN, LSTM и 2D CNN, который способен определять жесты некоторых цифр и букв русского жестового языка на видео. Была применена аугментация данных и проведена оценка качества алгоритма на основе анализа его производительности и посчитанных метрик.

Дальнейшее развитие алгоритма может включать улучшение его точности распознавания, создание более подходящих моделей классификации жестов, а также расширение его функциональности для работы с другими типами жестов и объектов. Данный алгорит может быть применён в различных областях, связанных с компьютерным зрением, управлением компьютером и социальной интеграцией.

Список литературы

1. *Abdallah M.*, *Gerges Samaan A. W.*, *Fazliddin Makhmudov Y.-I. C.* Light-Weight Deep Learning Techniques with Advanced Processing for Real-Time Hand Gesture Recognition // Sensors. 2023. Vol. 23, no. 2.
2. *Miah A. S. M.*, *Al Mehedi Hasan J. S.*, *Yuichi Okuyama Y. T.* Multistage Spatial Attention-Based Neural Network for Hand Gesture Recognition // Computers. 2023. Vol. 12, no. 13.
3. *MediPipe*. Документация MediPipe. URL: https://developers.google.com / mediapipe / solutions / vision / hand _ landmarker (дата обр. 09.05.2023).