УЧЕНЫЕ ЗАПИСКИ КАЗАНСКОГО УНИВЕРСИТЕТА
Том 155, кн. 3 Физико-математические науки *2013*

УДК 004.9312

НЕПРЕРЫВНОЕ РАСПОЗНАВАНИЕ БАЗОВЫХ ЖЕСТОВ В РЕАЛЬНОМ ВРЕМЕНИ С ПРИМЕНЕНИЕМ СКРЫТЫХ МАРКОВСКИХ МОДЕЛЕЙ

Э.М. Красильников

Аннотация

Предложена система распознавания 14 базовых жестов русского языка глухих в реальном времени. Основной особенностью нашего подхода является нахождение начала и конца жестов в непрерывном движении руки и их классификация. Сегментация производится по таким характеристикам, как скорость и изменение направления движения рук. Для обучения и классификации применен метод машинного обучения с учителем (скрытые марковские модели). Помимо траектории, система определяет места исполнения движения относительно частей тела. Для тестирования метода создана программная система распознавания жестов с помощью сенсора глубины. Эксперименты показали успешные результаты в 95% случаев.

Ключевые слова: распознавание жестов, скрытые марковские модели.

Введение

Жесты – это один из естественных способов передачи информации между людьми и взаимодействия с окружающей средой. Распознавание и интерпретация движений человека является объектом исследования широкого ряда дисциплин, от психологии до математики. С развитием вычислительных устройств и технологий связи значительный интерес проявляется к более эффективным и быстрым видам человеко-компьютерного взаимодействия. Помимо речевого управления, жесты являются богатым инструментом для взаимодействия с машиной, особенно для людей с ограниченным слухом. Жесты также могут применяться в качестве интерфейсов в виртуальных средах, удаленном управлении робототехникой.

Задача автоматического выделения жестов в непрерывном потоке является сложной из-за двух аспектов: неопределенность сегментации и пространственно-временная неустойчивость. Проблема сегментации состоит в определении начала и конца жеста в непрерывной траектории руки, так как даже опытные специалисты не могут с уверенностью указать точные границы значащих движений. Вторая проблема заключается в том, что один и тот же жест может отличаться по форме и длительности, даже для одного исполнителя.

Скрытые марковские модели (СММ) хорошо зарекомендовали себя при решении задач пространственно-временной неустойчивости в распознавании речи. Как следствие, существует ряд успешных применений СММ и для распознавания жестов. Одной из ранних и значимых работ в этом направлении является статья [1]. Эффективность приложений СММ зависит от набора признаков и обучающих примеров. Например, Ш. Айкелер с соавторами [2] использовали моменты при описании признаков двухмерных жестов. Непрерывное распознавание они реализовали путем отдельного обучения некоторым переходным жестам. В [3] предложен способ сегментации жестов с помощью динамического программирования. С. Ван с соавторами [4] аппроксимируют полученные входные данные кубическим сплайном

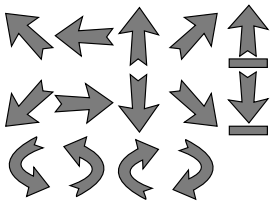

Рис. 1. 14 распознаваемых жестов

и вычисляют инвариантные моменты для определения признаков движения. В статье [5] система отдельно обучалась переходным жестам.

Целью настоящей работы является разработка метода и программная реализация определения начала – конца жестов и классификации отдельных движений. В список задач входит подготовка набора обучающих примеров для машинного обучения и подбор признаков для различения жестов.

В описанных выше работах в качестве базовых единиц рассматриваются жесты, означающие слова и (или) словосочетания. Часто распознавание таких сложных движений вызывает ряд ошибок и неточностей. В нашем подходе мы разбиваем все жесты на базовые единицы. Относительная простота распознавания таких структур позволяет описать их геометрическими правилами. Таким образом, мы имеем возможность автоматически генерировать необходимое количество примеров для обучения модели в короткие сроки.

1. Реализация

Любой жест языка глухих можно описать 4 признаками, такими как характер и направление движения (траектория), местоположение рук относительно частей тела, конфигурация пальцев, мимика. В силу технических ограничений мы рассматриваем первые 2 признака, то есть траекторию и положение рук. В качестве движений для обучения и тестирования нашей системы мы выбрали 14 базовых жестов, входящих в словарь русского языка жестов по классификации Димскис [6]. Данные жесты включают в себя 8 направлений во фронтальной плоскости, 2 движения «от себя» и «к себе», 4 движения полукругом по и против часовой стрелки (см. рис. 1).

Такой набор не является достаточным для полноценного общения на языке жестов, но входит в состав большинства употребительных.

Для нахождения и отслеживания руки в трехмерном пространстве мы используем сенсорное устройство Microsoft Kinect [7]. Каждую секунду мы регистрируем до 30 наборов координат положений рук человека, исполняющего жест. Используя эти данные, строим траекторию движения руки и отображаем на экране в виде отрезков (см. рис. 2).

Предварительная работа перед обучением и распознаванием делится на следующие этапы:

Этап 1. Выделение жестов из непрерывного потока;

Этап 2. Подготовка входных параметров;

Этап 3. Извлечение признаков.

Рассмотрим эти этапы последовательно. После этого разберем алгоритм обучения и классификации с помощью СММ.

Этап 1. Выделение жестов из непрерывного потока. В [8] была показана высокая корреляция между изменением скорости и временем конца жестов. В качестве основной характеристики для сегментации мы также применяем среднюю скорость за последние 100 мс и сравниваем ее с заданным пороговым значением,

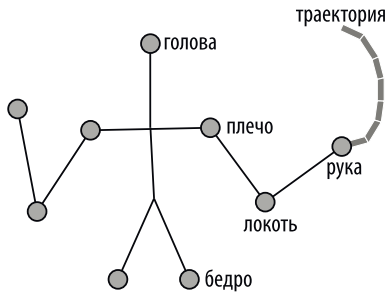

Рис. 2. Верхняя часть туловища и траектория

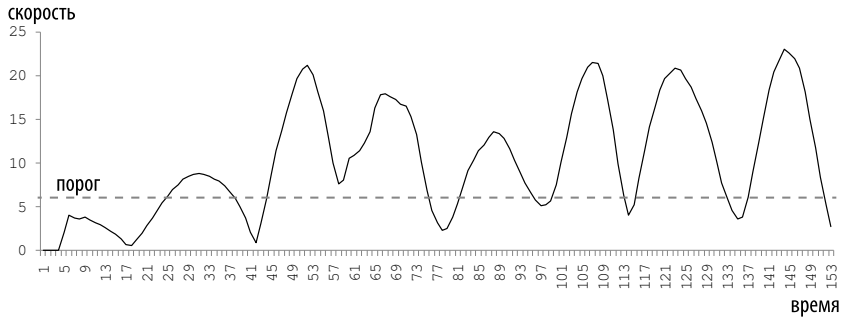

Рис. 3. Изменение скорости на тестовом примере

выявленным опытным путем. Локальные минимумы, расположенные ниже порога, будем считать границей жеста (рис. 3).

Когда движение совершается аномально быстро, невозможно отследить конец жеста по изменению скорости. Поэтому, в отличие от других работ, мы добавляем еще один критерий: изменение направления движения. Для него также задается пороговое значение, выявленное экспериментально.

Этап 2. Подготовка входных параметров (векторизация). Перед тем как приступить к обучению СММ, необходимо обработать входные данные (см. Алгоритм 1), избавиться от избыточности, усилить некоторые признаки. В нашем случае, мы нормализуем рассматриваемые участки траектории до куба $1000 \times \times 1000 \times 1000$. Другими словами, максимальное значение по каждой из координат не будет превышать 1000 условных единиц.

Алгоритм 1. Сдвиг и Нормализация

1: Находим максимумы и минимумы координат внутри рассматриваемого отрезка.

2: Сдвигаем траекторию в начало координат, вычитая найденные минимумы из соответствующих координат.

3: Находим коэффициент масштабирования

$$\frac{1000}{\max(x_{\max} - x_{\min}, y_{\max} - y_{\min}, z_{\max} - z_{\min})}.$$

4: Умножаем все текущие координаты на коэффициент масштабирования.

После этого уменьшаем количество точек в траектории и равномерно распределяем по всей длине.

Этап 3. Извлечение признаков. Выбор подходящих признаков критически важен для успешного распознавания с помощью СММ. После ряда экспериментов мы выбрали по 2 признака для обучения: направление текущего отрезка относительно оси координат и угол поворота между двумя соседними отрезками траектории (см. Алгоритм 2). Эту пару мы рассматриваем для проекций на плоскость OXY и OXZ соответственно.

Алгоритм 2. Извлечение признаков

1: Берем три последовательных точки траектории

$$d_1(x, y, z),\ d_2(x, y, z),\ d_3(x, y, z).$$

2: Вычисляем направление отрезка между двумя соседними точками траектории на плоскости OXY (первый признак)

$$\operatorname{sign}(d_{2y} - d_{1y}) \arccos\left(\frac{d_{2x} - d_{1x}}{\sqrt{(d_{2x} - d_{1x})^2 + (d_{2y} - d_{1y})^2}}\right).$$

3: Получаем два трехмерных вектора, соответствующих двум соседним отрезкам траектории

$$\mathbf{A}(x, y, z) = (d_{2x} - d_{1x}, d_{2y} - d_{1y}, d_{2z} - d_{1z})$$

$$\mathbf{B}(x, y, z) = (d_{3x} - d_{2x}, d_{3y} - d_{2y}, d_{3z} - d_{2z}).$$

4: Вычисляем угол поворота между a и b на плоскости OXY (второй признак)

$$\operatorname{arctg}\left(\frac{\mathbf{A}_x \cdot \mathbf{B}_y - \mathbf{B}_x \cdot \mathbf{A}_y}{\mathbf{A}_x \cdot \mathbf{B}_x + \mathbf{A}_y \cdot \mathbf{B}_y}\right).$$

5: Находим аналог первого признака для плоскости OXZ

$$\operatorname{sign}(d_{2z} - d_{1z}) \arccos\left(\frac{d_{2x} - d_{1x}}{\sqrt{(d_{2x} - d_{1x})^2 + (d_{2z} - d_{1z})^2}}\right).$$

6: Находим аналог второго признака для плоскости OXZ

$$\operatorname{arctg}\left(\frac{\mathbf{A}_x \cdot \mathbf{B}_z - \mathbf{B}_x \cdot \mathbf{A}_z}{\mathbf{A}_x \cdot \mathbf{B}_x + \mathbf{A}_z \cdot \mathbf{B}_z}\right).$$

Таким образом, мы получаем массив данных, где каждая точка траектории описывается этими 4 признаками. Первую и последнюю точку мы не принимаем во внимание.

Обучение скрытых марковских моделей. При инициализации СММ в качестве топологии мы используем последовательную модель с тремя состояниями. Обучение классификатора начинается с сортировки примеров по отдельным классам (см. Алгоритм 3). Каждый пример-жест, представленный в виде последовательности точек, векторизуется до куба размерностью 1000. Далее мы извлекаем набор признаков для каждой точки траектории. После этого данные с меткой класса подаются на вход СММ. Таким образом, мы информируем систему, что подобные

признаки характерны для текущего класса жестов. Повторяем действия для всех примеров.

Алгоритм 3. Обучение классификатора

1: Даны Q обучающих примера $< (O_1, y_1), \ldots, (O_Q, y_Q) >$, где O_i – это наблюдаемая последовательность координат руки с меткой y_i, $y_i \in \in \{1, \ldots, M\}$, $i = 1, \ldots, Q$; M – число классов (в нашем случае жестов).
2: Делим эти Q примеров на M групп так, чтобы каждая группа содержала элементы с одной меткой.
3: Подготавливаем данные из O_i: нормализуем и равномерно распределяем точки.
4: Из каждой подготовленной O_i извлекаем последовательность признаков F_i (по четыре признака на точку).
5: С помощью алгоритма Baum-Welch [1] обучаем СММ, подавая на вход признаки F_i и соответствующую метку y_i.

Классификация на основе СММ. Существуют 3 основные задачи, которые необходимо решить для использования СММ: оценивание, декодирование и обучение.

Оценивание: Даны наблюдаемая последовательность и модель. Необходимо вычислить вероятность того, что данная наблюдаемая последовательность построена именно для данной модели.

Декодирование: Даны наблюдаемая последовательность и модель. Необходимо подобрать последовательность состояний системы, которая лучше всего соответствует наблюдаемой последовательности, то есть «объясняет» наблюдаемую последовательность.

Обучение: Необходимо подобрать параметры модели таким образом, чтобы она как можно лучше описывала реальную наблюдаемую последовательность

Эти 3 задачи решаются с использованием известных алгоритмов Forward, Viterbi и Baum-Welch соответственно. Подробнее с СММ читатель может ознакомиться в работе [1] и др.

Подготовка жеста к классификации с помощью СММ происходит так же, как при обучении, а именно: производится векторизация данных и выделение набора признаков. Далее мы передаем признаки на вход нашей обученной модели для определения класса жеста. На выходе выдается набор оценок схожести с каждым видом жеста. Выбирая максимальное значение, мы сопоставляем данные с наименованием жеста (см. Алгоритм 4).

Алгоритм 4. Классификация

1: Дана наблюдаемая последовательность координат руки $O = \\ = o_1 o_2 \ldots o_T$.
2: Подготавливаем данные: нормализуем и распределяем точки.
3: Из данной последовательности извлекаем последовательность векторов признаков F.
4: С помощью алгоритма Viterbi [1], подавая на вход признаки F, вычисляем вероятность принадлежности к каждому классу M: $P(O|\lambda)$, где λ – это параметры модели.
5: Выбираем класс $M_{\max}$, набравший максимальную вероятность $P_{\max}(O|\lambda)$.

Табл. 1

Результаты тестов

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	94	0	0	0	0	0	0	0	0	6	0	0	0	0
2	0	100	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	100	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	100	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	97	0	0	0	0	0	0	0	3	0
6	0	0	0	0	0	100	0	0	0	0	0	0	0	0
7	0	0	0	3	0	0	93	0	3	0	0	0	0	0
8	0	0	0	0	0	0	0	100	0	0	0	0	0	0
9	3	0	0	0	7	0	0	0	90	0	0	0	0	0
10	3	0	0	0	0	0	0	7	0	90	0	0	0	0
11	0	0	0	7	0	0	0	3	0	0	90	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	100	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	90	10
14	0	0	0	0	3	0	0	0	0	3	0	0	3	91

1) полукруг вправо-вверх, 2) полукруг влево-вниз, 3) полукруг влево-вверх, 4) полукруг вправо-вниз, 5) вверх, 6) вниз, 7) вправо, 8) влево, 9) вправо-вверх, 10) влево-вверх, 11) вниз-вправо, 12) вниз-влево, 13) к себе, 14) от себя.

2. Эксперименты

При обучении системы применяется математическая модель построения жестов. Таким образом, мы имеем возможность генерировать любое количество движений, что экономит ресурсы на подготовку обучающих примеров человеком вручную. Для улучшения результатов распознавания генератор варьирует местоположение, размерность и форму траектории; добавляет шумы, сдвигающие точки на небольшую случайную величину в сторону; случайным образом удаляет часть точек внутри траектории. В нашем случае было создано по 2000 жестов для каждого класса, что в сумме составляет 28000 различных движений. Это число является оптимальным при данных условиях, и дальнейшее увеличение количества обучающих примеров не дает заметного прироста в производительности распознавания. Но так как система должна распознавать реальные естественные движения, при тестировании и оценивании результатов обучения мы используем примеры, исполненные человеком вручную.

В табл. 1 приведены результаты тестов программного продукта при распознавании жестов в реальном времени. Строкам в табл. 1 соответствуют классы жестов, исполненные человеком; столбцам – классы жестов, распознанные компьютером. Для каждого класса (по строкам) указано, как наш алгоритм распознал данный жест и в скольких процентах случаев. Так, например, класс жестов 1 (полукруг вправо-вверх) в 94% опытов был распознан верно, а в 6% ошибочно принят за класс 10 (влево-вверх). В среднем же предложенный метод дает правильный результат с вероятностью 95%.

Заключение

Для тестирования описанного подхода мы реализовали программную систему с пользовательским интерфейсом на языке C#. Для отслеживания координат скелета исполнителя мы применяем устройство Microsoft Kinect [7] и библиотеку OpenNI [9]. В общей сложности обучение системы производилось на более 700 жестах, исполненных человеком, и 28000 жестах, сгенерированных компьютером. Для обучения и классификации CMM мы настроили под нашу задачу систему

с открытым кодом Accord.net [10], содержащую вспомогательные алгоритмы для машинного обучения.

Эксперименты показали, что описанный метод дает хорошие результаты в распознавании предложенных 14 жестов, а также комплексных жестов, составленных из этого набора. Данный словарь можно легко расширить и использовать на практике.

Summary

E.M. Krasilnikov. Continuous Real-Time Recognition of Basic Gestures Using Hidden Markov Models.

A system for real-time recognition of 14 basic gestures of the Russian sign language was suggested. The main feature of our approach is the ability to find the beginning and the end of the gestures in a continuous hand movement and to classify them. Segmentation was produced using such characteristics as the velocity and the direction change of hand movement. Hidden Markov models (supervised machine learning approach) were used for training and classification. Besides trajectory, the location of the movement was detected with respect to parts of the body. A software system of gesture recognition with depth sensor was developed for testing our method. The experiments showed successful results in 95% of cases.

Keywords: gesture recognition, hidden Markov models.

Литература

1. *Yang J., Xu Y.* Hidden Markov Model for Gesture Recognition: Technical Report CMU-RI-TR-94-10. – Pittsburgh, PA: Carnegie Mellon Univ., Robotics Institute, 1994. – 27 p.
2. *Eickeler S., Kosmala A., Rigoll G.* Hidden Markov model based continuous online gesture recognition // Proc. Fourteenth Int. Conf. on Pattern Recognition. – 1998.– V. 2. – P. 1206–1208.
3. *Lv F., Nevatia R.* Recognition and segmentation of 3-D human action using HMM and multi-class adaBoost // Proc. 9th Eur. Conf. on Computer Vision. – Berlin; Heidelberg: Springer-Verlag, 2006. – Pt. 4 – P. 359–372.
4. *Wang X., Xia M., Cai H., Gao Y., Cattani C.* Hidden-Markov-models-based dynamic hand gesture recognition // Math. Problems Eng. – 2012. – V. 2012. – P. 986134-1–986134–11.
5. *Nam Y., Wohn K.* Recognition of space-time hand-gestures using hidden Markov model // Proc. ACM Symposium on Virtual Reality Software and Technology. – Hong Kong, 1996. – P. 51–58.
6. *Димскис Л.С.* Изучаем жестовый язык. – М.: Академия, 2002. – 128 с.
7. Microsoft Kinect. – URL: http://www.microsoft.com/en-us/kinectforwindows/.
8. *Gibet S., Marteau P.-F.* Approximation of Curvature and Velocity for Gesture Segmentation and Synthesis // Gesture Workshop 2007. – Berlin; Heidelberg: Springer-Verlag, 2009. – P. 13–23.
9. Open Natural Interaction Framework. – URL: http://www.openni.org/.
10. Accord.NET. – URL: http://code.google.com/p/accord/.

Поступила в редакцию
12.06.13

Красильников Эдуард Мансурович – аспирант кафедры теоретической кибернетики, Казанский (Приволжский) федеральный университет; младший научный сотрудник, Институт информатики Академии наук Республики Татарстан, г. Казань, Россия.

E-mail: *sphinx2412@gmail.com*