

ИДЕНТИФИКАЦИЯ ЛИЧНОСТИ ПО БИОМЕТРИЧЕСКИМ ХАРАКТЕРИСТИКАМ ЖЕСТОВ НА СЕНСОРНОМ ЭКРАНЕ

К.И. Лейфер

студент, e-mail: leyfer.kirill@gmail.com

Д.Н. Лавров

к.т.н., доцент, e-mail: dmitry.lavrov72@gmail.com

Омский государственный университет им. Ф.М. Достоевский

Аннотация. В статье проводится исследование возможных способов идентификации пользователя по жестам, вводимым им на сенсорном экране смартфона.

Ключевые слова: биометрия, идентификация, сенсорный экран, смартфон.

Введение

В наши дни смартфоны, оснащённые сенсорными дисплеями, получили повсеместное распространение. Сенсорный экран в смартфонах применяется с 1992 года, и за это время он превратился в довольно точное и отзывчивое устройство ввода. В большинстве современных смартфонов тачскрин способен распознавать до 10 одновременных касаний с точностью до одного пикселя, а так же позволяет определять площадь каждого из касаний в широком диапазоне. Всё это позволяет отслеживать с высокой точностью каждый жест пользователя, оставляемый им на сенсорном экране. Можно предположить, что жесты, вводимые пользователем на сенсорном экране, могут обладать неким набором уникальных для каждого человека особенностей, то есть содержать в себе набор некоторых биометрических признаков. Сенсорный экран, исходя из перечисленных выше особенностей, вполне способен зафиксировать эти признаки, выступая в роли биометрического сенсора.

Поиск таких биометрических характеристик и построение на их основе системы идентификации пользователей представляют большой интерес с точки зрения информационной безопасности и будут подробно рассмотрены в данной статье.

Для поиска биометрических характеристик в сенсорных жестах необходимо накопить некоторое количество тестовых данных. Эту задачу можно возложить на клиент-серверное приложение. Клиентская часть должна устанавливаться на устройства тестовых пользователей и отслеживать все жесты, вводимые на сенсорном экране. Эта часть должна собирать всю необходимую статистику не создавая пользователю никаких неудобств, а затем автоматически отправлять

собранные данные на сервер. Серверная часть проекта должна сохранять всю статистику в базе данных.

После сбора данных необходимо провести исследование полученной информации и поиск биометрических признаков, позволяющих построить систему биометрической идентификации пользователей. Затем необходимо построить данную систему и протестировать её работу.

При разработке системы использовались руководства [1, 2]. Работа опиралась на ряд выводов, сделанных в [3], и проводилась в рамках программы исследований по биометрии кафедры компьютерных технологий и сетей факультета компьютерных наук ОмГУ им. Ф.М. Достоевского [4–18]. В перечисленных работах исследуются статические и динамические биометрические характеристики. Наибольшую сложность вызывают динамические характеристики: идентификация по голосу, по походке, клавиатурный почерк и др. Жесты на сенсорном экране также относятся к динамическим характеристикам.

1. Описание и принцип работы проекта TouchLogger

Клиентская часть

Разработанное клиент-серверное приложение получило название TouchLogger. Клиентская часть представляет собой android-приложение, которое устанавливается на смартфоны с включённым полным доступом к устройству. Сбор данных клиентской частью осуществляется из любого приложения, в котором работает пользователь, будь то клиент социальной сети, экран блокировки или виртуальная клавиатура. Пользователь может сгенерировать довольно большой объём данных, поэтому их отправка на сервер осуществляется только при наличии соединения с Интернетом по Wi-Fi. Перед отправкой собранные данные помещаются во временное хранилище на накопителе устройства, и как только устанавливается Wi-Fi соединение, все накопленные данные автоматически отправляются на сервер, и временное хранилище очищается.

Стоит учитывать, что жесты сенсорного ввода могут содержать любую информацию о пользователе: от данных кредитных карт до личной переписки. Поэтому, несмотря на небольшой масштаб исследования, стоит предусмотреть некоторые механизмы защиты информации. Для этого в проекте TouchLogger используется следующий криптографический протокол:

$$A \rightarrow B : E_{K_o}(K), E_K(M),$$

где M — открытое сообщение, K_o — открытый ключ алгоритма RSA, K — сеансовый ключ, A — клиент, B — сервер.

Клиент, сгенерировав сеансовый ключ K , шифрует им своё сообщение, а сам ключ K шифрует с помощью заранее известного всем клиентам открытого ключа K_o . Зашифрованное сообщение вместе с зашифрованным ключом клиент отправляет на сервер. Сервер с помощью секретного ключа K_c расшифровывает

сеансовый ключ и уже с его помощью расшифровывает сообщение и помещает его в базу данных.

Итак, собираемые данные передаются по сети в зашифрованном виде и используются только в исследовательских целях. Однако тестовый пользователь всё равно может счесть эти меры сохранности личных данных недостаточными и найти какой-либо способ избежать отправки части данных на сервер. Бороться с этим невозможно, поэтому самый лучший выход из данной ситуации — внедрить в клиентскую часть функцию «паузы», которая будет приостанавливать сбор данных по желанию пользователя.

Серверная часть

Серверная часть написана на языке JavaScript с использованием платформы Node.js. Данная платформа позволяет в довольно короткий срок разработать несложный сервер, что и стало основной причиной её выбора. В качестве СУБД была выбрана MongoDB. Эта СУБД представляет документо-ориентированный подход к хранению данных, иначе называемый NoSQL (в противопоставление традиционным SQL базам данных). Выбор был продиктован необходимостью хранения большого количества вложенных данных, а также тем, что документ в MongoDB можно напрямую создать из JavaScript объекта или из документа в формате JSON. Последняя особенность также значительно упростила и ускорила разработку серверной части.

2. Анализ полученных данных

В сборе пользовательских данных приняли участие 19 пользователей, установивших на свои устройства клиентскую часть проекта. Экраны у большей части смартфонов имели диагональ 5 дюймов и разрешение 1920x1080 пикселей. В результате за 14 дней тестирования в сумме было получено около 200000 жестов.

Классификация данных

Для того, чтобы проанализировать полученные данные, необходимо классифицировать их некоторым образом с целью дальнейшего исследования отдельных классов. В этой работе данные были классифицированы следующим образом:

1. по количеству касаний в одном жесте:
 - а) однопальцевые
 - б) многопальцевые (multitouch)
2. по типу касания:

- а) «тачи» (от англ. «touch» — прикосновение) — простое нажатие на экран. Для этих жестов характерно небольшое (часто — нулевое) расстояние, проделанное пальцем от момента касания до окончания жеста.
- б) «свайпы» (от англ. «swipe» — происхождение термина неизвестно, означает быстрый «смахивающий» жест на сенсорном экране). Для таких жестов характерно большое расстояние, проделанное пальцем за время жеста.

Для разбиения всех однопальцевых жестов на тачи и свайпы необходимо вычислить расстояние, пройденное пальцем в течение жеста. Затем необходимо найти пороговое значение этого расстояния так, чтобы все жесты, длина которых меньше этого значения, можно было отнести к тачам, а остальные — к свайпам. Очевидно, подобная задача стояла и перед разработчиками ОС Android, поэтому в исходных кодах можно отыскать величину этого порогового значения, она составляет 4 dp (dp — density-independent pixels, «пиксели, не зависящие от плотности» — абстрактная единица измерения длины, эквивалент одного пикселя при плотности экрана 160 точек на дюйм). На экране смартфона с разрешением 1920x1080 и диагональю 5 дюймов 4 dp примерно соответствует 12 пикселям. Итак, все жесты длиной более 12 пикселей будем считать свайпами, остальные — тачами.

Имея пороговое значение, классифицируем все жесты на одно- и многопальцевые, а первые разделим на тачи и свайпы. Построим процентное соотношение этих жестов (см. рис. 1).

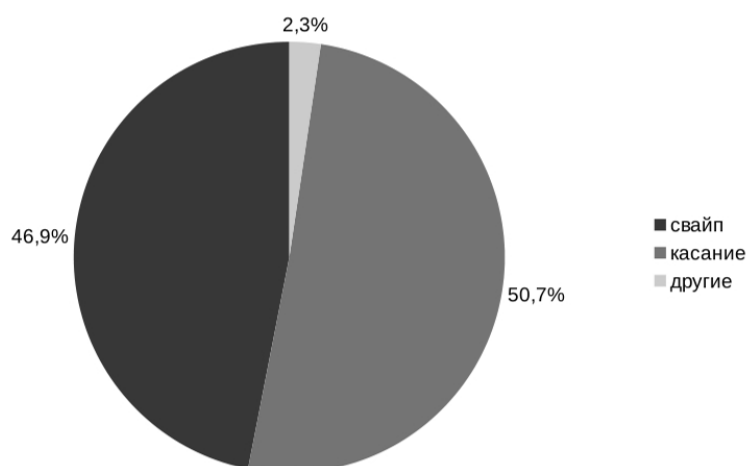


Рис. 1. Распределение жестов на тачи, свайпы и многопальцевые жесты

Как видно из графика, многопальцевые (мульти-тач) жесты составляют менее 3% от общего количества событий ввода, что оставляет для анализа при-

близительно по 300 таких жестов с каждого пользователя. Этого явно недостаточно для полноценного исследования, поэтому в дальнейшем будем анализировать только тачи и свайпы.

Анализ распределения тачей по площади касания

Для каждого события ввода известно большое количество различных параметров, в том числе площадь касания. Это число от 0 до 1. У каждого пользователя распределение этого значения должно отличаться в силу различной толщины пальцев и различной частоты использования разных пальцев для ввода данных. Проверим эту гипотезу, построив распределение всех тачей по значению средней площади касания, по площади касания в начале тача и в конце тача. Пример полученных распределений для двух разных пользователей приведён на рис. 2.

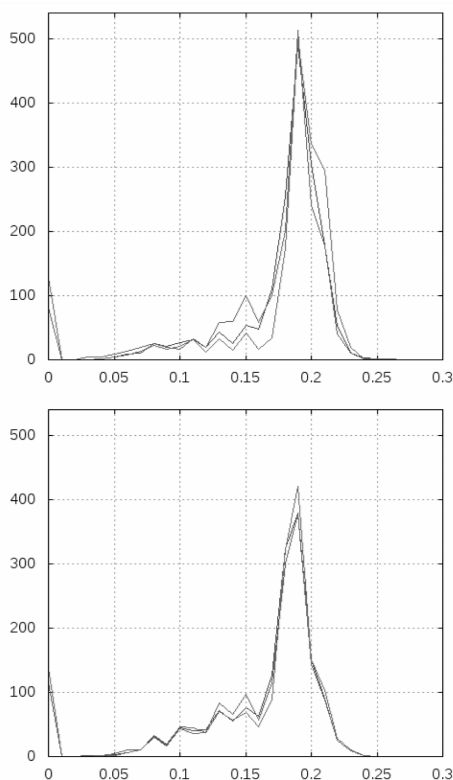


Рис. 2. Распределение тачей по площади касания

По оси X отложено значение площади касания, по оси Y — количество жестов. Три разных линии на графике представляют распределение средней площади касания, площади касания в начале и конце жеста. График отражает незначительные отличия в распределении площади касания среди разных пользователей, соответственно, ни одна из трёх характеристик не является уникальной для пользователей, следовательно, построение на их основе системы

идентификации невозможно.

Анализ тачей по продолжительности

У разных пользователей продолжительность касаний, зависящая от многих параметров, может значительно отличаться. Проверим это гипотезу, построив распределение тачей по продолжительности касания. На рис. 3 приведён пример такого распределения для четырёх пользователей на основании данных, полученных за некоторый промежуток времени.

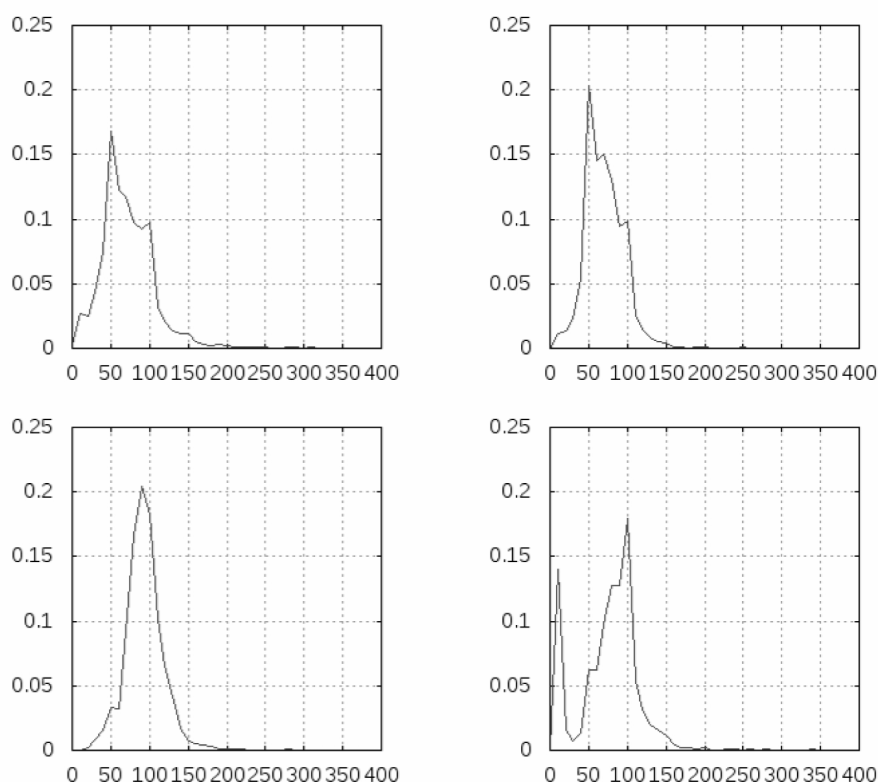


Рис. 3. Распределение тачей по продолжительности

На этом рисунке по оси X отложена продолжительность касаний в миллисекундах, по оси Y — доля жестов данной продолжительности. На первый взгляд, формой графики значительно отличаются друг от друга, а значит, распределение тачей по продолжительности может быть использовано в качестве биометрической характеристики в системе биометрической идентификации. Однако, если построить графики распределения этой характеристики для одного и того же пользователя за разные промежутки времени, они так же будут значительно отличаться формой. Отсюда следует, что несмотря на наличие некой уникальности этой характеристики, система идентификации, построенная на её основе, покажет довольно плохие результаты.

Анализ распределения свайпов по расстоянию

Другая характеристика, которая может быть уникальной для пользователей — это распределение свайпов по их длине. Для ускорения расчётов будем считать, что длина свайпа — это расстояние между точкой касания и точкой, где пользователь отпустил палец. То есть в расчётах будем использовать не длину пути, а расстояние. Такое упрощение обосновано тем, что распределение свайпов по расстоянию практически не отличается от распределения по длине пути, но в то же время вычисляется значительно быстрее.

Итак, построим распределение свайпов по расстоянию. На рис. 4 приведён пример графиков распределения этой величины для четырёх разных пользователей.

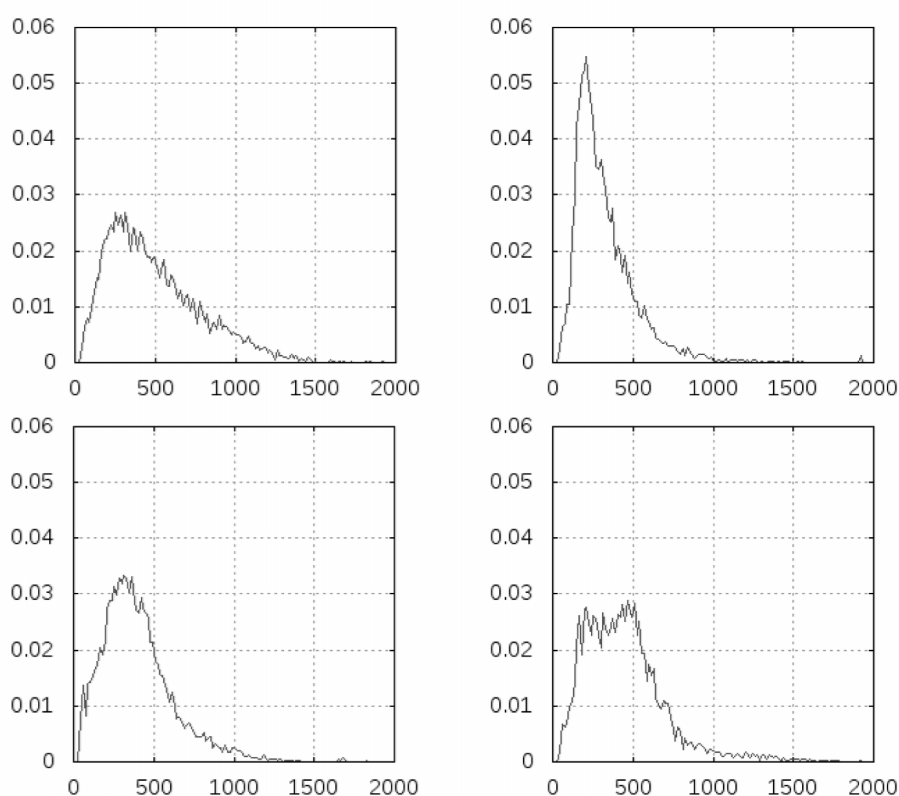


Рис. 4. Распределение свайпов по расстоянию

Графики отличаются по форме, более того, при выборке данных для одного и того же пользователя за разные промежутки времени форма графика распределения изменяется незначительно. Это говорит о том, что данная характеристика является уникальной для каждого пользователя и по ней можно построить систему биометрической идентификации. Однако не стоит ожидать от системы, построенной по данной характеристике, выдающихся результатов. На рисунке выше видно, что отличия в форме графиков незначительные, а для большого количества пользователей станут ещё менее различимыми.

Анализ распределения свайпов по продолжительности

Ещё одна характеристика, представляющая интерес — это распределение свайпов по продолжительности в миллисекундах. Построим распределение этой величины и рассмотрим несколько примеров такого распределения на рис. 5.

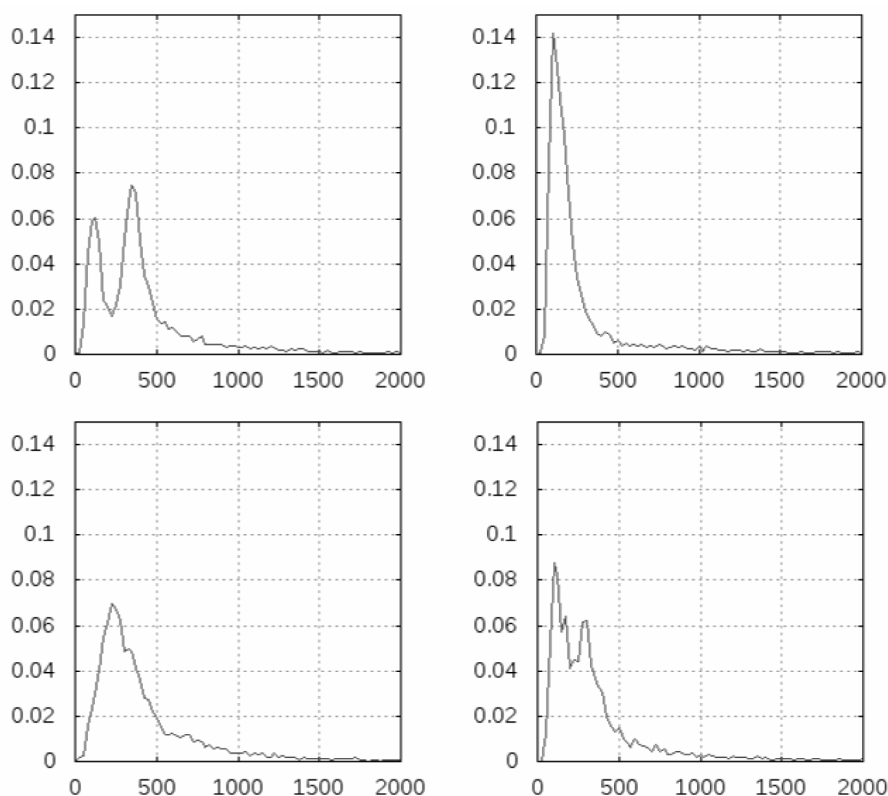


Рис. 5. Распределение свайпов по продолжительности

Данная характеристика демонстрирует на графиках гораздо более сильное разнообразие, которое может обеспечить хороший процент успешных идентификаций пользователя по этим признакам. Более того, распределение данных, полученных в разные промежутки времени для одного и того же пользователя, будет практически идентичным. Значит, эту характеристику также можно использовать для построения системы идентификации.

Другие способы анализа данных

Кроме построения распределения некоторой величины существуют и другие способы анализа пользовательских данных. На рис. 6 представлено распределение тачей по координатам.

Если применить к этим данным некий алгоритм кластеризации, то для каждого пользователя можно будет построить эталонный набор кластеров точек касания, с которыми затем можно будет сравнивать поступающие на вход тестовые данные.

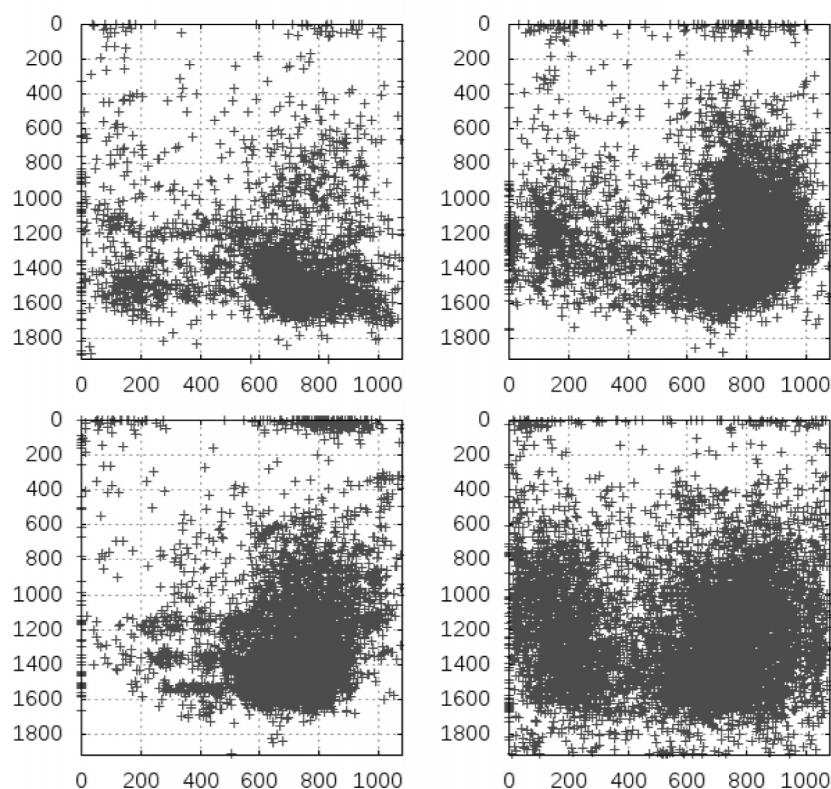


Рис. 6. Распределение точек по координатам

На рис. 7 для четырёх различных пользователей показаны графики другой характеристики, для которой можно построить эталон подобным образом. На этом графике по оси X отмечена продолжительность свайпа в мс, а по оси Y — его длина в пикселях. Таким образом, одна точка на графике некоторым образом отражает скорость одного свайпа.

Перечисленные выше способы анализа данных представляют большой интерес и обязательно будут исследованы в следующих работах.

3. Построение системы идентификации

Итак, в предыдущих пунктах были рассмотрены различные характеристики, извлечённые из тестовых данных. Самыми многообещающими оказались распределения свайпов по продолжительности и длине, поэтому построим две системы идентификации по каждой из этих характеристик и сравним процент успешных идентификаций для этих систем.

Для формирования эталонов выберем для каждого пользователя первые по времени поступления свайпы в количестве 1500 штук. Для полученных выборок построим распределения по продолжительности свайпов и по их длине. Таким образом, для каждого пользователя будут сформированы эталоны по двум признакам. Теперь необходимо построить массив тестовых данных. Для

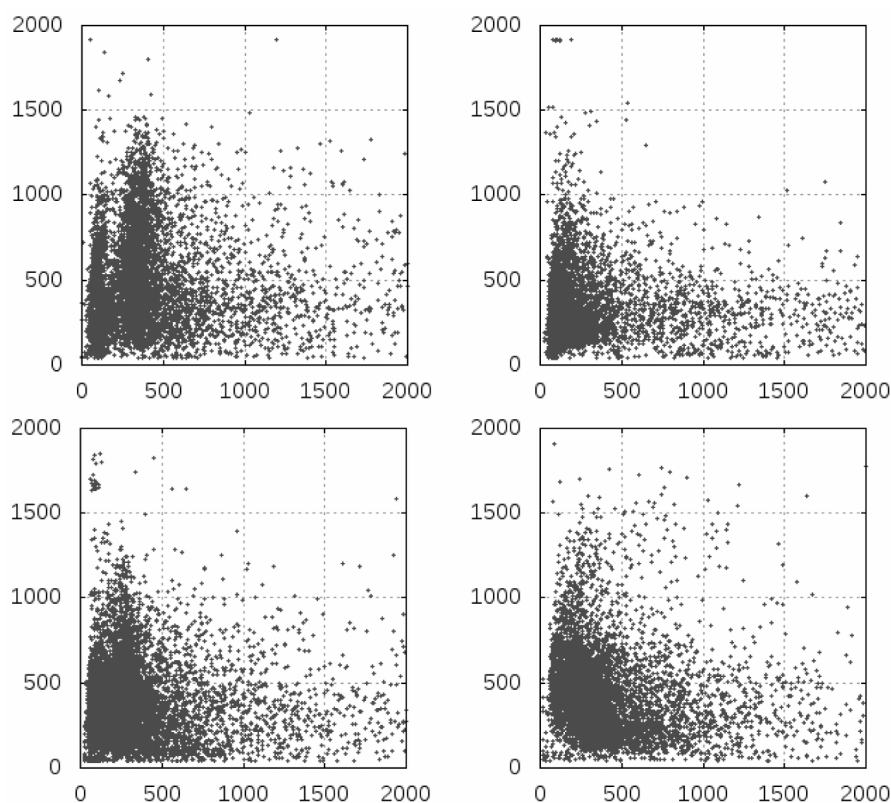


Рис. 7. Распределение свайпов по скорости

этого все оставшиеся события ввода для каждого из пользователей разобьём на выборки по 200 свайпов. Построим распределения нужных характеристик для каждой из полученных выборок. Получим набор тестовых данных, которые затем будем сравнивать с эталонами.

Распределение зависимости количества свайпов от некоторого параметра можно представить в виде вектора. Рассмотрим это на примере. Пусть у пользователя было взято 1500 свайпов. Посмотрим, как много свайпов имеют длину от 0 до 199 пикселей, от 200 до 399 пикселей и так далее и запишем результат в таблицу 1. Количество свайпов, находящееся в определённых пределах, можно нормировать, разделив на общее количество свайпов. Эти данные представлены в третьей строке таблицы 1.

Теперь в третьей строке записан в точности вектор, характеризующий распределение свайпов по длине в пикселях. Однако разные смартфоны обладают экранами различного разрешения, и это отличие также необходимо устранить при идентификации пользователей. Для этого нормирование проводилось не только по количеству свайпов, но и по их длине в пикселях. Длина каждого свайпа делилась на максимально возможное значение этого параметра ($\sqrt{w^2 + h^2}$, где w — ширина экрана в пикселях, h — высота экрана в пикселях), поскольку разрешения экранов смартфонов, с которых приходили тестовые данные, также были известны. Ещё одно отличие примера выше от построенной

Таблица 1. Таблица распределения свайпов по длине

Длина свайпа, px	от 0 до 199	от 200 до 399	от 400 до 599	от 600 до 799	от 800 до 999	от 1000 до 1199
Кол-во свайпов данной длины	453	581	252	131	70	13
Доля свайпов данной длины	0,3	0,39	0,17	0,09	0,05	0,01

системы идентификации заключается в размере векторов. В примере выше шаг составляет 200 пикселей, что даёт вектор из 6 элементов. В реальной работе при построении вектора распределения свайпов по длине шаг составлял 0,02, соответственно, эталонные и тестовые векторы состояли из 50 элементов. Векторы распределения свайпов по продолжительности имели шаг в 5 миллисекунд, что при исследовании свайпов продолжительностью до 2 секунд дало векторы размером в 400 элементов. Также для каждого вектора, будь то тестовый или эталонный, известен идентификатор того пользователя, чьи данные были использованы для построения этого вектора. Это необходимо для тестирования системы идентификации, так как позволяет понять, принадлежат ли два разных вектора одному и тому же пользователю.

Тестирование системы идентификации

После формирования эталонов и тестовых векторов запускается алгоритм идентификации пользователей. Алгоритм для каждого вектора из массива тестовых данных находит наиболее близкий эталонный вектор. Если тестовый вектор и ближайший к нему эталон принадлежат одному и тому же пользователю, идентификация признаётся успешной, иначе считается, что произошла ошибка второго рода. В конце вычисляется процент успешных идентификаций для каждого из исследуемых параметров.

В итоге, были получены следующие результаты:

1. Количество успешных идентификаций по распределению длины свайпов: 51%
2. Количество успешных идентификаций по распределению продолжительности свайпов: 93%

Это неплохие результаты для динамической биометрической характеристики. Однако текущая версия алгоритма не может отличить зарегистрированных пользователей от незарегистрированных, то есть не учитывает того, что в системе могут находиться тестовые данные пользователей, для которых не существует эталонов. Для того, чтобы это исправить, необходимо ввести некоторое пороговое значение и модифицировать алгоритм таким образом, чтобы он регистрировал ошибки первого и второго рода.

Пороговым значением в данном случае будет максимально допустимое расстояние между тестовым и эталонным векторами. Если наименьшее расстояние между тестовым вектором и всеми эталонами превысит пороговое значение, пользователь, чьи данные сформировали тестовый вектор, будет признан незарегистрированным в системе. Если же на самом деле для данного пользователя существует эталонный вектор, будет сгенерирована ошибка первого рода. Если минимальное расстояние не превысит порогового значения, считается, что пользователь в системе зарегистрирован и алгоритм попытается идентифицировать пользователя в системе так же, как это было описано выше, до введения в алгоритм порога.

При тестировании модифицированного алгоритма для формирования тестовых данных были использованы в том числе данные пользователей, которые не смогли отправить достаточное количество жестов для формирования эталонов и поэтому в предыдущей версии алгоритма не рассматривались. Это позволило сформировать дополнительные тестовые данные заведомо незарегистрированных в системе пользователей. Модифицированный алгоритм запускался только для распределения свайпов по продолжительности в миллисекундах, так как эта характеристика показала лучшие результаты в предыдущем тесте. Пороговый коэффициент был выбран таким образом, чтобы на тестовой базе данных количество ошибок первого и второго рода совпадало. В результате при выбранном пороговом коэффициенте 1,35 количество успешных идентификаций составило 76% от общего количества тестовых данных, количество ошибок первого и второго рода, соответственно, по 12%.

Снижение эффективности алгоритма при введении в систему незарегистрированных пользователей показывает необходимость поиска более совершенных биометрических характеристик в событиях ввода, чему будут посвящены следующие работы.

Заключение

В результате проделанной работы было реализовано клиент-серверное приложение для сбора событий ввода с сенсорного экрана и отправки этих событий на сервер. Сформирована тестовая база из более чем 200000 жестов, собранных с 19 пользователей. Проведён анализ большого числа характеристик данных жестов на предмет их уникальности среди пользователей. Выявлены характеристики, на основе которых может быть построена система биометрической идентификации. Реализована и протестирована система биометрической идентификации на основе распределения свайпов по длине и по продолжительности. Получены удовлетворительные результаты тестирования.

ЛИТЕРАТУРА

1. Майер Р. Android 4. Программирование приложений для планшетных компьютеров и смартфонов. Эксмо, 2013. 816 с.

2. Yaghmour K. Embedded Android. Porting, Extending and Customizing. O'reilly, 2013. 412 p.
3. Angulo J., Wästlund E. Exploring Touch-screen Biometrics for User Identification on Smart Phones [Электронный ресурс] URL: <http://dl.ifip.org/db/conf/primelife/primelife2011/AnguloW11.pdf> (дата обращения: 02.11.2015)
4. Шарун И.В., Лавров Д.Н. Проектирование и разработка программного обеспечения для моделирования идентификации по геометрии руки // Математическое и компьютерное моделирование [Электронный ресурс] : материалы III Международной научной конференции (Омск, 12 ноября 2015 г.) / [отв. за вып. И. П. Бесценный]. – Электрон. текст. дан. – Омск : Изд-во Ом. гос. ун-та, 2015. 1 электрон. опт. диск (CD-ROM); 12 см. С. 151-152.
5. Лазаренко П.Н., Лавров Д.Н. Оценка алгоритмов скелетизации в рамках исследований по сегментации кровеносных сосудов // Математическое и компьютерное моделирование [Электронный ресурс] : материалы III Международной научной конференции (Омск, 12 ноября 2015 г.) / [отв. за вып. И. П. Бесценный]. Электрон. текст. дан. Омск : Изд-во Ом. гос. ун-та, 2015. 1 электрон. опт. диск (CD-ROM); 12 см. С. 153–155.
6. Гришенкова Н.П., Лавров Д.Н. Обзор методов идентификации человека по радужной оболочке глаза // Математические структуры и моделирование. Омск : Ом. гос. ун-т, 2014. №1(29). С. 43–64.
7. А.В. Альтергот, О.А. Вишнякова, Лавров Д.Н. Формат данных на основе XML-разметки для обмена биометрическими образцами // Математические структуры и моделирование. Омск : Омский государственный университет, 2013. Т. 28, № 2. С. 60–67.
8. Альтергот А.В., Вишнякова О.А., Долгополов В.П., Лавров Д.Н. Спецификатор бинарных данных для XML-формата обмена биометрическими данными (тезисы) печатная Математическое и компьютерное моделирование : сборник материалов научной конференции (Омск, 18 октября 2013 г.). Омск : Изд-во Ом. гос. ун-та, 2013. С. 49–52.
9. Вишнякова О.А., Казанцева А.Г., Лавров Д.Н. Проектирование мультибиометрической базы данных для компьютерного моделирования распознавания личности // Вестник Омского университета. 2013. N.4. С. 207–212.
10. Вишнякова О.А., Лавров Д.Н. Формат обмена данными в системе сбора и обработки биометрических образцов // Международная научно-практическая конференция «Информационные ресурсы в образовании». Нижневартовск : НГГУ, 17-19 апреля 2013. С. 146–148
11. Первушин Е.А., Лавров Д.Н. Алгоритм учёта влияния параметров речевого сигнала в задаче распознавания дикторов // Вестник Омского университета. 2011. N.4. С. 153–156.
12. Первушин Е.А., Лавров Д.Н. Алгоритм извлечения признаков речевого сигнала во временной области для задачи распознавания дикторов // Вестник Омского университета. 2011. №.2. С. 182–185.
13. Первушин Е.А., Лавров Д.Н. Использование нормированных кадров сигнала в задаче распознавания по походке // Математические структуры и моделирование. Омск : Омский государственный университет им. Ф.М. Достоевского. 2011. Вып.23. С. 38–42.
14. Казанцева А.Г., Лавров Д.Н. Распознавание личности по походке на основе

- wavelet-параметризации показаний акселерометров // Математические структуры и моделирование. Омск : Омский государственный университет им. Ф.М. Достоевского, 2011. Вып. 23. С. 31–37.
15. Вишнякова О.А., Лавров Д.Н. Подходы к задаче идентификации диктора (научная статья) (РИНЦ). печатная Математические структуры и моделирование: Под ред. А.К.Гуца. Омск : Омский государственный университет им. Ф.М.Достоевского, 2011. Вып. 22. С. 48–54.
 16. Первушин Е.А., Лавров Д.Н. Система идентификации диктора на основе выделения информативных участков речевого сигнала // Информационные технологии и автоматизация управления. Матер. II межвуз. науч.-практ. конф. ОмГТУ 06-09 апреля 2010 года. / Под общ. науч. ред. В.Н. Задорожного. Омск : Изд-во ОмГТУ, 2010. С. 188-189.
 17. О.А. Вишнякова, Лавров Д.Н. Идентификация речевого сигнала на базе вейвлет-преобразования // Информационные технологии и автоматизация управления: матер. межвуз. науч.-практ. конф. ОмГТУ 2-24 апреля 2009 года. / Под общ. науч. ред. В.Н. Задорожного. Омск : Изд-во ОмГТУ, 2009. С. 167-168.
 18. Венедиктова Е.В., Лавров Д.Н. Идентификация голоса по спектральным характеристикам с помощью линейного классификатора // Математические структуры и моделирование : Под ред. А.К.Гуца. Омск : ОмГУ, 2008. Вып. 18. С. 108–115

IDENTIFICATION OF PERSON BASED ON BIOMETRIC CHARACTERISTICS OF TOUCH SCREEN GESTURES

K.I. Leyfer

Student, e-mail: leyfer.kirill@gmail.com

D.N. Lavrov

Ph.D. (Eng.), Associate Professor, e-mail: dmitry.lavrov72@gmail.com

Omsk State University n.a. F.M. Dostoevskiy

Abstract. The article presents a study of possible ways to identify a user by his touch gestures on the screen of smartphone.

Keywords: gesrutes, identification, screen, smartphone.