

УДК 004.93.11, 004.354

*А. В. Хельвас¹, Н. Г. Беляйкина², А. А. Гилья-Зетин³, Д. Д. Черникова³,
В. М. Шабунин³, Е. О. Япрынтцев⁴*

¹ООО «Лаборатория моделирования систем»

²Национальный исследовательский университет МИСИС

³Московский физико-технический институт (государственный университет)

⁴Московский инженерно-физический институт

Распознавание жестов с помощью нейронной сети и применение этого подхода для создания игровых гаджетов нового поколения

В статье рассмотрена задача распознавания жестов, выполняемых жезлом (палочкой), находящимся в руке человека, на основе анализа потока данных с инерциального модуля, встроенного внутрь палочки. Для анализа использована рекуррентная нейронная сеть с долгой краткосрочной памятью. Приведены результаты эксперимента по распознаванию четырех простейших жестов. Для результатов распознавания даны традиционные метрики, описывающие их качество.

Ключевые слова: распознавание жестов, инерциальный датчик, нейронная сеть, волшебная палочка.

*A. V. Helvas¹, N. G. Belyaikina², A. A. Gilya-Zetinov³, D. D. Chernikova³,
V. M. Shabunin³, Y. O. Yapryntsev⁴*

¹Systems Simulation Laboratory LLC

²The National University of Science and Technology MISiS

³Moscow Institute of Physics and Technology (State University)

⁴Moscow Engineering Physics Institute

Gesture recognition by neural network and its implementation for the new generation of interactive gadgets for children

The article deals with the problem of recognizing gestures performed by a stick, which is in the person's hand on the basis of data stream analysis from the inertial measurement unit module inside the stick. The Long shortterm memory (LSTM) neural network is used for analysis. The results of four simplest gestures recognition experiment are presented. For the results of recognition, the common quality metrics are presented.

Key words: gesture recognition, IMU, LSTM neural network, magic stick.

1. Введение

26 июня 2017 года весь мир отметил 20-летие со дня выхода первой книги о Гарри Поттере. Популярность серии фильмов про Гарри Поттера является уникальным культурологическим явлением, породившим целую индустрию игровых гаджетов, повторяющих те, которыми пользовались герои фильма.

Современные технологии позволяют частично реализовать функционал волшебной палочки. На Amazon.com продается устройство Harry Potter Magic Wand (рис. 1), которое с помощью встроенного инерциального модуля IMU обеспечивает распознавание 13 жестов и может ставить им в соответствие команды, передаваемые внешнему устройству через инфракрасный порт.



Рис. 1. Устройство «Harry Potter Magic Wand»

Таким образом, «волшебная палочка» оказывается интересным элементом «Интернета вещей» в качестве инструмента управления IoT-устройствами для детей и подростков.

С нашей точки зрения, несмотря на наличие у ИК-порта таких достоинств, как простота реализации процесса управления, использованный авторами подход имеет ряд существенных недостатков:

- наличие весьма ограниченного набора внешних устройств, имеющих встроенный ИК-порт управления;
- невозможность гибкой перенастройки с одного объекта управления на другой;
- невозможность управления объектами в виртуальной и дополненной реальности.

Прикладной целью работы является создание прототипа волшебной палочки, лишённого перечисленных недостатков и встроенного в единую облачную инфраструктуру.

В рамках данной статьи описаны:

- конструкция макета, на основе которого проводились исследования;
- алгоритм распознавания жестов с помощью нейронной сети;
- результаты проведенных экспериментов.

2. Описание аппаратной части решения

При разработке дизайна и конструкции решения мы исходили из того, что оно относится к индустрии технических игрушек и предназначено для управления игровыми устройствами, оснащенными механизмами управления в рамках концепции IoT (Internet of Things).

2.1. Описание макета устройства

Разработанный макет устройства, использованный далее в экспериментах, имеет следующие технические характеристики:

- протоколы подключения к сети: Wi-fi IEEE 802.11 и Bluetooth IEEE 802.15.1;
- время непрерывной работы без подзарядки аккумулятора – 4 часа;
- вес устройства 150 г;
- макет устройства имеет форм-фактор цилиндра с размерами: диаметр 31 мм, длина 202 мм (при конструировании финальной версии изделия диаметр будет уменьшен до 18 мм);
- диапазон рабочих температур: $-5... +40$ °C;

- напряжение питания +5 В;
- частота опроса датчиков IMU 300 Гц;
- корпус устройства и общие требования по безопасности должны соответствовать ГОСТ 25779-90;
- допустимая температура окружающей среды от -10°C до $+40^{\circ}\text{C}$;
- влажность не более 90%.

Решение работает следующим образом.

При выполнении жеста с помощью однокристалльного IMU 9255 (9-координатное устройство позиционирования) происходит измерение линейных и угловых ускорений и изменения вектора напряженности магнитного поля Земли. Далее с помощью микроконтроллера STM32F051x происходит предварительная обработка полученных данных и далее передача информации с помощью wi-fi-модуля ESP8266 в приложение с развёрнутой рекуррентной нейронной сетью в облаке. Нейросеть, основываясь на полученных данных, распознает жесты и формирует управляющий сигнал.

Схема решения представлена на рис. 2.

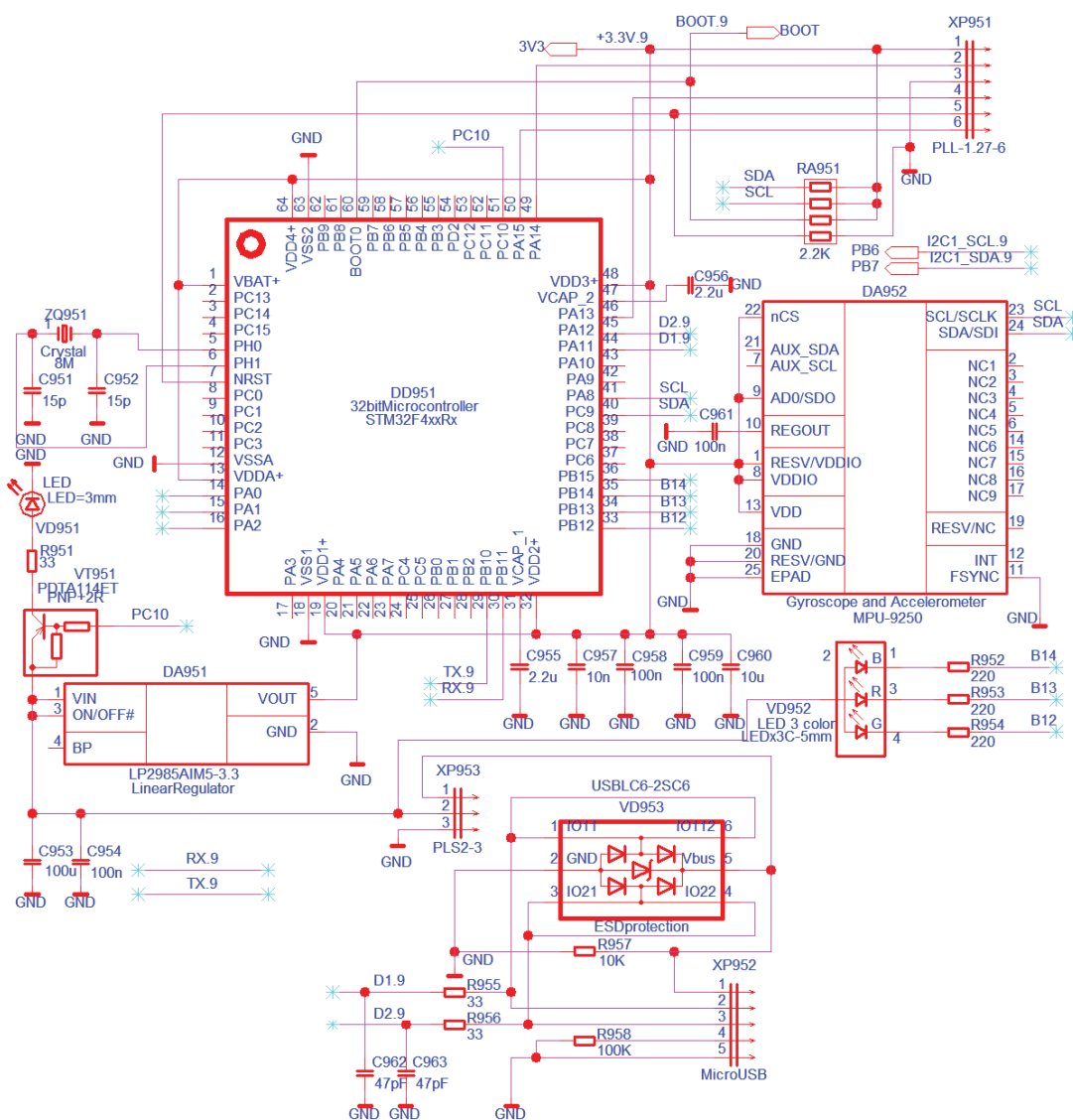


Рис. 2. Схема устройства

В состав устройства входят:

- блок питания, для подачи постоянного напряжения +3.7 В для работы всех модулей устройства;
- контроллер, являющийся аппаратно-программным интерфейсом для всей периферии;
- модуль индикации осуществляет индикацию состояния работы устройства;
- модуль управления необходим для выбора режимов работы;
- 9-координатный IMU, предназначенный для регистрации параметров движения устройства;
- приёмопередатчик выполняет функцию обеспечения сетевого взаимодействия с программным обеспечением на сервере и объектом управления.

В составе макета использован микроконтроллер STM32f4, обладающий избыточными для серийного образца характеристиками, но позволяющий проводить эксперименты по уточнению возможностей разрабатываемого решения при переходе к задачам биометрической идентификации пользователя и диагностики состояния пользователя, которые выходят за рамки данной статьи.

Устройство индикации подключается к регистрам общего назначения и представляет собой RGB-светодиод, обеспечивающий обратную связь с пользователем. В случае если устройство потеряет связь с сетью, загорается синий светодиод. Об успешном распознавании жеста сообщает зелёный светодиод. При возникновении проблем – красный.

Для связи с сетью использован подключенный через UART приёмопередатчик ESP8266, который обеспечивает взаимодействие по протоколу wi-fi.

Заявленную автономность в 4 часа поддерживает встроенный литий-ионный аккумулятор ёмкостью 2800 мАч с выходным напряжением +5 В и регулятор напряжения LP2985AIM5.

Внешний вид разработанного макета устройства приведен на рис. 3.

Прототип, 1 версия устройства для управления игрушками на основе распознавания жестов, представлен на рис. 3.



Рис. 3. Макет устройства для управления на основе распознавания жестов

На рис. 4 изображена система управления игрушкой с wi-fi-интерфейсом.

Рассмотрим принцип соединения с управляемым устройством-игрушкой. Так как устройство «привязано» к домашней сети, достаточно программно реализовать распознавание её MAC-адреса. Посылка команды управления содержит служебную информацию (адрес и т.д.) и код управляющей команды.

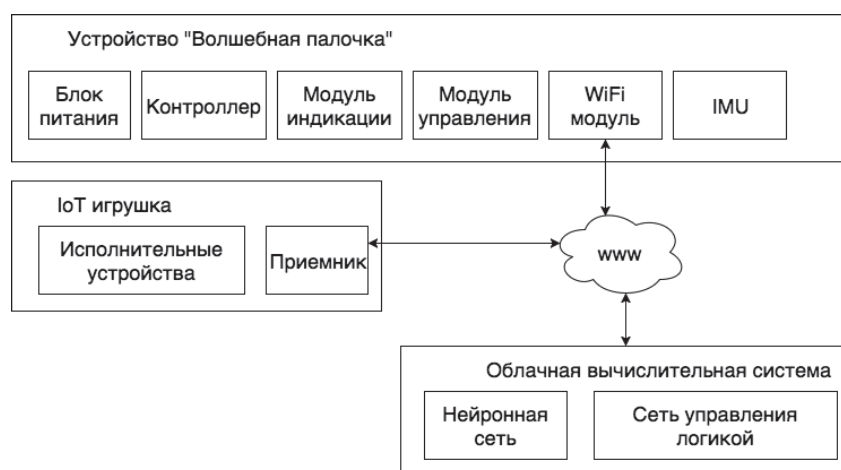


Рис. 4. Система управления IoT-устройствами

2.2. Реализация сенсорной поверхности устройства

Для передачи дополнительных сигналов устройству предложен принципиально новый подход к реализации сенсорного датчика.

На поверхность ручки гаджета 1, являющуюся изолятором, наносится ряд металлизированных площадок с электрическими контактами 4 (см. рис. 5).

Для формирования контактов может использоваться, например, технология травления проводящего слоя 2.

Далее на поверхность наносится слой пластика, в котором находится некоторое количество одностенных углеродных нанотрубок. Благодаря нанотрубкам слой 3 становится проводящим.

За счет влияния прикосновений пальцем 5 на сопротивление между контактами R_{kn} мы получаем возможность по набору R_{kn} решить задачу определения положения пальца относительно поверхности.

Изучение возможностей управления проводимостью слоя 3 путем изменения его толщины d и создания градиента концентрации нанотрубок по оси X выходит за рамки данной публикации.

Также представляет интерес возможность обработки слоя 3 механическим способом или лазерным лучом с целью управления проводимостью слоя 3.

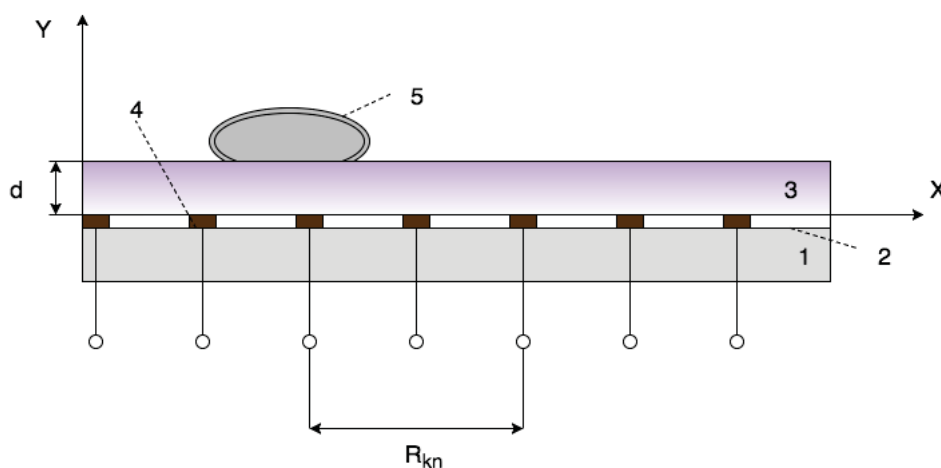


Рис. 5. Предлагаемое решение для сенсорной поверхности устройства

3. Алгоритмы распознавания жестов

3.1. Описание решения

Для распознавания жестов предложено применить подход, основанный на использовании рекуррентной нейронной сети с долгой краткосрочной памятью (Long short-term memory, LSTM). Данный вид сетей ранее успешно применялся при решении схожей задачи распознавания движений руки по показаниям акселерометра «умных» часов [2], а также задачи определения вида деятельности человека по нескольким датчикам (HAR) [3], [4].

3.2. Исходные данные

Обучающая выборка представляет собой совокупность потоков данных, каждый из которых содержит серию жестов, совершенных одним пользователем.

Каждый элемент потока содержит в себе:

- 1) прошедшее время t_i с момента начала записи;
- 2) измеренный вектор ускорения \vec{A}_i ;
- 3) измеренный вектор угловой скорости $\vec{\omega}_i$;
- 4) измеренный вектор напряженности магнитного поля \vec{M}_i ;
- 5) строковое поле Mv_i , равное коду жеста в момент начала его исполнения и «none» – в остальные моменты времени (это поле необходимо для обучения классифицирующего алгоритма и присутствует только в потоках данных из обучающей выборки).

Для обучения и классификации непрерывный поток данных разбивается на перекрывающиеся участки фиксированной длины t_w , с постоянным расстоянием между ними Δt . Размер участка должен превышать ожидаемую продолжительность жеста t_{move} .

Использованные в работе значения составили: t_w – 2 секунды, t_{move} – 1.5 секунды, Δt – 200 миллисекунд.

В зависимости от степени перекрытия участка с размеченными с помощью поля Mv_i областями жестов, при обучении классификатора возможны следующие варианты.

- 1) Участок не пересекается ни с одним из размеченных в потоке жестов либо доля его пересечения ниже некоторого предела r . В таком случае участку присваивается класс «none».
- 2) Один из размеченных жестов полностью содержится внутри участка. В этом случае участку присваивается класс, соответствующий классу совершенного жеста.
- 3) Участок пересекается с размеченным жестом M , и доля его пересечения выше предела r . Это – особый случай. С одной стороны, доля пересечения может быть недостаточна для определения жеста, и поэтому участок не может быть однозначно размечен классом M . С другой стороны, определение классификатором этого участка как M ошибкой не является. Исключение этих участков из обучающей выборки может привести к ложным классификациям на границах совершаемых жестов, поэтому таким участкам присваиваются сразу два класса – «none» и M , особым образом учитываемых в алгоритме обучения классификатора.

Полученная совокупность участков в дальнейшем обрабатывается независимо.

3.3. Предварительная обработка

Решение задачи поиска и классификации жеста на участке разбивается на этапы.

- 1) Перевод показаний акселерометра в неподвижную систему координат.
- 2) Переход в систему отсчета, инвариантную относительно направления жеста.
- 3) Преобразование последовательности значений ускорения в последовательность векторов, инвариантную относительно операции растяжения по времени.
- 4) Классификация полученной последовательности векторов и вектора дополнительных признаков при помощи нейронной сети.

Переход в неподвижную систему координат осуществляется методом оценки ориентации устройства по показаниям акселерометра, гироскопа и магнитометра. Этот алгоритм описан, например, в [1].

Используется гипотеза о том, что существует некоторая плоскость, приближающая траекторию жеста достаточно хорошо, и нормаль к этой плоскости и будет являться направлением жеста.

Формально это можно записать следующим образом.

Пусть \vec{A}_i – ускорение устройства в неподвижной системе координат в момент времени t_i , \vec{n}_1 – нормаль, \vec{p}_1 – радиус-вектор некоторой точки плоскости. Тогда параметры этой плоскости можно определить путем минимизации суммы квадратов расстояний от точек в пространстве ускорений до этой плоскости :

$$\min_{n_1, p} \sum_{i=1}^n ((\vec{A}_i - \vec{p}_1) \vec{n}_1)^2. \quad (1)$$

Полученная нормаль этой плоскости n_1 используется в качестве первого базисного вектора для новой системы координат.

Вторым базисным вектором является проекция вектора ускорения свободного падения g на полученную плоскость.

Третий базисный вектор \vec{n}_3 берется равным векторному произведению \vec{n}_1 и \vec{n}_2 .

После перехода в систему координат $\{\vec{n}_1, \vec{n}_2, \vec{n}_3\}$ проводится ряд преобразований:

- 1) к каждой компоненте ускорения $a_{i,j}, j \in [1, 3]$, в новой системе координат применяется 1-мерный гауссов фильтр;
- 2) из последовательности векторов ускорения удаляются все элементы за исключением локальных максимумов и минимумов компонент $a_{i,1}, a_{i,2}$;
- 3) все значения нормируются на максимум модуля соответствующей компоненты;
- 4) в вектора экстремумов добавляются новые компоненты. Компонента $a_{i,4}$ равна индексу компоненты, экстремум, который наблюдается в данном векторе – в диапазоне $[0, 1]$. Компонента $a_{i,5}$ равна 1 в случае максимума, 0 – в случае минимума.

Полученная последовательность 5-компонентных векторов имеет, как правило, небольшой размер и пригодна в качестве входных данных для классификатора на основе рекуррентной нейронной сети.

Помимо последовательности $a_{i,j}$, на вход сети подается также вектор дополнительных признаков b_i :

- 1) b_1 – z -компонента (сонаправленная с вектором гравитационного поля Земли) вектора n_1 ;
- 2) b_2 – z -компонента вектора n_2 ;
- 3) b_3 – максимум модуля a_1 (до нормировки), деленный на g ;
- 4) b_4 – максимум модуля a_2 (до нормировки), деленный на g .

3.4. Структура нейронной сети

Предлагаемая архитектура сети, размеры слоев и функции активации изображены на рис. 6. Размер последнего слоя совпадает с числом распознаваемых жестов.

Результатом работы нейронной сети являются вероятности p_i , $i \in [0, N]$, где N – число жестов, p_0 – вероятность того, что на данном участке жест совершен не был.

В качестве алгоритма оптимизации весов сети использовалась разновидность стохастического градиентного спуска – алгоритм ADAM [5]. Сеть обучалась на 70% от собранных данных, остальные 30% использовались для валидации.

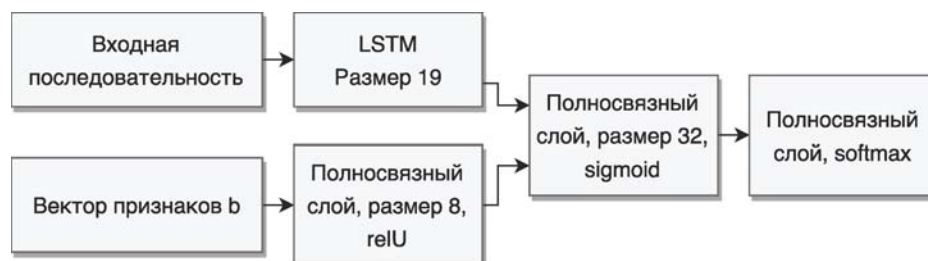


Рис. 6. Структура нейронной сети

4. Обсуждение экспериментальных результатов

4.1. Постановка эксперимента

Для построения обучающей выборки был использован следующий сценарий.

Участник обучающей группы берет в палочку с установленным программным обеспечением и включает запись. Далее повторным нажатием кнопки отмечает начало жеста и совершает жест.

Аналогичным образом участник совершает серию движений одного типа. Далее двукратным нажатием кнопки выключает запись. Аналогично участник проводит запись серии жестов для движений других типов.

Был проведен эксперимент с десятью участниками для четырех простейших видов движений:

- влево–вправо (жест G_1);
- вверх–вниз (жест G_2);
- круговое движение по часовой стрелке (жест G_3);
- круговое движение против часовой стрелки (жест G_4).

Также результатом может быть решение «ни один из перечисленных», которому поставим в соответствие обозначение G_0 .

Для обучения нейросети была использована часть массива данных, включающая в себя жесты, полученные от первых семи участников эксперимента. Жесты оставшихся трех участников были использованы для валидации полученных результатов.

4.2. Оценка эффективности алгоритма

В простейшем случае метрикой, служащей для определения эффективности алгоритма, может быть величина A (Accuracy):

$$A = \frac{P}{N},$$

где P – количество жестов, по которым классификатор принял правильное решение, а N – размер обучающей выборки.

Однако эта метрика не дает исчерпывающего представления об эффективности алгоритма в случае, когда доля жестов сильно смещена в сторону какого-нибудь класса.

Для более эффективной оценки воспользуемся таким понятием, как матрица неточностей.

Матрица неточностей – матрица размера $N \times N$, где N – количество классов (в нашем случае $N = 4$). Столбцы этой матрицы резервируются за экспертными решениями, а строки – за решениями классификатора. Когда мы классифицируем жест из тестовой выборки, мы инкрементируем число, стоящее на пересечении строки класса, который вернул классификатор, и столбца класса, к которому действительно относится жест.

Для определения эффективности алгоритма используем также понятия полноты (precision) и точности класса (recall), а также F-меры алгоритма.

Точность системы в пределах класса – доля исходов, действительно принадлежащих данному классу относительно всех исходов, которые система отнесла к этому классу.

Полнота системы – доля найденных классификатором исходов, принадлежащих классу относительно всех исходов этого класса в тестовой выборке.

Точность класса равняется отношению соответствующего диагонального элемента матрицы неточностей и суммы всей строки класса.

Полнота класса равняется отношению диагонального элемента матрицы и суммы всего столбца класса.

Результирующая точность классификатора рассчитывается как арифметическое среднее его точности по всем классам. Аналогичным образом рассчитывается полнота.

$$Precision_c = \frac{A_{c,c}}{\sum_{i=1}^n A_{c,i}};$$

$$Recall_c = \frac{A_{c,c}}{\sum_{i=1}^n A_{i,c}}.$$

F-мера представляет собой гармоническое среднее между точностью и полнотой нашего алгоритма:

$$F = \frac{2 * Precision * Recall}{Precision + Recall}.$$

4.3. Особенности обработки полученных результатов

Назовем действительным тот тип жеста, который совершил участник эксперимента. Если алгоритм в качестве результата выдаст не тот жест, который совершил участник эксперимента, назовем такой жест ошибочным.

Алгоритм разбивает полученный файл с серией жестов на отдельные фреймы.

В случае, если фрейм попадает в промежуток времени, когда не совершалось жеста, или промежуток, полностью лежащий внутри отрезка времени совершения жеста, данные в столбцах Expected и Required совпадают. В этом случае мы считаем, что классификатор правильно определяет тип жеста, если то же значение имеет колонка Result.

В случае, когда фрейм частично попадает на отрезок времени совершения жеста, а частично – на отрезок, где жест не совершается, допустимо получить в качестве результата как действительный жест $G_1 - G_4$, так и G_0 . Это не будет являться ошибкой, поэтому необходимо дополнительно указать, как такие случаи будут учитываться при вычислении матрицы неточностей.

Обозначим ячейку матрицы неточностей как «Столбец–Строка». Где первое значение – название столбца, т.е. экспертное решение, а второе – название строки, т.е. решение классификатора.

В таком случае, если Required не совпадает с Expected, а Result – это действительный жест либо G_0 , увеличиваем на 1 значение в ячейке «Действительный жест–Действительный жест».

Если же Result – это ошибочный жест, увеличиваем на 0.5 значение в ячейке «Действительный жест–Ошибочный жест» и в ячейке «None–Ошибочный жест».

4.4. Результаты вычисления метрик

Матрица неточностей – это матрица размера 5 на 5. Столбцы этой матрицы резервируются за экспертными решениями, а строки – за решениями нашего классификатора.

Число, стоящее на пересечении строки класса, который вернул классификатор, и столбца класса, к которому действительно относится данное измерение во время классификации описывает точность реализуемого подхода.

В табл. 1 приведена матрица неточностей для обучающей выборки, а в табл. 2 – матрица неточностей для валидационной выборки.

Т а б л и ц а 1

Матрица неточностей для обучающей выборки

	none	leftright	updown	clockwise	counterclock
none	3697.0	11.0	23.0	7.0	2.0
leftright	61.5	1384.0	2.0	1.0	0.5
updown	29.0	0.0	1657.0	0.0	0.0
clockwise	29.0	0.5	1.5	1524.0	0.0
counterclock	52.0	0.5	0.5	0.0	1524.0

Т а б л и ц а 2

Матрица неточностей для валидационной выборки

	none	leftright	updown	clockwise	counterclock
none	784.0	6.0	24.0	0.0	0.0
leftright	21.0	430.0	0.0	0.0	0.0
updown	15.0	0.0	465.0	0.0	0.0
clockwise	10.0	1.0	0.0	451.0	0.0
counterclock	12.0	0.0	0.0	0.0	429.0

По результатам эксперимента для обучающей выборки были получены метрики, приведенные в табл. 3.

Т а б л и ц а 3

Метрики для обучающей и валидационной выборки

	Обучающая выборка	Валидационная выборка
Accuracy	0.9779	0.9664
Precision	0.9746	0.9669
Recall	0.9848	0.9732
F-Measure	0.9797	0.9700

Как видно, для трех человек, не участвовавших в обучении классификатора, получены очень хорошие результаты. Это говорит о высокой эффективности предложенного алгоритма классификации.

Были также рассчитаны таблицы контингентности для каждого класса жестов.

Недиагональные элементы таблицы контингентности соответствуют вероятностям ложно-положительных и ложно-отрицательных срабатываний алгоритма.

Нужно заметить, что для класса жестов G_0 число ложно-положительных решений соответствует количеству случаев ошибочного пропуска жеста.

Соответственно число ложно-отрицательных срабатываний соответствует количеству случайных срабатываний алгоритма в то время, когда жест пользователем на самом деле не выполняется и обнаружение движения произошло случайно.

Изменяя требования к числу ложно-отрицательных срабатываний, мы можем управлять «чувствительностью» алгоритма.

Вероятности для всех классов указаны в процентах.

Т а б л и ц а 4

Таблица контингентности для использованных классов

		Экспертная оценка	
		Положительная	Отрицательная
Оценка системы для класса G_0 «None»	Положительная	95.1	0.9
	Отрицательная	5.9	99.1
Оценка системы для класса G_1 «L-R»	Положительная	99.0	0.8
	Отрицательная	1.0	99.2
Оценка системы для класса G_2 «U-D»	Положительная	97.7	0.4
	Отрицательная	2.3	99.6
Оценка системы для класса G_3 «CW»	Положительная	99.6	0.4
	Отрицательная	0.4	99.6
Оценка системы для класса G_4 «CCW»	Положительная	99.9	0.6
	Отрицательная	0.1	99.4

4.5. Размер обучающей выборки

Можно считать, что размер обучающей выборки является достаточным, если удается определять жесты с заданными вероятностными характеристиками.

В нашем эксперименте размер обучающей выборки был весьма невелик, но позволил определять жесты с параметрами, указанными в табл. 4. Достоверность распознавания оказалась для нашей выборки весьма высокой. Предполагается провести целенаправленное исследование параметров распознавания для различных выборок и различных групп пользователей, но эта работа выходит за рамки настоящего исследования.

При добавлении новых жестов систему необходимо будет дообучить, дополнив обучающую выборку таким образом, чтобы в результате параметры распознавания были лучше минимально приемлемых.

5. Выводы

В статье был предложен оригинальный подход к распознаванию жестов, выполняемых манипулятором в форм-факторе дирижерской или игровой палочки на основе рекуррентной нейронной сети с долгой краткосрочной памятью (LSTM).

Продемонстрировано эффективное распознавание несложных жестов разработанной нейронной сетью и возможность пополнения библиотеки распознаваемых жестов. Для полученных экспериментальных данных проведен расчет качества распознавания и подтверждена правильность выбранного подхода для решения поставленной задачи.

Обсуждение вопросов, связанных с созданием эффективного алфавита жестов и возможностями биометрического распознавания пользователя и диагностики его психофизического состояния, выходит за рамки данной статьи.

Описано разработанное устройство, использованное в экспериментах, и предложен принципиально новый подход к превращению поверхности устройства в сенсорную панель.

На основе созданного устройства и пакета программных решений возможно создание инновационного устройства и его вывод на рынок IoT в качестве устройства управления, ориентированного на детскую аудиторию.

Литература

1. *Madgwick S., Harrison A., Vaidyanathan R.* Estimation of IMU and MARG orientation using a gradient descent algorithm. Rehabilitation Robotics (ICORR) // 2011 IEEE International Conference. P. 1–7.
2. *Shin S., Sung W.* Dynamic hand gesture recognition for wearable devices with low complexity recurrent neural networks. Circuits and Systems (ISCAS) // IEEE International Symposium. 2016. P. 2274–2277.
3. *Guan Y., Ploetz T.* Ensembles of Deep LSTM Learners for Activity Recognition using Wearables // arXiv preprint arXiv:1703.09370. 2017.
4. *Chen Y. [et al.]* LSTM Networks for Mobile Human Activity Recognition. 2016.
5. *Kingma D., Ba J.* Adam: A method for stochastic optimization // arXiv preprint arXiv:1412.6980. 2014.

References

1. *Madgwick S., Harrison A., Vaidyanathan R.* Estimation of IMU and MARG orientation using a gradient descent algorithm. Rehabilitation Robotics (ICORR). 2011 IEEE International Conference. P. 1–7.
2. *Shin S., Sung W.* Dynamic hand gesture recognition for wearable devices with low complexity recurrent neural networks. Circuits and Systems (ISCAS). IEEE International Symposium. 2016. P. 2274–2277.
3. *Guan Y., Ploetz T.* Ensembles of Deep LSTM Learners for Activity Recognition using Wearables. arXiv preprint arXiv:1703.09370. 2017.
4. *Chen Y. [et al.]* LSTM Networks for Mobile Human Activity Recognition. 2016.
5. *Kingma D., Ba J.* Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980. 2014.

Поступила в редакцию 03.07.2017